

# Erlang Web atıları



15 Ekim 2010  
Özgür Web Günleri  
Yeditepe Üniversitesi



# Agner Krarup Erlang



Joe Armstrong



# Neden Erlang?

- Birden fazla çekirdekli işlemcilerle hızlı çalışan uygulama ihtiyacı.
- Hata toleranslı (fault-tolerant) ve kod değiştirme anında servis dışı kalmayan uygulamalara olan dayanılmaz arzu
- Yazılacak kod satırlarında ciddi azalma.
- Artan yük karşılama sorununda yüksek başarımlar
- Dağıtık hesaplama ihtiyaçları
- Dağıtık sistemlerde 1 numero! Bulutlar, bulutlarrrr.



# Kod Neye Benziyor?

```
ts_utils.erl (~/.development/noesis...lang/apps/tsung-svn/src/tsung) - VIM
%% Func: keylsearch/2
%% Purpose: wrapper around httpd_utils module funs (maybe one day
%%          these functions will be added to the stdlib)
%%-----
keylsearch(Tuple,String)->
    case release_is_newer_or_eq("5.7") of %% should be removed in R13B
        true ->
            proplists:get_value(String,Tuple);
        false ->
            httpd_util:keylsearch(Tuple,String)
    end.

%%-----
%% Func: mkeylsearch/2
%% Purpose: multiple keylsearch:
%% Take as input list of {Key, Value} tuples (length 2).
%% Return the list of values corresponding to a given key
%% It is assumed here that there might be several identical keys in the list
%% unlike the lists:key... functions.
%%-----
mkeylsearch(List, Key) ->
    Results = lists:foldl(
        fun({MatchKey, Value}, Acc) when MatchKey == Key ->
            [Value | Acc];
        ({_OtherKey, _Value}, Acc) ->
            Acc
        end,
        [],
        List),
    case Results of
        [] -> undefined;
        Results -> lists:reverse(Results)
    end.

232,5 31%
```

# Ama Fakat Neden Erlang???

- Yüksek-seviye yapılar, (High-Level Constructs)  
Örnek: TCP segmentlerinin decode edilmesi

```
decode(<< SourcePort:16, DestinationPort:16,  
        SequenceNumber:32,  
        AckNumber:32,  
        DataOffset:4, _Reserved:4,  
        Flags:8, WindowSize:16,  
        Checksum:16, UrgentPointer:16,  
        Payload/binary>>) when DataOffset>4 ...
```

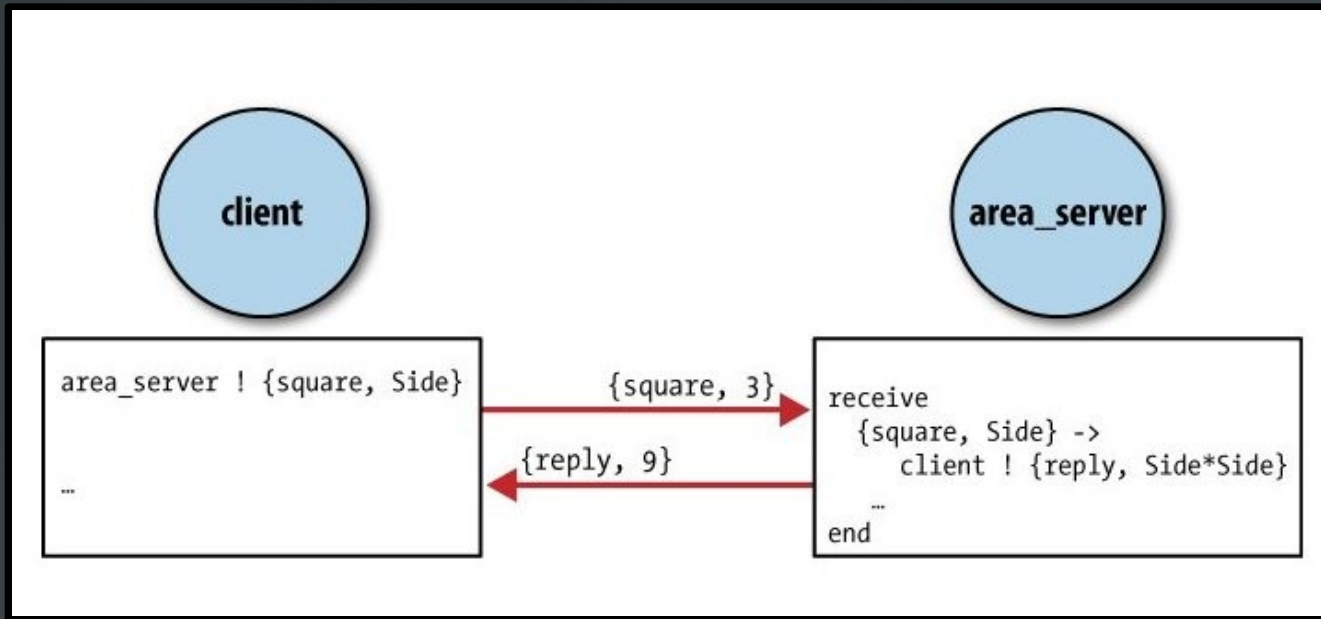
- Bunu C ya da Java ile yapmayı düşünün :-)

- Pattern matching, kompleks veri yapılarından kolayca bilgi çekmek
- Bit sequence pattern matching, örneği yukarda
- Fonksiyonlar first-class datadır. Başka fonksiyonlarca çağrılabilir, değişkene gömülebilirler
- List comprehension, listeler üzerinde her türlü düzenleme, list filters, generators





# Ölçeklenebilir, Güvenli ve Verimli Paralellik



"area server" Prosesi şekillerin çevresini hesaplayıp istemciye gönderiyor.

- Her Erlang prosesi kendi bellek alanında çalışır ve kendi heap ve stack'i vardır, proses başına OS thread açılmaz,
- Deadlock gibi sıkıntılardan kurtulmak için, prosesler birbirlerinin işine karışmazlar,
- Prosesler, birbirleri arasında mesaj gönderme yoluyla iletişim kurarlar, mesaj herhangi bir Erlang türü veri tipi olabilir,
- Mesaj gönderimi asenkronudur, mesajı bir kere gönderdi mi, proses işine devam eder,
- Mesajlar proses mail boxtan seçilerek getirilir, böylece gelip gelmedikleri konusunda endişe edilmez. Bu sayede çok daha sağlam (robust) bir paralel çalışma elde edilir.

# Ölçeklenebilir, Güvenli ve Verimli Paralellik

- Hızlı ve ölçeklendirebilir paralel işlemler,
- Kaç adet birbirinden bağımsız proses olduğuna bakılmaksızın, proses yaratma süreleri milisaniye aralığına iner,
- Mesaj geçme sayesinde, aynı VM içinde, verinin bir prosesin bellek alanından diğerine kopyalanması milisaniyeler içinde olur.



# Sağlamlık

- Prosesler birbirlerine bağlanıp durumları kontrol edilebilir.

OTP middleware kütüphaneleri;

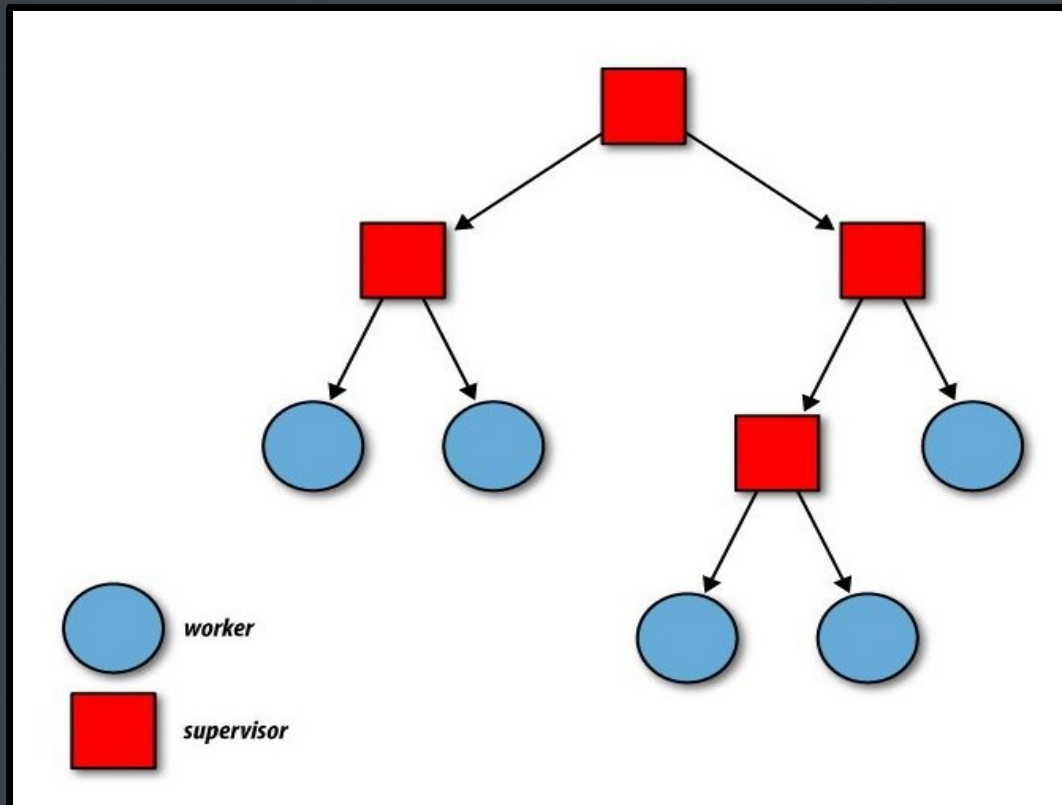
- İstisna izleme (exception monitoring)
- Hata düzeltme (error handling)
- Jenerik yapılar; sunucu, finite state machine, event handlers
- Code profiling
- Unit tests





# Dağıtık Hesaplama (Computation)

- Dağıtım modeli TCP/IP tabanlıdır,
- Sunucunuzun nerede olduğu farketmez, (location transparency)
- Prosesler kolaylıkla bir sunucu kümesine dağıtılabilir,



# Entegrasyon ve Açıklık

- C, Java, Ruby, Python, Perl, Lisp gibi diller Erlang ile birlikte çalışabilir,
- C ve Java için yüksek seviye kütüphaneler sayesinde eski (legacy) kodlar ya da ihtiyaç duyulan uygulamalar, dağıtık Erlang noktalarında çalıştırılabilir,



# Pazarlamacılar İçin Neden Erlang???

- Aynı işi 100 sunucu yerine 10 sunucuda yapabilirsiniz.
- Aynı iş için 120.000 satır kod yerine 40.000 satır kod yazılır. Adam saat maliyetleri hayli aşağı düşer.
- Ölçeklendirme sorunu olan her projede uzun zamanlı projelendirme maliyetlerini düşürür,
- Kritik uygulamalarda sistem hiç durdurulmadan çalışabilirsiniz,
- Mevcut yazılım alt yapısı, ağır yük nedeniyle kullanılamaz hale geldiyse mutlaka öneriniz,
- Kimsenin henüz bilmediği bir şey, hayli gizemli ve karizmatik bir etki yaratabilir ;-)



# Kimler Kullanıyor?

- Amazon, SimpleDB servisi
- Yahoo!, Delicious servisi
- Facebook chat servisi
- T-Mobile SMS ve yetkilendirme sistemi
- Motorola call processing ürünleri
- Ericsson GPRS ve 3G mobil ağları
- Goldman Sachs internal borsa sistemi

## Meşhur Uygulamalar

- Wings 3D, model ve doku poligon örgüleri.
- Ejabberd (XMPP) tabanlı anlık mesaj sunucusu
- CouchDB “schema-less” dokuman-tabanlı veritabanı
- RabbitMQ, AMQP mesaj protokolü uyarlaması







# Erlang Web Çatıları



- Erlang Web
- Nitrogen Web Framework
- Zotonic
- Chicago Boss
- Erylweb



# Bileşenler

- Web Servers
  - INETS
  - YAWS
  - Mochiweb
- Web Server Containers
  - Simple Bridge
  - EWGI
- REST Toolkit
  - Webmachine



# Nitrogen Web Framework



# Nitrogen Web Framework

- Event driven
  - AJAX
  - Jquery
  - Comet
  - Özel template sistemi
  - Data binding
- 
- Mochiweb, YAWS ya da INETS ile kullanılabilir



# Erlang Web

- MVC
- XML to XHTML template engine
- ErlyDTL templates
- Wpart Templates
- AJAX desteği zayıf
- Comet özelliği yok
- Mnesia veya CouchDB bağlantısı





# Zotonic



# Zotonic

- Event driven AJAX
  - MVC
  - JQuery
  - Web sockets ve Comet
  - ErlyDTL Django templates
  - PostgreSQL
- 
- Mochiweb, Webmachine



# Chicago Boss

- Ruby on Rails görünümlü Erlang
- MVC
- ErlyDTL Django templates
- PostgreSQL, MySQL, Tokyo Tyrant, Mnesia
- Modelden kod üretimi
  
- Mochiweb, Misultin, SimpleBridge



# Karşılaştırma

	Zotonic	Chicago Boss	Erlang Web	Nitrogen
<b>MİMARİ</b>				
Event-driven	✓	-	-	✓
Data Model	✓	✓	✓	-
View/Controller	✓	✓	✓	✓
Hata loglama	✓	✓	✓	✓
DATA MODEL				
Kod üretimi	-	✓	-	yok
Data onaylama	✓	✓	✓	
Edoc Üretimi	-	✓	-	
I18N/L10N	✓			
I18n özellikleri	✓	✓	✓	-
<b>AJAX</b>				
JSON üretimi	✓	✓	-	✓
Comet	✓	-	-	✓
JS form onaylama	✓	-	-	✓
<b>VERİTABANI DESTEĞİ</b>				yok
CouchDB	-	-	✓	
Mnesia	-	✓	✓	
MySQL	-	✓	-	
Postgresql	✓	✓	-	
Tokyo Tyrant	-	✓	-	
<b>ŞABLONLAR</b>				
ErlyDTL	✓	✓	✓	-
ErITL	-	-	-	-
WparT	-	-	✓	-
Nitrogen Records	-	-	-	✓
<b>CONTAINER</b>				
EWGI	-	-	✓	-
SimpleBridge	-	✓	-	✓
Webmachine	✓	-	-	-
<b>WEB SERVERS</b>				
Inets	-	-	✓	✓
Mochiweb	✓	✓	-	✓
Yaws	-	-	✓	✓



# Sorular





# Kaynakça

## Fun

- Epic Erlang Filmi: <http://www.youtube.com/watch?v=uKfKtXYLG78>

## Erlang öğrenmek için

- <http://learnyouosomeerlang.com/>
- <http://www.erlang.org>

## Kitaplar

- Programming Erlang: Software for a Concurrent World <http://www.pragprog.com/titles/jaerlang/programming-erlang>
- Erlang Programming: <http://oreilly.com/catalog/9780596518189>

## Web Çatıları

- Zotonic: <http://www.zotonic.com/>
- Erlang Web: <http://www.erlang-web.org/>
- Nitrogen: <http://nitrogenproject.com/>
- Erylweb: <http://www.erlyweb.org/>
- Chicago Boss: <http://www.chicagoboss.org>





[kunthar@gmail.com](mailto:kunthar@gmail.com)

