

# Milestone: eXoa: Example-oriented API

Sanghoon Lee  
POSTECH

with  
Jinhan Kim, Seung-won Hwang, Sunghun Kim

July 10, 2009

# Contents

- Introduction
- eXoa
  - Overview
  - Using Source code repositories
  - Summarization
  - Diversification
  - Ranking
  - Embedding
- Evaluation
- Case Study
- Improvements

**List.Add 메서드(System.Collections.Generic) - Mozilla Firefox**

파일(F) 편집(E) 보기(V) 바로 가기(S) 북마크(B) 도구(I) 도움말(H)

msdn

[http://msdn.microsoft.com/ko-kr/library/3wcytfid1\(VS.85\).aspx](http://msdn.microsoft.com/ko-kr/library/3wcytfid1(VS.85).aspx)

**.NET Framework Developer Center**

홈 라이브러리 학습 다운로드

인쇄용 버전 전송 혼란스 추가...

MSDN > MSDN Library > .NET 개발 > 미진 버전 > .NET Framework

Add 메서드

제대역록 보기 설정

모두 측소 언어 필터 : C++

## List.Add 메서드

개체를 List의 끝 부분에 추가합니다.

네임스페이스: System.Collections.Generic  
이름별칭: mscorlib(mscorlib.dll)

구문

C++

```
using namespace System;
using namespace System::Collections::Generic;

void main()
{
    List<String^>^ dinosaurs = gcnew List<String>();

    Console::WriteLine("\nCapacity: {0}", dinosaurs->Capacity);

    dinosaurs->Add("Tyrannosaurus");
    dinosaurs->Add("Amargasaurus");
    dinosaurs->Add("Mamenchisaurus");
    dinosaurs->Add("Deinonychus");
    dinosaurs->Add("Compsognathus");

    Console::WriteLine();
    for each(String^ dinosaur in dinosaurs)
    {
        Console::WriteLine(dinosaur);
    }

    Console::WriteLine("\nCapacity: {0}", dinosaurs->Capacity);
    Console::WriteLine("Count: {0}", dinosaurs->Count);

    Console::WriteLine("\nContains(\"Deinonychus\"): {0}",
        dinosaurs->Contains("Deinonychus"));

    Console::WriteLine("\nInsert(2, \"Compsognathus\")");
    dinosaurs->Insert(2, "Compsognathus");

    Console::WriteLine();
}
```

완료

# Javadoc

No examples !!!

The image displays two side-by-side screenshots of Mozilla Firefox browser windows, both showing the Java API documentation for the `ArrayList` class.

**Left Window (Class Overview):**

- Page Title: `ArrayList (Java Platform SE 6) - Mozilla Firefox`
- Address Bar: `http://java.sun.com/javase/6/docs/api/java/util/ArrayList.html`
- Toolbar Buttons: Back, Forward, Stop, Refresh, Home, Favorites, Search, and ABP (Ad Blocker).
- Header: `Overview Package Class Use Tree Deprecated Index Help`, `PREV CLASS NEXT CLASS`, `SUMMARY: NESTED | FIELD | CONSTR | METHOD`.
- Header: `Java™ Platform Standard Ed. 6`, `FRAMES NO FRAMES ALL CLASSES`, `DETAIL FIELD | CONSTR | METHOD`.
- Content:
  - java.util**
  - Class `ArrayList<E>`**
  - `java.lang.Object`
    - `<--> java.util.AbstractCollection<E>`
    - `<--> java.util.AbstractList<E>`
    - `<--> java.util.ArrayList<E>`
  - All Implemented Interfaces:**
    - `Serializable, Cloneable, Iterable<E>, Collection<E>`
  - Direct Known Subclasses:**
    - `AttributeList, RoleList, RoleUnresolvedList`
  - public class `ArrayList<E>`**  
extends `AbstractList<E>`  
implements `List<E>, RandomAccess, Cloneable, Serializable`
  - Resizable-array implementation of the List interface. Imp...  
addition to implementing the List interface, this class pro...  
list. (This class is roughly equivalent to Vector, except th...
  - The size, isEmpty, get, set, iterator, and listIterator ...  
that is, adding n elements requires O(n) time. All of the o...  
compared to that for the LinkedList implementation.
  - Each ArrayList instance has a *capacity*. The capacity is ...  
large as the list size. As elements are added to an ArrayList, ...  
specified beyond the fact that adding an element has cost...
  - An application can increase the capacity of an ArrayList ...  
operation. This may reduce the amount of incremental real...
  - Note that this implementation is not synchronized.**  
the threads modifies the list structurally, it *must* be synch...  
one or more elements, or explicitly resizes the backing ar...

**Right Window (Method Documentation):**

- Page Title: `ArrayList (Java Platform SE 6) - Mozilla Firefox`
- Address Bar: `http://java.sun.com/javase/6/docs/api/java/util/ArrayList.html`
- Toolbar Buttons: Back, Forward, Stop, Refresh, Home, Favorites, Search, and ABP (Ad Blocker).
- Header: `Overview Package Class Use Tree Deprecated Index Help`, `PREV CLASS NEXT CLASS`, `SUMMARY: NESTED | FIELD | CONSTR | METHOD`.
- Header: `Java™ Platform Standard Ed. 6`, `FRAMES NO FRAMES ALL CLASSES`, `DETAIL FIELD | CONSTR | METHOD`.
- Content:
  - add**
  - `public boolean add(E e)`
  - Appends the specified element to the end of this list.
  - Specified by:** `add in interface Collection<E>`
  - Overrides:** `add in interface List<E>`
  - Parameters:** `e` – element to be appended to this list
  - Returns:** `true` (as specified by `Collection.add(E)`)
  - add**
  - `public void add(int index, E element)`
  - Inserts the specified element at the specified position in this list. Shifts the element currently at that position (if any) and any subsequent elements to the right (adds one to their indices).
  - Specified by:** `add in interface List<E>`
  - Overrides:** `add in class AbstractList<E>`
  - Parameters:** `index` – index at which the specified element is to be inserted  
`element` – element to be inserted
  - Throws:** `IndexOutOfBoundsException` – if the index is out of range (`index < 0 || index > size()`)
  - remove**
  - `public E remove(int index)`

# Javadoc + Examples = Example Oriented API documentation

## Javadoc

add

```
public boolean add(E e)
```

Appends the specified element to the end of this list.

Specified by:

[add](#) in interface [Collection<E>](#)

Specified by:

[add](#) in interface [List<E>](#)

Overrides:

[add](#) in class [AbstractList<E>](#)

Parameters:

e – element to be appended to this list

Returns:

true (as specified by [Collection.add\(E\)](#))

add

```
public void add(int index,  
                E element)
```

Inserts the specified element at the specified position in this list. Shifts the elements to the right (adds one to their indices).

Specified by:

[add](#) in interface [List<E>](#)

Overrides:

[add](#) in class [AbstractList<E>](#)

Parameters:

index – index at which the specified element is to be inserted

element – element to be inserted

Throws:

## Examples

```
String name;  
ArrayList acc = new ArrayList(headers.size());  
for (Iterator i = headers.iterator(); i.hasNext();) {  
    InternetHeader header = (InternetHeader) i.next();  
    if (header.equals(name) && header.line != null) {  
        acc.add(header.getValue());  
    }  
}
```

# Good Examples

- Simplicity

```
public String[] getHeader(String name) {  
    ArrayList acc = new ArrayList(headers.size());  
    for (Iterator i = headers.iterator(); i.hasNext();) {  
        InternetHeader header = (InternetHeader) i.next();  
        if (header.equals(name) && header.line != null) {  
            acc.add(header.getValue());  
        }  
    }  
    int size = acc.size();  
    if (size == 0) {  
        return null;  
    }  
    String[] h = new String[size];  
    acc.toArray(h);  
    return h;  
}
```



```
String name;  
ArrayList acc = new ArrayList(headers.size());  
for (Iterator i = headers.iterator(); i.hasNext();) {  
    InternetHeader header = (InternetHeader) i.next();  
    if (header.equals(name) && header.line != null) {  
        acc.add(header.getValue());  
    }  
}
```

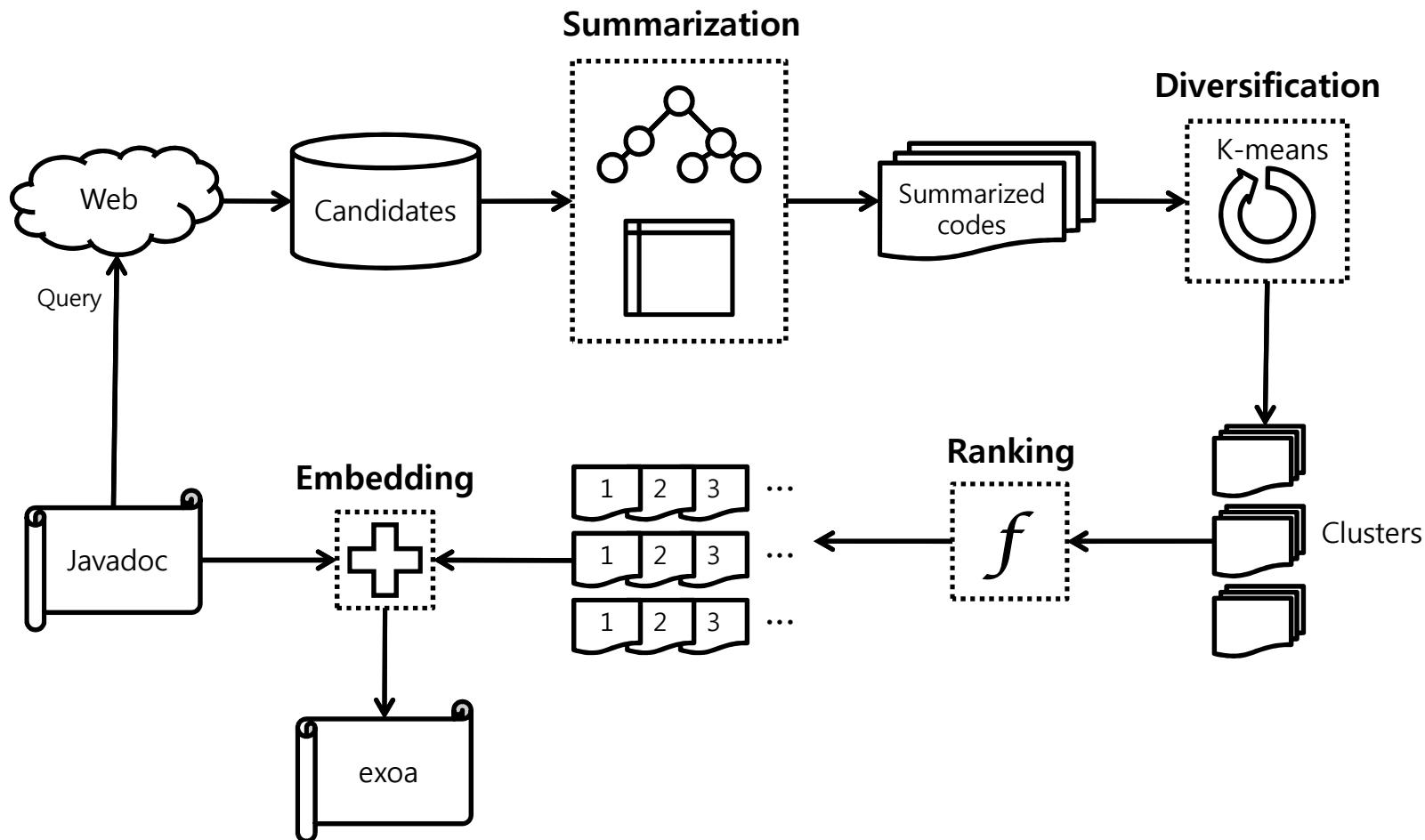
- Various Usages

```
ArrayList list = new ArrayList();  
list.add("one");  
list.add("two");  
list.add("three");
```

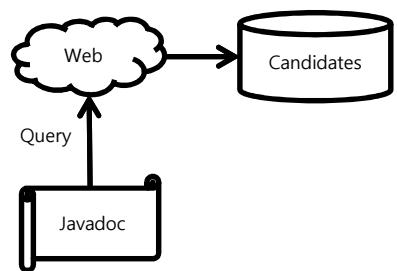


```
ArrayList list = new ArrayList();  
for (int i = 0; i < original.length; i++) {  
    if (!original[i].equals(element)) {  
        list.add(original[i]);  
    }  
}
```

# Overview of eXoa



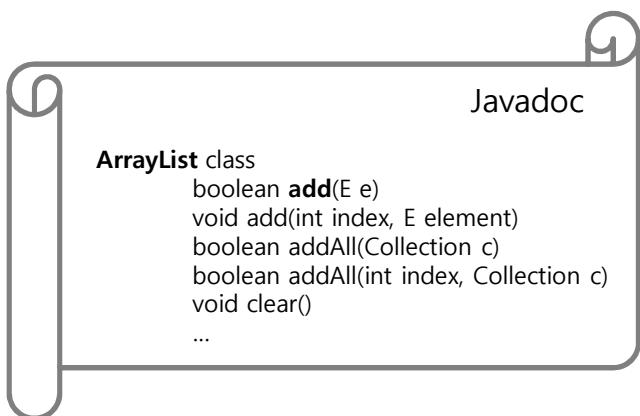
# Using Source code repositories



powered by blackducksoftware™

Searching 2,117,339,531 lines of open source code.

A screenshot of a web-based search interface. At the top, there is a search bar containing the text "ArrayList add". Below the search bar are two dropdown menus: "Java" and "All Licenses". To the right of these dropdowns is a blue "Search" button. The background of the interface is dark blue.



## SortedList.java

```
}
```

```
 /**
 * Adds the specified object in proper location to keep list sorted.
 * @param elementToAdd the element to be added
 */
```

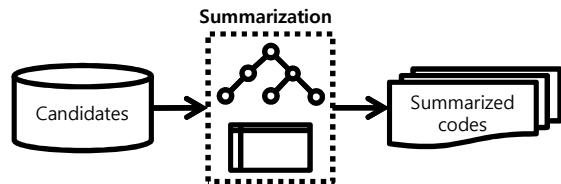
## SSLSocket.java

```
import java.io.PrintStream;
import java.math.BigInteger;
import java.net.InetAddress;
import java.net.Socket;
import java.net.SocketAddress;
```

## TopologicalSorting.java

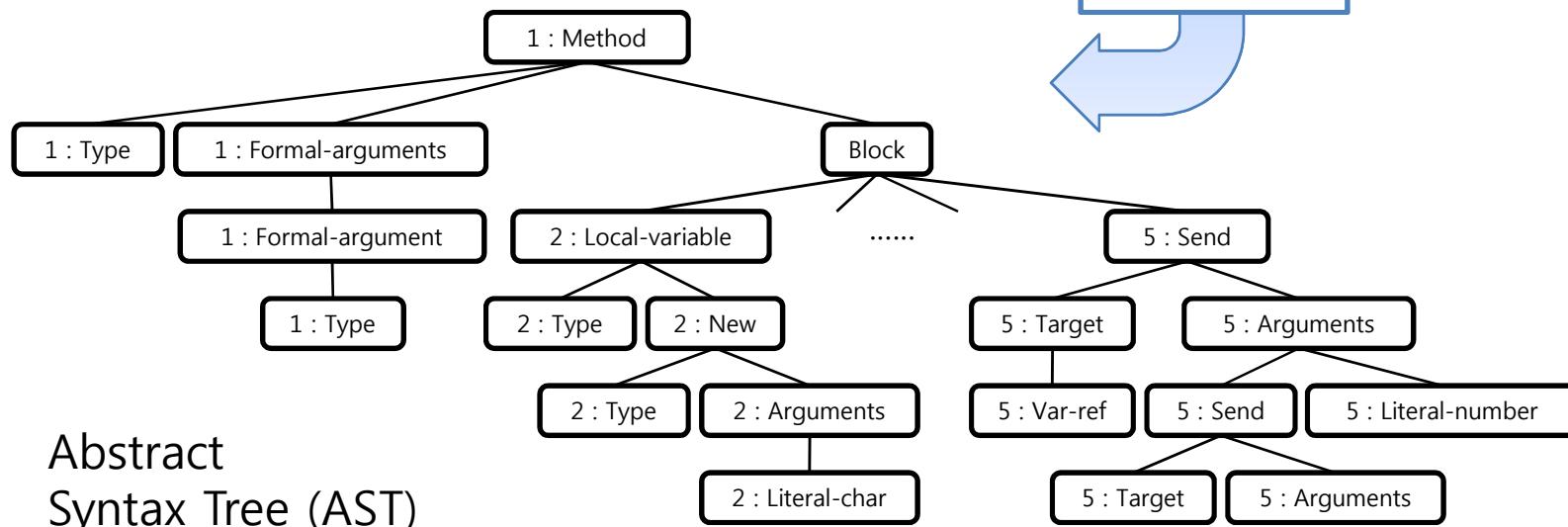
```
}
// We're guaranteed a single root by the above construction.
List rootList = new ArrayList();
rootList.add(startat);
// Finally, perform the topological sort.
```

# Summarization



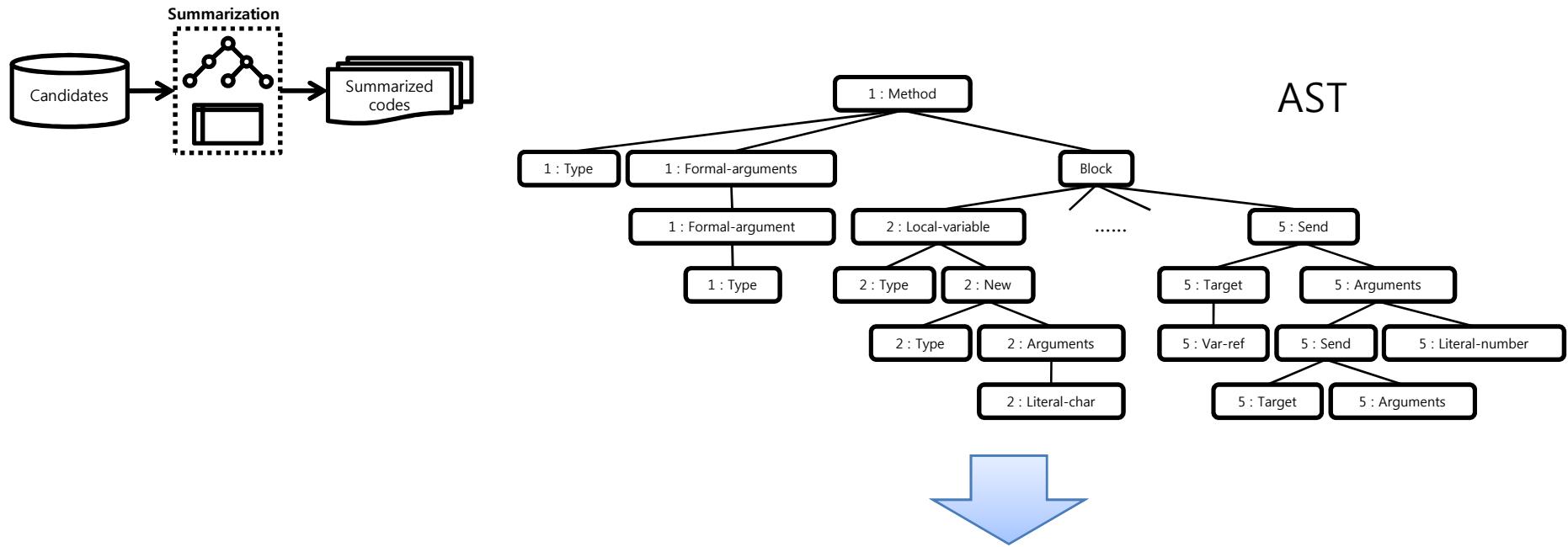
Method-based sliced code

```
public void test(TestHarness harness) {  
    Character a = new Character('\\uffda');  
    Character b = new Character('Z');  
    harness.check(a.hashCode(), 65498);  
    harness.check(b.hashCode(), 90);  
}
```



Java2XML

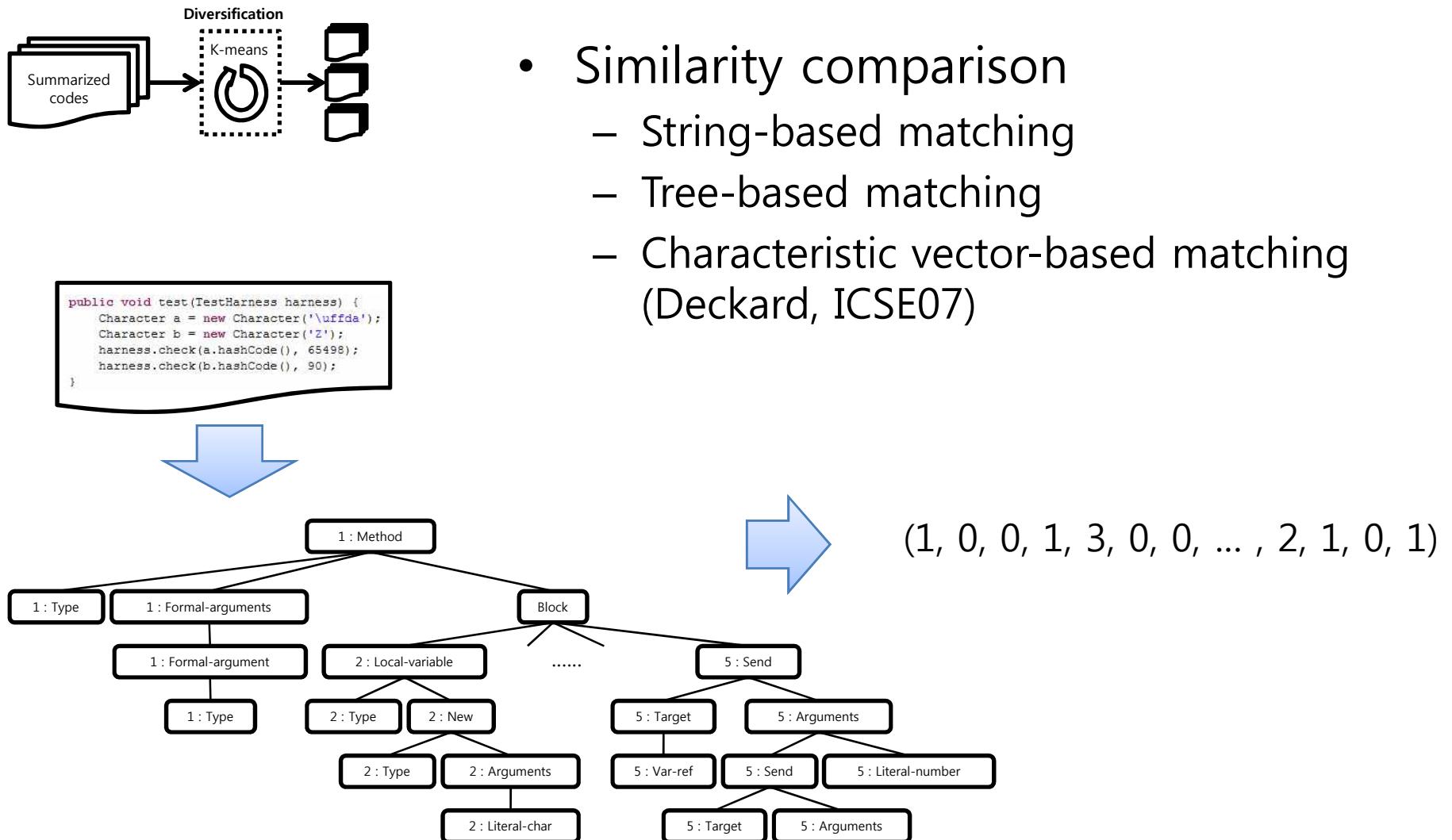
# Summarization



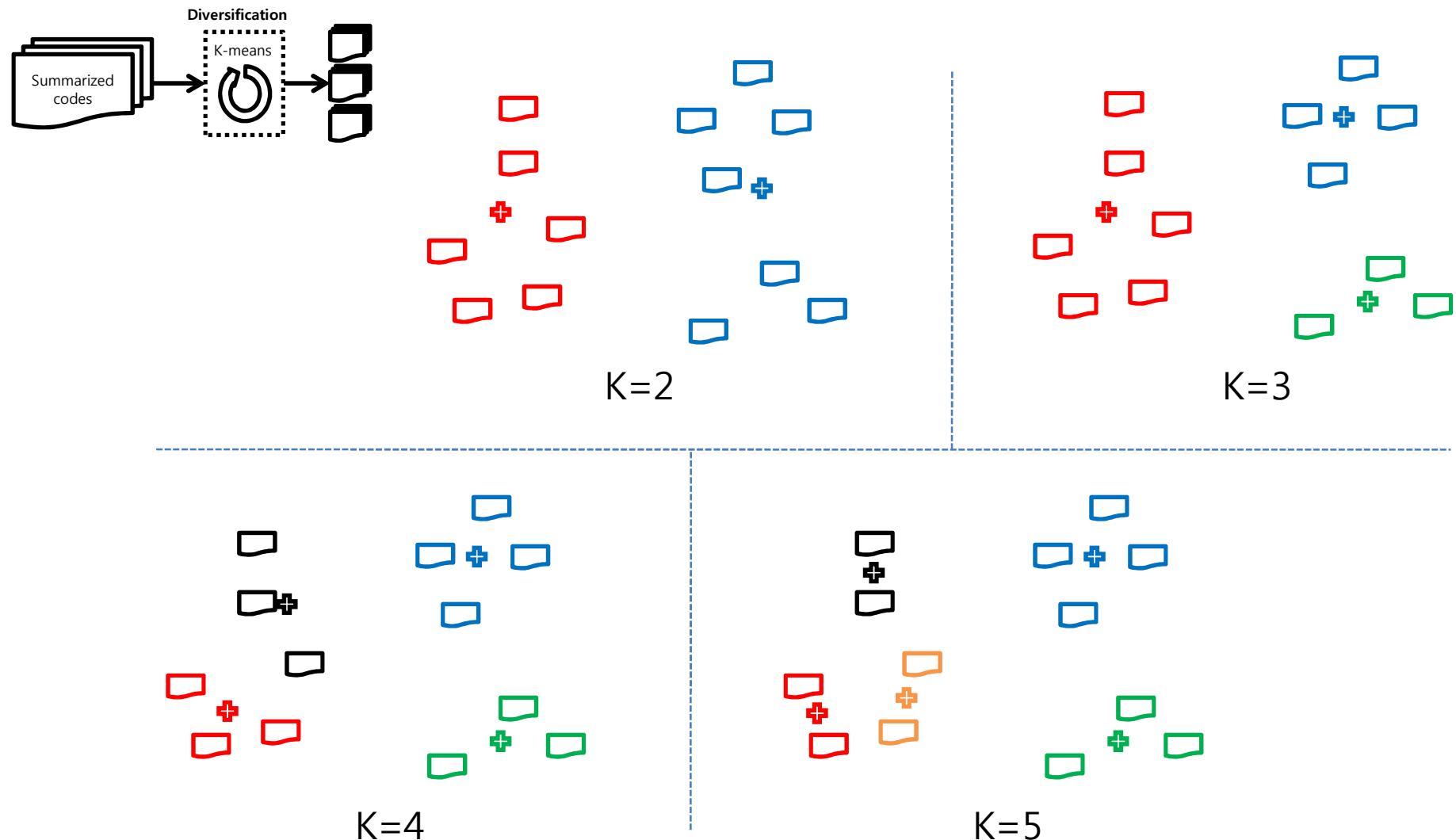
Intra  
Analysis  
Table  
(IAT)

Line	Class	API name	Arguments
4	a : Character : 2	hashCode	
5	b : Character : 3	hashCode	
4	harness : TestHarness : 1	check	int, int
5	harness : TestHarness : 1	check	int, int
...	...	...	...

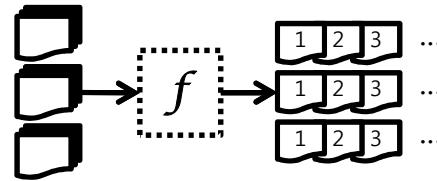
# Diversification



# Diversification

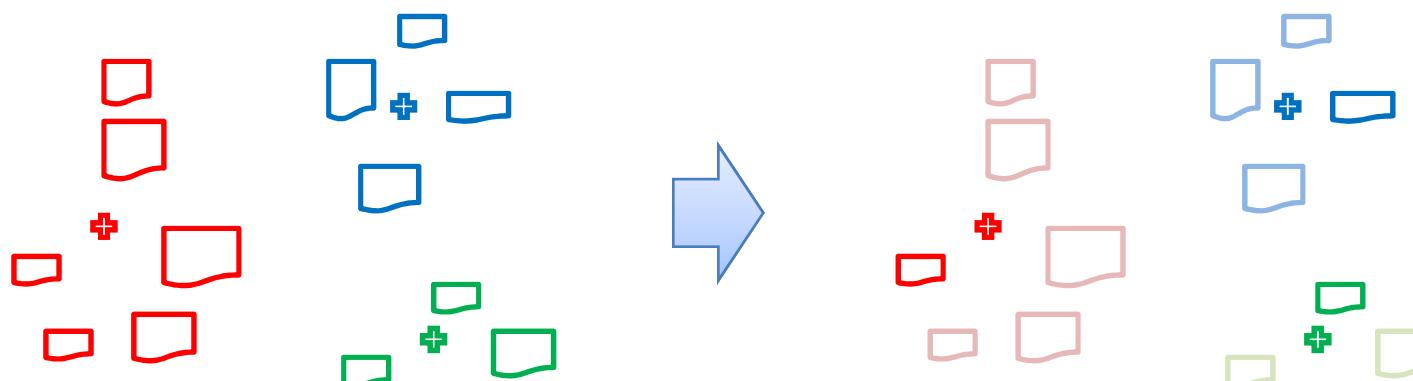


# Ranking

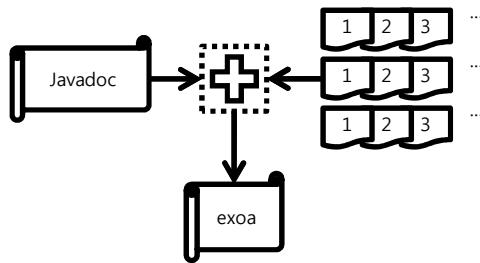


- The most representative example in a cluster
- Measures
  - Distance from centroid
  - Conciseness
  - Correctness
  - + User Feedback

K=3



# Embedding



(a) Popularity

**clone**  
Creates a copy of this object.  
Creates  
Creates an object to produce a string.  
**format**  
Formats an object and appends the resulting text to a given string buffer.  
Format  
Formats an object producing an ASCII character representation.  
**parseObject**  
Parses text from the beginning of the given string to produce an object.  
**parseObject**  
Parses text from a string to produce an object.

Method Summary

Creates  
Creates an object to produce a string.  
Format  
Formats an object producing an ASCII character representation.  
parseObject  
Parses text from the beginning of the given string to produce an object.  
parseObject  
Parses text from a string to produce an object.

Methods inherited from class `java.lang.Object`

`equals`, `finalize`, `getClass`,  `hashCode`, `notify`, `notifyAll`, `wait`, `wait`

Method Detail

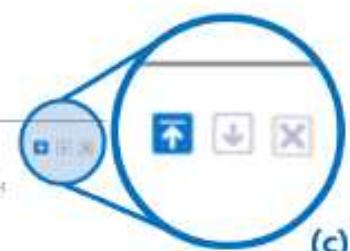
**format**

```
public final String format(Object obj)
    Formats an object to produce a string. This is equivalent to
    format(obj, new StringBuffer(), new FieldPosition(0)).toString()
Parameters:
    obj - The object to format.
Returns:
    Formatted string.
Throws:
    IllegalArgumentException - If the Format cannot format the given object
Examples: How examples...
```

(b) Code example

**format(parsed)**

```
public Object parseObject(String str, Format format) {
    ...
    Object parsed = FORMAT.parseObject(str);
    ...
    return format.format(parsed).equals(str);
}
```



(c) User Feedback

# Evaluation

- Comparison with Koders
  - Accuracy

	Relevant code and snippet	Relevant code	Unrelated code
Koders	22%	8%	70%
eXoaDocs	92%	6%	2%

Table I  
TOP-5 QUERY RESULTS COMPARISON BETWEEN KODERS AND  
**eXoaDocs** USING RANDOMLY SELECTED 100 APIs

- Diversification
  - Koders : 30.8% are duplicated (among 22%)
  - eXoa : only 8.7% are duplicated (among 92%)

# Case Study

- Productivity

Group	Task <sub>1</sub>	Task <sub>2</sub>	Task <sub>3</sub> & Task <sub>4</sub>
eXoaDocs	8:53	23:34	30:32
JavaDocs	14:40	25:03	32:03

Table II

AVERAGING THE CUMULATIVE COMPLETION TIME OF ONLY THOSE WHO COMPLETED THE TASK (MIN:SEC).

Group	Total lookups	Distinct lookups	Relevant lookups	$\frac{\text{relevant}}{\text{distinct}}$
eXoaDocs	5.67	3.25	2.33	0.72
JavaDocs	17.58	7.5	3.25	0.43

Table III

THE AVERAGE NUMBER OF DOCUMENT LOOKUPS

- Code quality

Group	Task <sub>1</sub>	Task <sub>2</sub>	Task <sub>3</sub>	Task <sub>4</sub>
eXoaDocs	11	11	3	3
JavaDocs	10	8	4	1

Table IV

THE NUMBER OF SUBJECTS PASSING THE TEST SUITE FOR EACH TASK

- Cognitive load

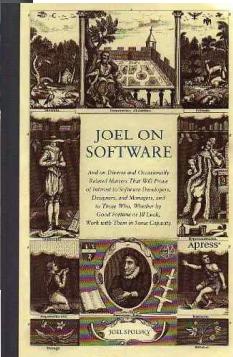
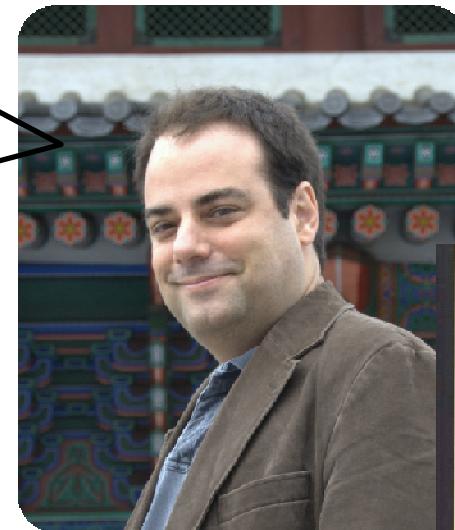
Group	Extremely useful	Very useful	Useful	Somewhat useful	Not useful
eXoaDocs	3	4	3	2	0
JavaDocs	1	2	5	2	2

Table V

POST-STUDY SURVEY RESULT

# Developer Feedback

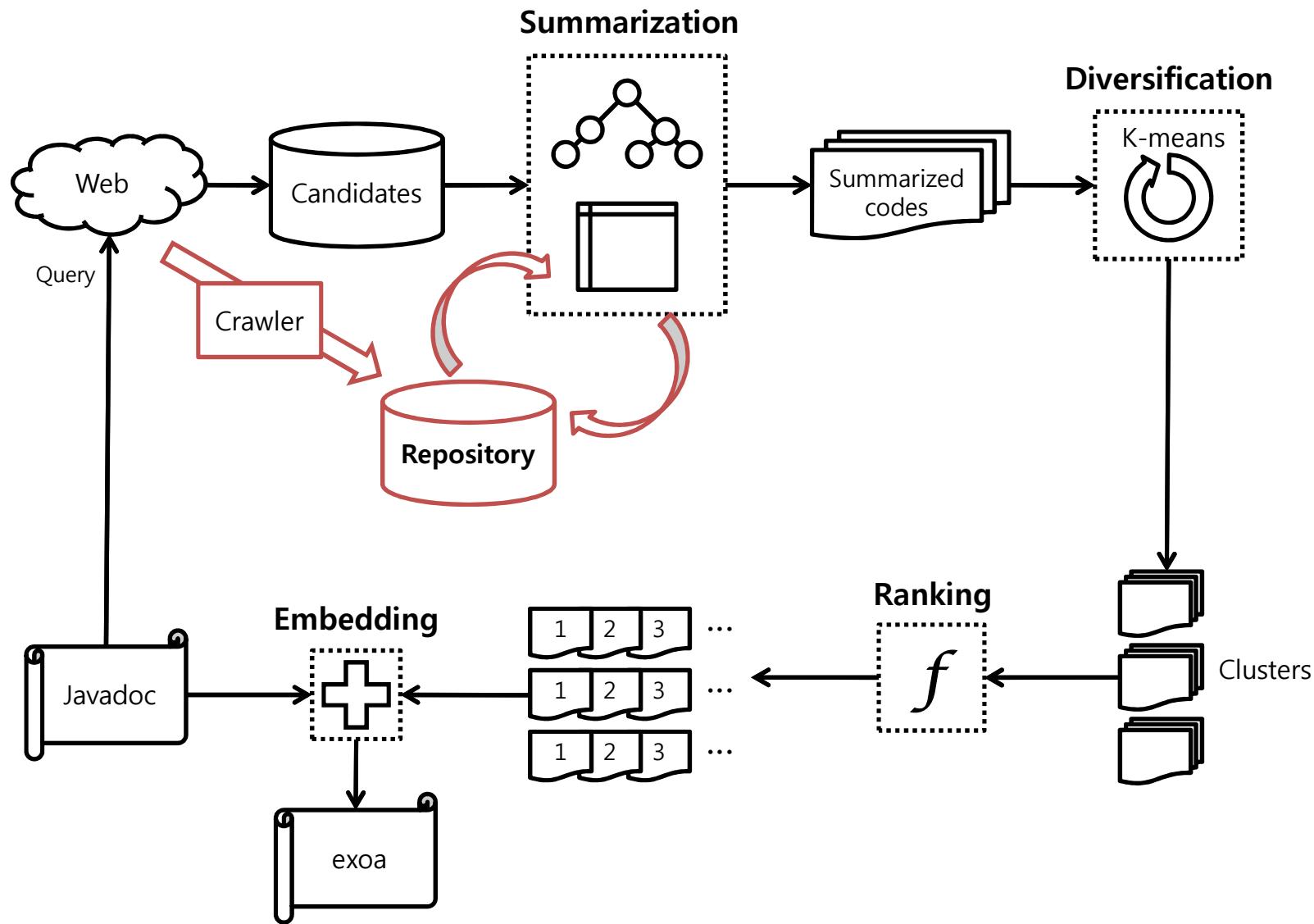
**Joel Spolsky:** "I think this is a **fantastic idea**. Just yesterday, I was facing this exact problem... the API documentation wasn't good enough, and I would have killed for a couple of examples. It sounds like a **very smart idea...**"



**Developer 2:** "Automatic example finding sounds really good. This would help developers significantly. In fact, **I was struggled many times to understand APIs without examples.**"

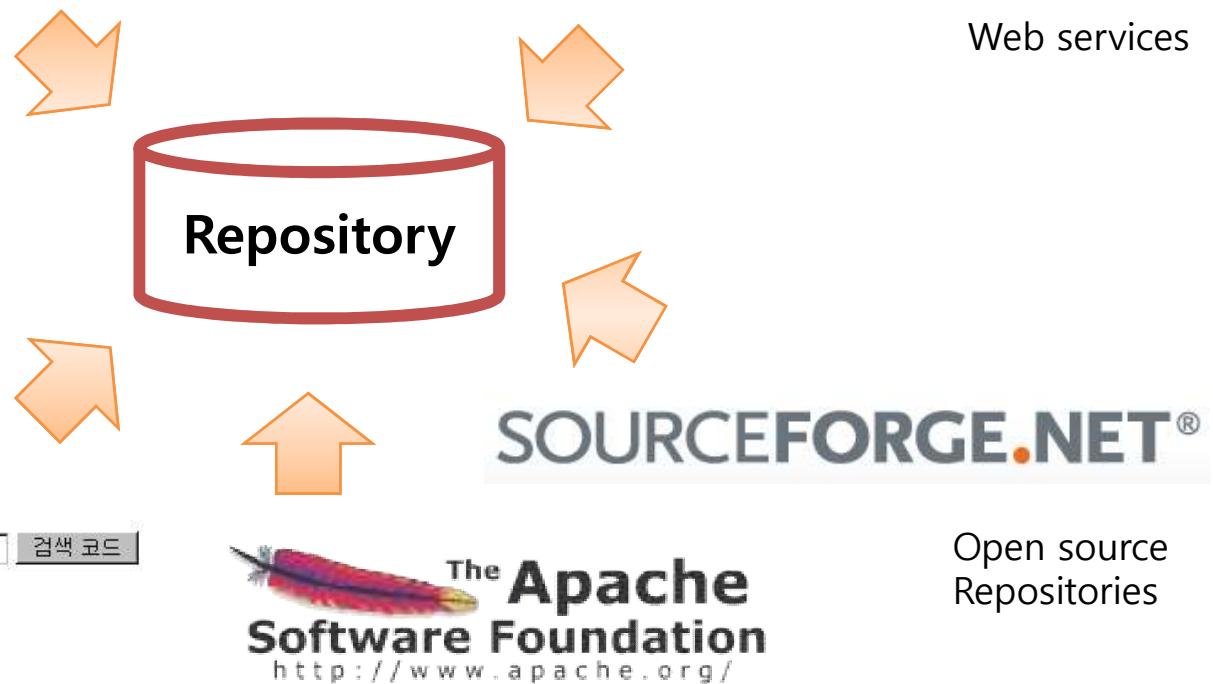
**Developer 3:** "API documents that have code example is really helpful... Using these code examples, **I can reduce develop time significantly**. However, JavaDocs provides very few code examples, so I need to find additional code examples. I like MSDN because most functions and methods contain code examples. For this reason, **automatically finding and adding code examples seems a wonderful idea...**"

# Improvements - 1



# Improvements - 2

<http://koders.com/default.aspx?s=ArrayList+add&btn=&la=Java>



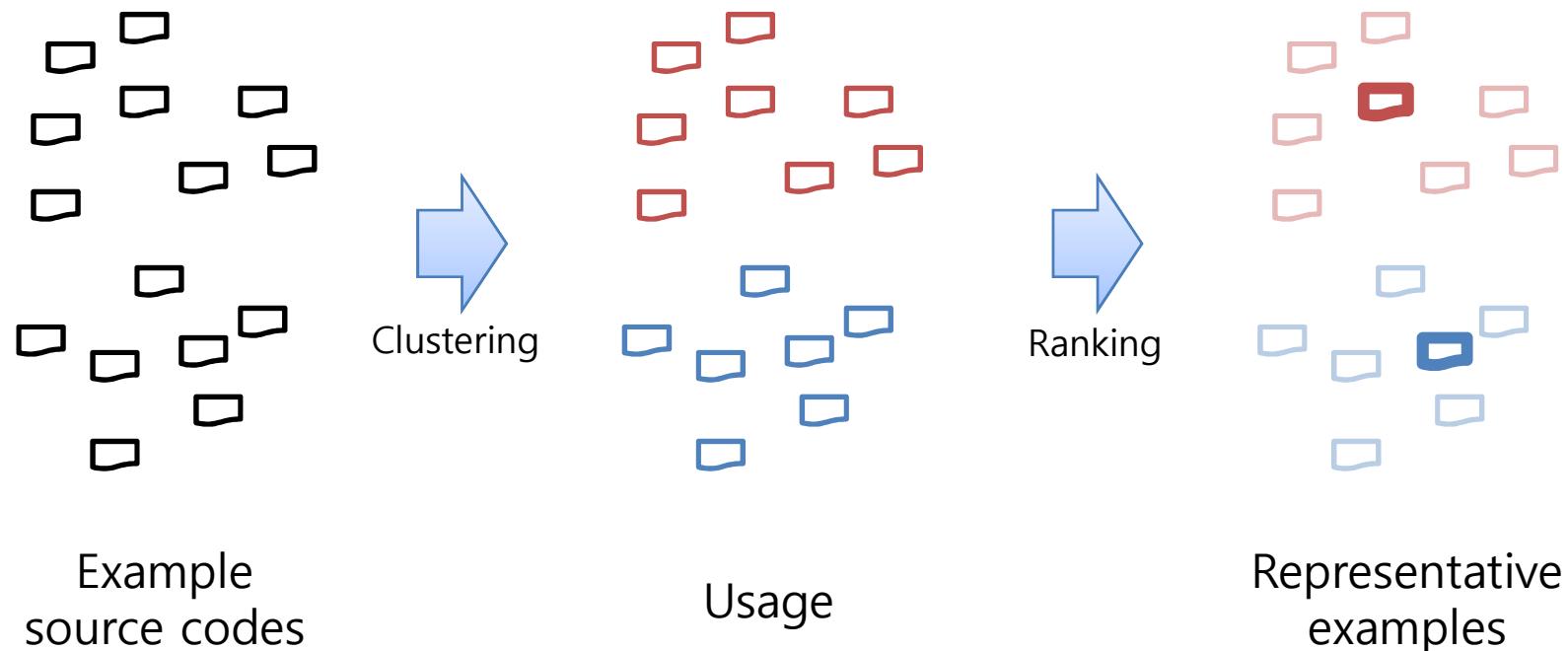
# Improvements - 3

- Simplify examples on a method call

```
public String[] getHeader(String name) {  
    ArrayList acc = new ArrayList(headers.size());  
    for (Iterator i = headers.iterator(); i.hasNext();) {  
        InternetHeader header = (InternetHeader) i.next();  
        if (header.equals(name) && header.line != null) {  
            acc.add(header.getValue());  
        }  
    }  
    int size = acc.size();  
    if (size == 0) {  
        return null;  
    }  
    String[] h = new String[size];  
    acc.toArray(h);  
    return h;  
}  
  
String name;  
ArrayList acc = new ArrayList(headers.size());  
for (Iterator i = headers.iterator(); i.hasNext();) {  
    InternetHeader header = (InternetHeader) i.next();  
    if (header.equals(name) && header.line != null) {  
        acc.add(header.getValue());  
    }  
}
```



# Improvements - 4



To be continued in 'Lightning talk'...

# **Q & A**

**Thank you for listening ...**