

# Improve computational efficiency with the latest programming languages.

Grzegorz Michalski, Andrzej Grosser  
 Institute of Computer and Information Sciences,  
 Faculty of Mechanical Engineering and Computer Science,  
 Czestochowa University of Technology, Czestochowa, Poland

## Introduction

### Most popular coding Languages of 2017

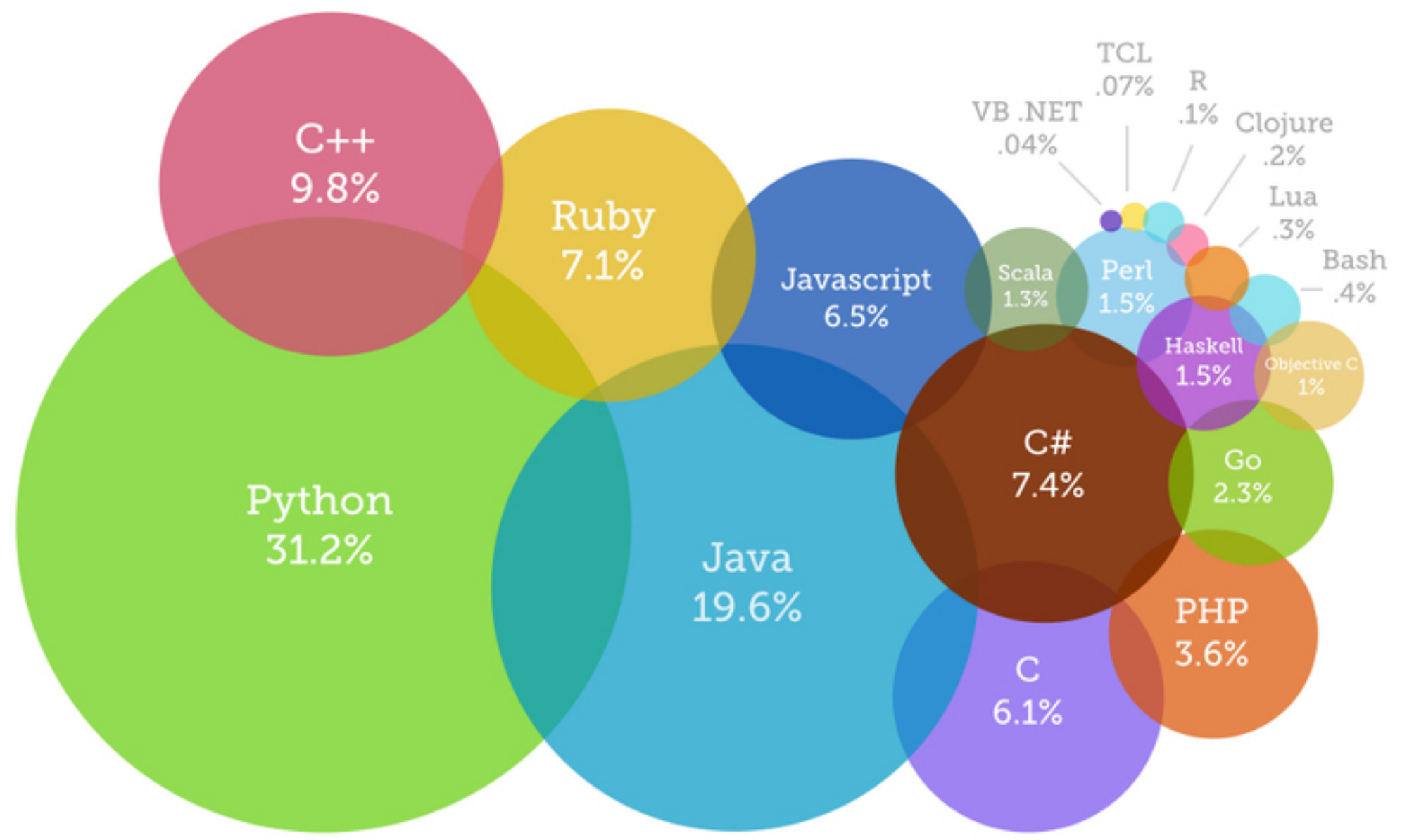


Figure : Language popularity in 2015

## Concurrency in C++ x17

Native C++ thread:

```
#include <iostream>
#include <thread>
using namespace std;
```

```
void func(int x) {
    cout << "Inside thread " << x << endl;
}
```

```
int main() {
    thread th(&func, 100);
    th.join();
    cout << "Outside thread" << endl;
    return 0;
}
```

Built-in mutex example:

```
int accum = 0;
mutex accum_mutex;
void square(int x) {
    int temp = x * x;
    accum_mutex.lock();
    accum += temp;
    accum_mutex.unlock();
}
```

## Model the build the system of equations – parrallel version

The values of elements in  $i$ -row of the coefficient matrix depend on the finite elements, which include the node connected with the node with index  $i$ . A single node in the finite element mesh can belong to several finite elements. No direct method exists to identify these finite elements solely on the basis of the node index. These indices can only be read from the finite element mesh. Fig. 2 shows the process of building the global coefficient matrix. Non-zero elements of the global coefficients matrix are determined as the sum of several values which depend on those finite elements which include the pair of nodes with indexes corresponding to the indexes of the row and column of those elements. Constructing the global matrix of coefficient in this way definitely makes the parallelization process more difficult.

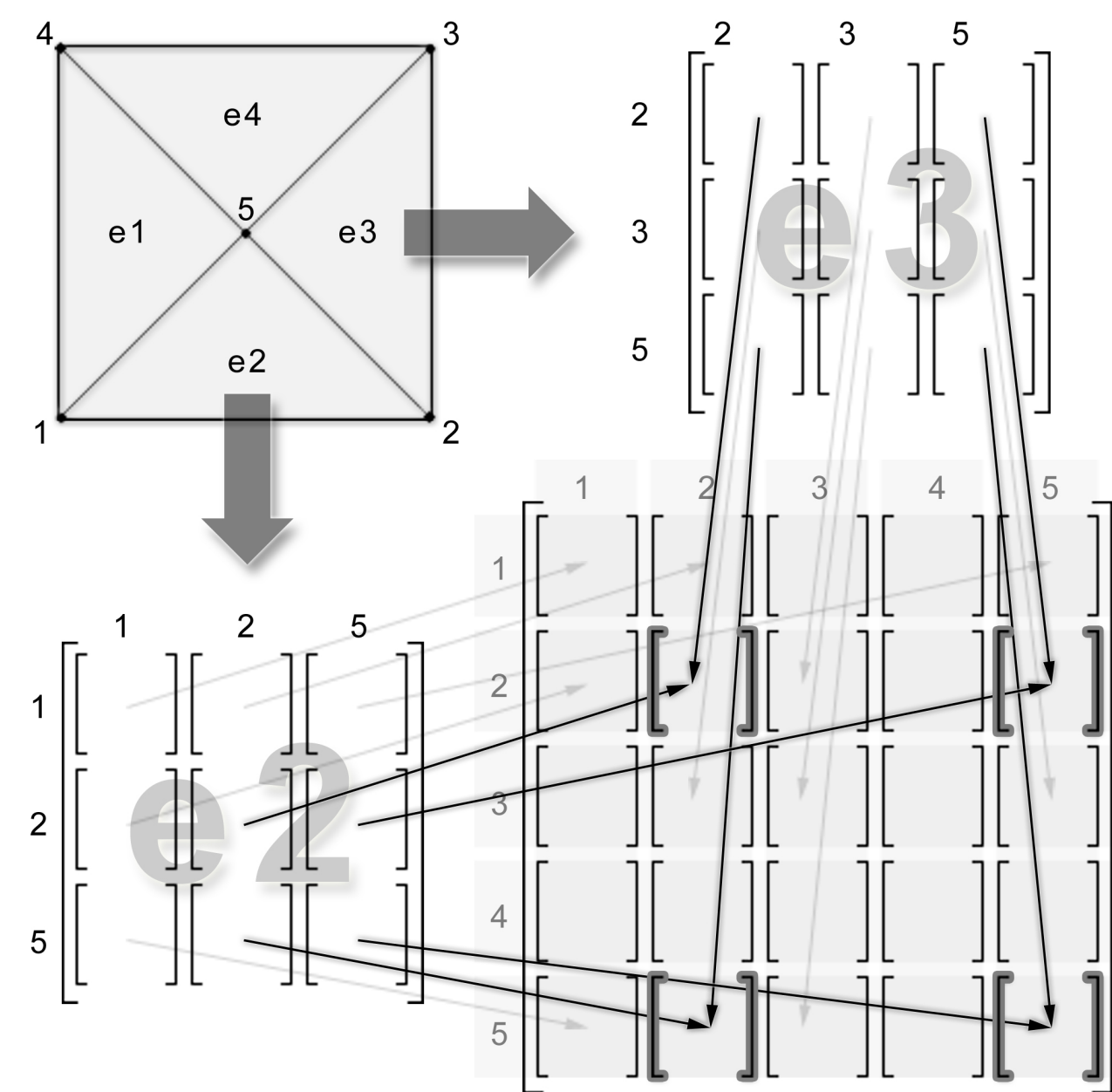


Figure : Process of building the global coefficient matrix

## CPP Language roadmap

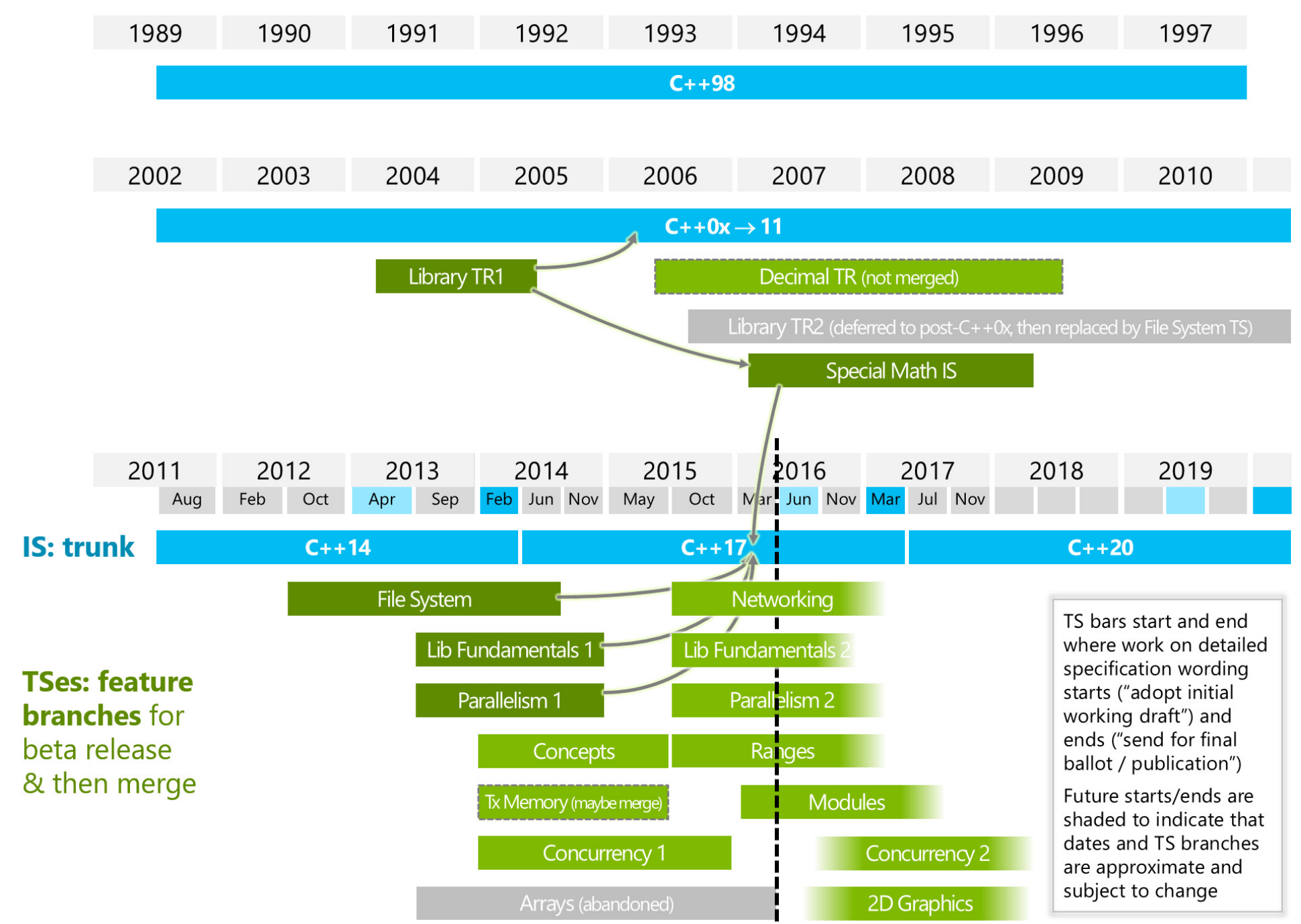


Figure : CPP Language roadmap



Ministerstwo Nauki i Szkolnictwa Wyższego

Organizacja IX Konferencji Modelowanie Matematyczne w Fizyce i Technice (MMFT 2017)  
 - zadanie finansowane w ramach umowy 829/P-DUN/2017 ze środków Ministra Nauki i Szkolnictwa Wyższego przeznaczonych na działalność upowszechniającą naukę.

