JavaCro

Elasticsearch as a search alternative to a relational database

PRIVREDNA BANKA ZAGREB

Kristijan Duvnjak & Mladen Maravić

JavaCro

What is Elasticsearch?

What is Elasticsearch (ES)?

JavaCro

- Document-oriented schema-free "database"
- Built on top of Apache Lucene
- Real-time search and data analytics
- Full-text search
- Distributed (horizontal scalability)
- High-avalability
- REST API

"Open Source (Apache 2)

distributed

RESTful

search engine

built on top of Lucene"

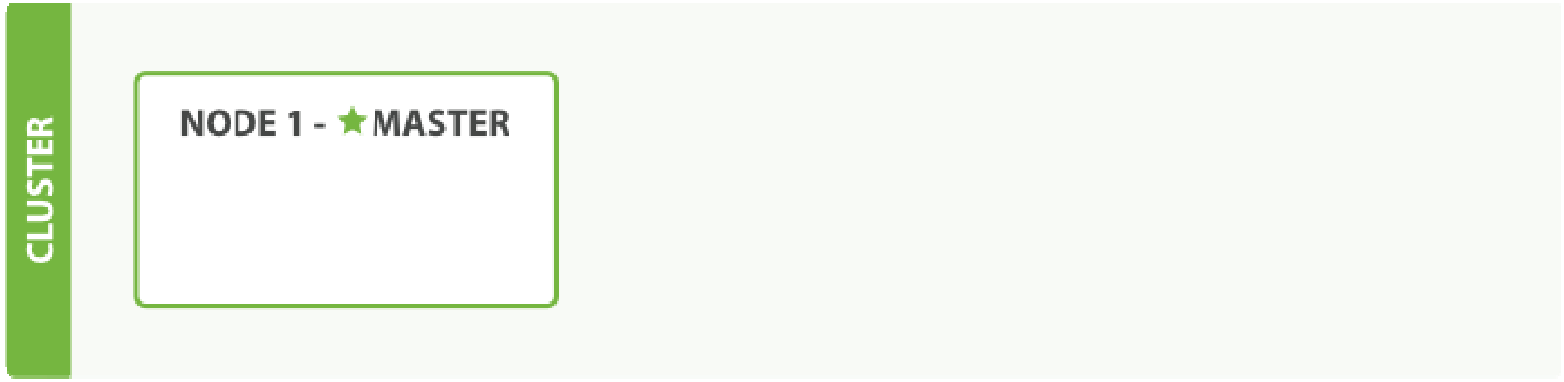
ES for relational database users...

JavaCro

Oracle	Elasticsearch
Database	Index
Partition	Shard
Table	Type
Row	Document
Column	Field
Schema	Mapping
Index	- (everything is indexed)
SQL	Query DSL

Clustering – single node cluster

JavaCro



- **Node** = running instance of ES
- **Cluster** = 1+ nodes with the same cluster.name
- Every cluster has **1 master node**
- **Clients talk to any node in the cluster**
- 1 Cluster can have any number of indexes

PRIVREDNA BANKA ZAGREB

About indexes & shards

JavaCro

Index

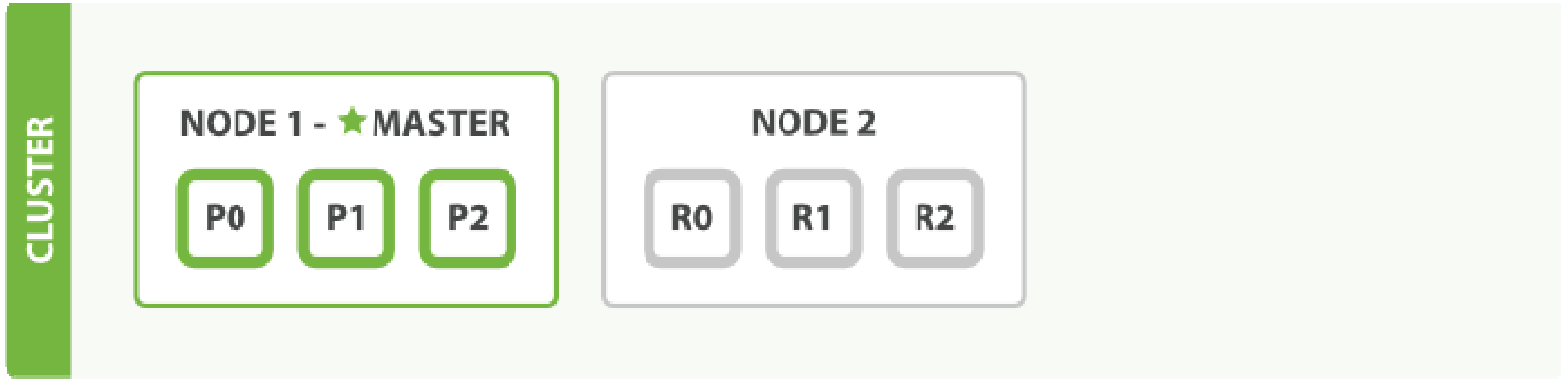
- All data is stored inside one or more indexes
- Index has one or more shards (change requires reindexing)
- One index is one folder somewhere on disk
- Backup an index? Just tar/zip the folder....

Shard

- Each shard is one full instance of Lucene
- Each shard can have zero or more replicas (can be changed at any time)

Clustering – adding a second node

JavaCro

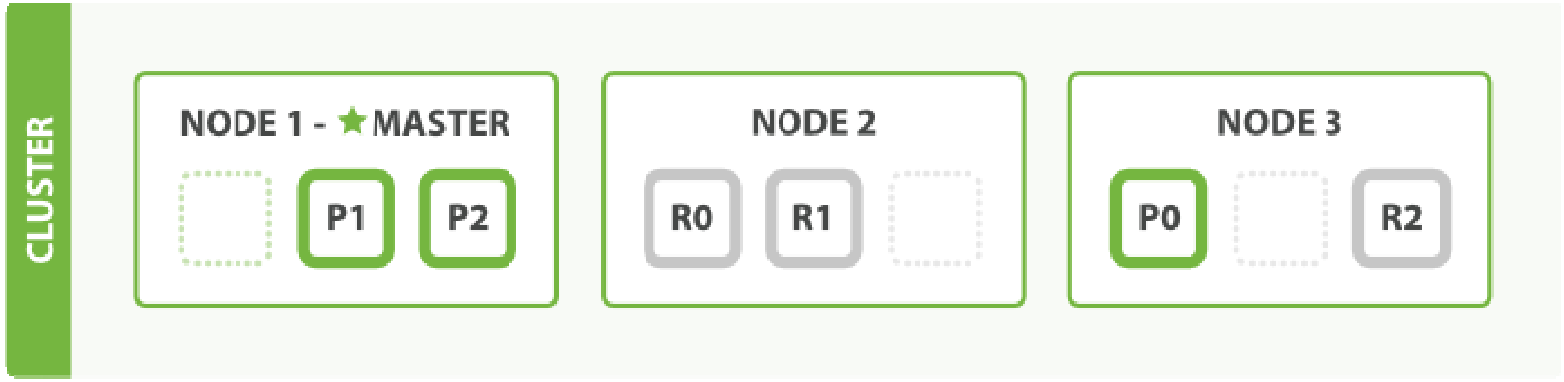


- Example above:
  - 3 indexes
  - Each index has one primary (P) and one replica (R) shard

PRIVREDNA BANKA ZAGREB

Clustering – adding a third node

JavaCro



- More primary shards:
  - faster indexing
  - more scale
- More replicas:
  - faster searching
  - more failover

PRIVREDNA BANKA ZAGREB

About documents...

JavaCro

- Documents are JSON-based
- Schema-free, but not necessarily!
- If no schema:
  - ES guesses field type
  - and indexes it
- With schema (or explicit mapping):
  - Mapping applies to specific document type (type is just a label)
  - Mapping defines the following for each field:
    - kind (string, number, date...)
    - to index or not
    - to store data or not

About documents...

JavaCro

- Each document has an ID (auto-generated or manually assigned)
- You can force placement of a document into a specific shard – routing!
- Versioning is available – optimistic version control !

Index details

JavaCro

- inverted index

  Elasticsearch Server 1.0 (doc 1)
  Mastering Elasticsearch (doc 2)
  Apache Solr 4 Cookbook (doc 3)

Term	Count	Document
1.0	1	<1>
4	1	<3>
apache	1	<3>
cookbook	1	<3>
elasticsearch	2	<1>,<2>
mastering	1	<2>
server	1	<1>
solr	1	<3>

Indexing example

JavaCro

```
POST /blog/blog_comment?routing=1
{
  "user_id" : 1,
  "date" : "2015-04-01T13:12:12",
  "comment" : "What's so cool about Elasticsearch?"
}
```

```
GET /blog/_mapping
{
   "blog": {
      "mappings": {
         "blog_comment": {
            "properties": {
               "comment": {
                  "type": "string"
               },
               "date": {
                  "type": "date",
                  "format": "dateOptionalTime"
               },
               "user_id": {
                  "type": "long"
               }
            }
         }
      }
   }
}
```

```
GET /blog/_search
{
   "took": 6,
   "timed_out": false,
   "_shards": {
      "total": 2,
      "successful": 2,
      "failed": 0
   },
   "hits": {
      "total": 1,
      "max_score": 1,
      "hits": [
         {
            "_index": "blog",
            "_type": "blog_comment",
            "_id": "AUzhH9M9HW_GzrF8oLAj",
            "_score": 1,
            "_source": {
               "user_id": 1,
               "date": "2015-04-01T13:12:12",
               "comment": "What's so cool about Elasticsearch?"
            }
         }
      ]
   }
}
```

Storing data - indexing

JavaCro

- data input: REST, Java API, Rivers*
- data analysis: tokenizer and one or more filters
- types of filters:
  - lowercase filter – makes all tokens lowercased
  - synonyms filter – changes one token to another on the basis of synonym rules
  - language stemming filters - reducing tokens into root or base forms, the stem
- different data storing needs
  - string analyze,not_analyze field configuration
  - _all in field
  - memory field data or doc values
- segments, segment merging, throttling
- routing, indexing with routing

So, we can store documents

and then what?!?

JavaCro

We query them!

- All the usual stuff (think of WHERE in SQL)
- Full text search with support for:
  - highlighting
  - stemming
  - ngrams & edge-ngrams
- Aggregations: term facets, date histograms, ranges
- Geo search: bounding box, distance,distance ranges, polygons
- Percolators (or reverse-search!)

Query details

JavaCro

- search types (query_then_fetch, query_and_fetch ...)
- same type of analysis as indexing
- explain plan
- sorting,aggregating data with in memory or on disk values
- search filters
  - Boolean
  - And/Or/Not
- filter cache, BitSets
- routing, searching with routing

PBZ use case

JavaCro

- turnovers by account: 600M documents, 200M/year
- routing by account number

- indexing performance, 30k-40k documents per second
- DB performance in seconds, ES performance in ms (3500 queries/sec):
  - find last 100 turnovers for a given account number: < 50 ms
  - find last 100 turnovers for a given account number where description contains some words: <100ms

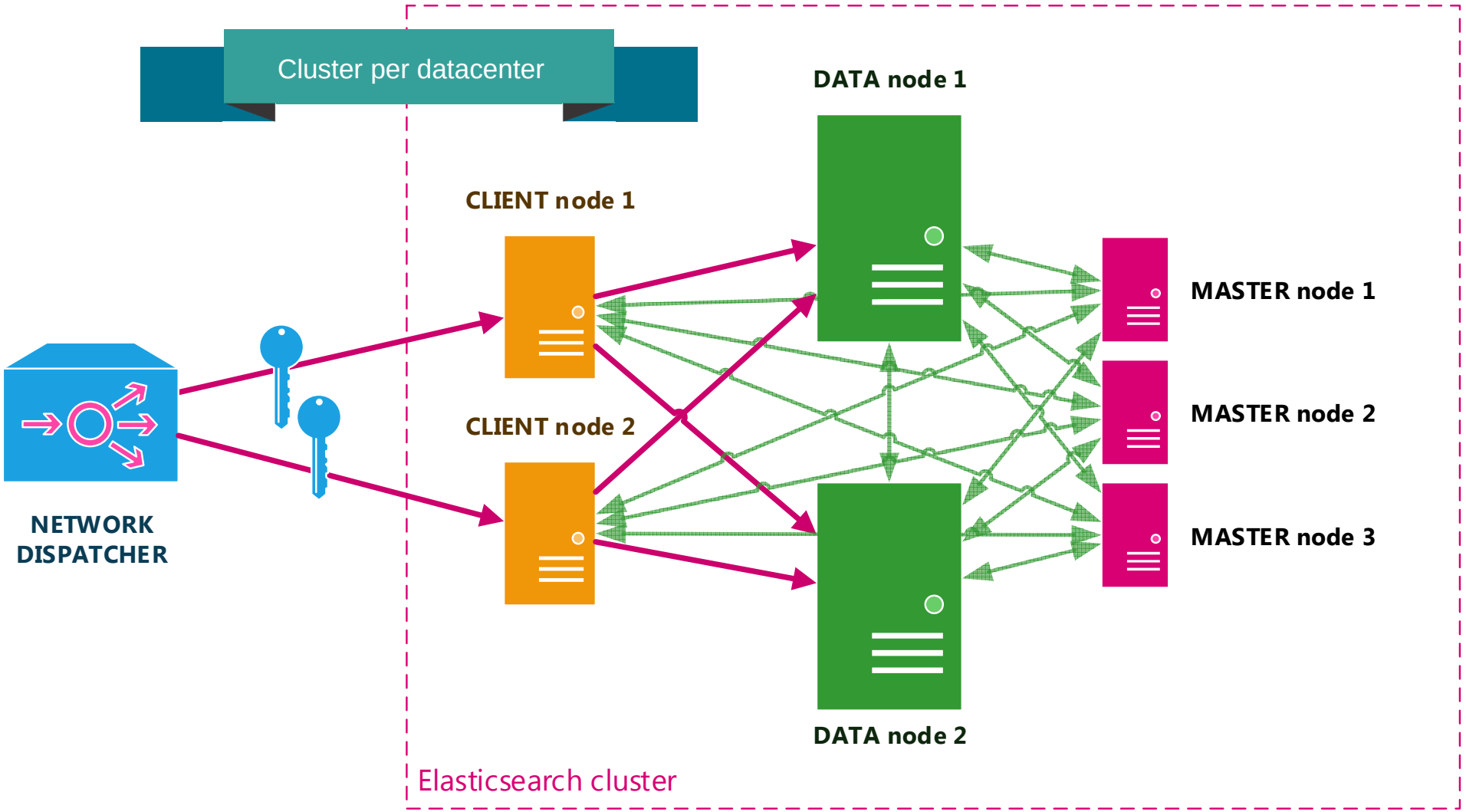
PRIVREDNA BANKA ZAGREB

JavaCro

Cluster architecture

PBZ ES cluster architecture

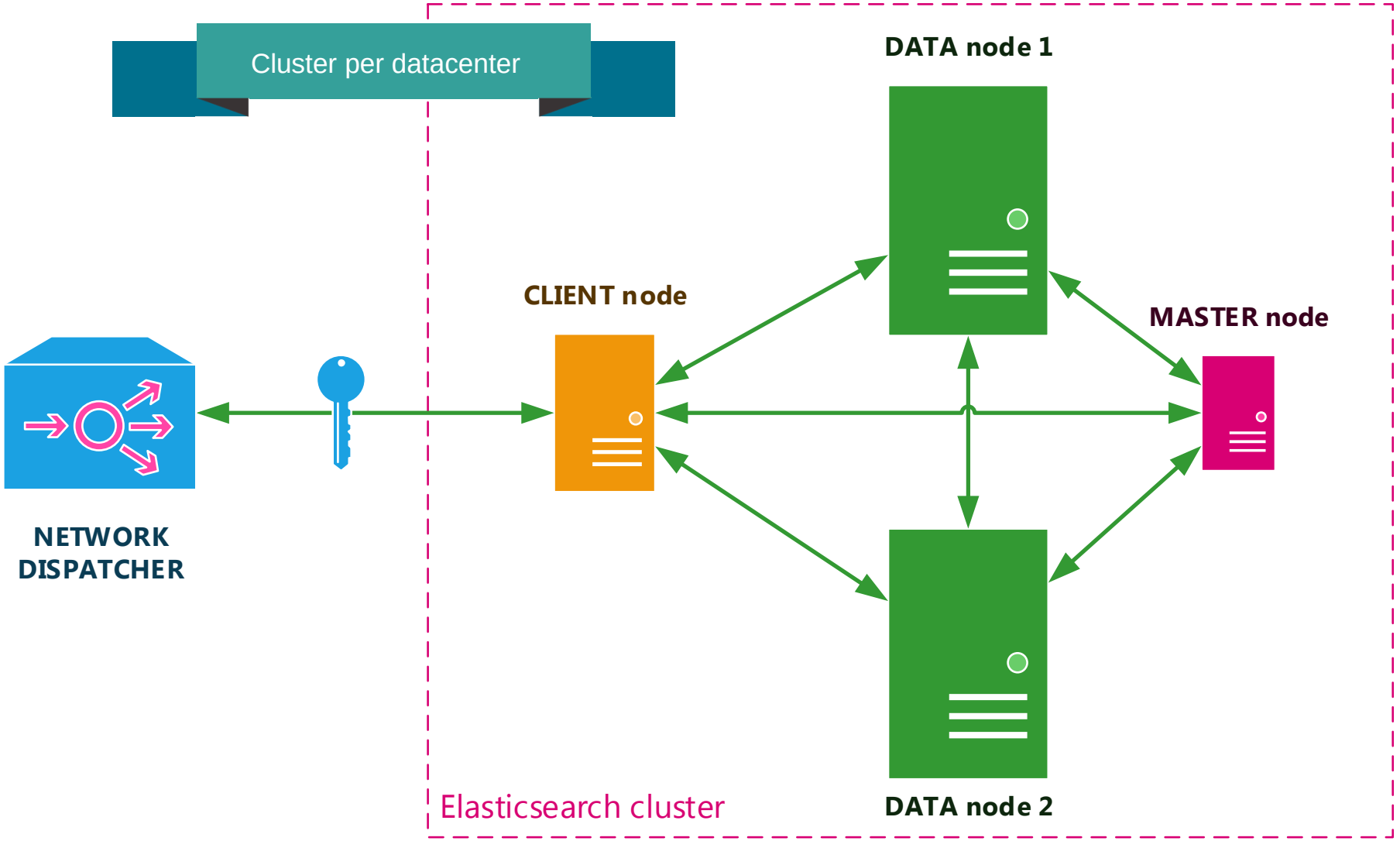
JavaCro



PRIVREDNA BANKA ZAGREB

PBZ ES cluster architecture

JavaCro



PRIVREDNA BANKA ZAGREB

Elasticsearch Administration

JavaCro

- plugins
  - Marvel – monitoring console (GC, throttiling, CPU, memory, heap, search/indexing statistics ...)
  - Sense – REST UI to Elasticsearch
  - custom plugins (JDBC rivers ...)
- security
  - Apache Web server
  - Elasticsearch Shield
- speeding up queries using warmers

JavaCro

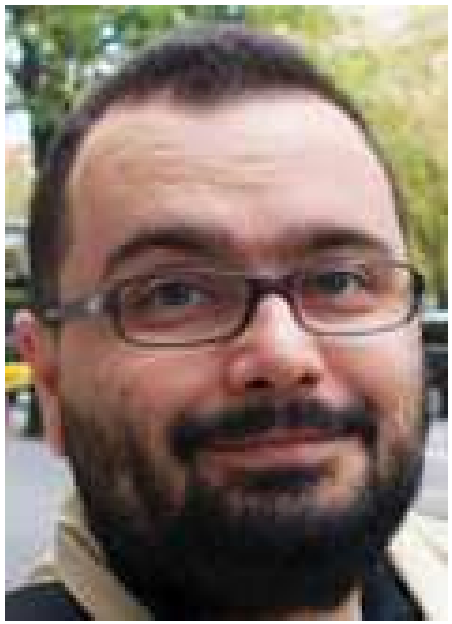
ELK

JavaCro

Q & A

Mladen Maravić & Kristijan Duvnjak

JavaCro



PRIVREDNA BANKA ZAGREB