Run-time measurement of COSMIC functional size for Java business applications: Initial results

Ahmet Ata Akca  Ayça Tarhan, Phd.

Hacettepe University
Our Study
Run-Time Measurement
Our Study

- Functional Size Measurement Method
  - COSMIC

- Applied Application
  - Three-tier Java business applications

- Measurement Time
  - At runtime

- Functional Size Measurement Instrument
  - Self-developed “Measurement Library”

- Measurement Approach
  - Functional processes that are triggered via graphical user interface (GUI) of the application
Measurement Library
**What is “Measurement Library”?**

Measurement Library is set up within the frame of the rules according to COSMIC method and it enables automatic measurement.

Measurement Library records the data movements that are triggered via graphical user interface of the application for three-tier Java Business Applications. Data movements are used to discover the functional processes.

It is crucial that all functions should be operated on GUI at least once to be measured.
UML Definition of “Measurement Library”

```java
public class Measurement {
    private String id;
    private int functionalProcessIndex;
    private long entryTime;
    private long dbTime;
    private long exitTime;
    private List<Integer> eventId = new ArrayList<>();
    private List<FunctionalProcess> functional = new ArrayList<>();

    public Measurement() {
    }

    public Measurement(String id, int functionalProcessIndex, long entryTime, long dbTime, long exitTime) {
        this.id = id;
        this.functionalProcessIndex = functionalProcessIndex;
        this.entryTime = entryTime;
        this.dbTime = dbTime;
        this.exitTime = exitTime;
    }

    public void notifyEntryMovement(ActionEvent evt) {
    }

    public int controlID(int id) {
        return id;
    }

    public void notifyDBMovement(String query) {
    }

    public void notifyExitMovement() {
    }

    public void calculateCosmic() {
    }
}
```
Calculation Steps and Phases by Using Measurement Library

- Pre-Calculation Phase:
  - Importing Measurement Library
  - Making Necessary Code Additions

- Calculation Phase:
  - Counting of Data Movements

- Record & Discover Phase:
  - Discovery of a Functional Process
  - Recording of Data Movements
How to Use “Measurement Library”?

Pre-Calculation Phase

In the “pre-calculation” phase preparation for the measurement is carried out in two main steps:

1. Importing Measurement Library
2. Making Necessary Code Additions
How to Use “Measurement Library”? 

Record & Discover Phase

In the “record & discover ” phase process for the measurement is carried out in two main steps:

- **Step 3**: Recording of data movements
- **Step 4**: Discovery of a functional process
How to Use "Measurement Library"?

Calculation Phase

In the "counting of data movements" phase process for the measurement is carried out in a step:
Example Application
Measurement Library is applied to a basic “Student Registration System” which performs some of the basic transactions.

• Within the scope of this Java application, the student can be searched, registered, updated and deleted from the system.

• Success and failure messages regarding the transactions displayed via GUI.
Step 1 Importing Measurement Library

1.1 Measurement Library has been imported from the SDB classes related to GUI and DB within the pre-calculation phase.
**Pre-Calculation Phase**

### Step 2 Making Code Additions

1. "NotifyEntryMovement()" call has been added into the `ActionPerformed()` method which is triggered by transactions on GUI.

2. "NotifyDBMovement()" call is added into `ExecuteQuery()` and `ExecuteQueryUpdate()` methods by which database queries are operating.

3. "NotifyExitMovement()" method has been called below `propertyChange()` method which is triggered when any component on GUI has been manipulated.

4. "CalculateCosmic()" method is called in `ActionPerformed()` method in which the movement of "Shut Down" button on GUI is clicked.
Record&Discover Phase

Step 3 Recording of Data Movements
3.1 When a user generates add, delete, search and update transactions, “Measurement Library” automatically records these data movements with their realization times.

Step 4 Discovery of a Functional Process
4.1 By using time info of data movements, “Measurement Library” automatically discover whether data movements compose a functional process or not. (For the sample application, the time info is set as one (1) second.)
Step 5 Counting of Data Movements

5.1 All related data movements have been counted for each previously recorded functional process. Thereafter, each data movement is calculated as 1 CFP as it is pointed out in COSMIC Measurement Manual.
### TABLE I. RESULTS OF AUTOMATIC MEASUREMENT FOR SDB

<table>
<thead>
<tr>
<th>Application Registry</th>
<th>Detected Functional Processes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Process1</td>
</tr>
<tr>
<td>Entry</td>
<td>1</td>
</tr>
<tr>
<td>Exit</td>
<td>1</td>
</tr>
<tr>
<td>Read</td>
<td>-</td>
</tr>
<tr>
<td>Write</td>
<td>1</td>
</tr>
<tr>
<td><strong>TOTAL:</strong></td>
<td>3</td>
</tr>
<tr>
<td><strong>12 FP</strong></td>
<td></td>
</tr>
</tbody>
</table>

### TABLE II. RESULTS OF MANUAL MEASUREMENT FOR SDB

<table>
<thead>
<tr>
<th>Measurement Registry</th>
<th>Process Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Add</td>
</tr>
<tr>
<td>Entry</td>
<td>1</td>
</tr>
<tr>
<td>Exit</td>
<td>1</td>
</tr>
<tr>
<td>Read</td>
<td>-</td>
</tr>
<tr>
<td>Write</td>
<td>1</td>
</tr>
<tr>
<td><strong>TOTAL:</strong></td>
<td>3</td>
</tr>
<tr>
<td><strong>13FP</strong></td>
<td></td>
</tr>
</tbody>
</table>
EXAMPLE APPLICATION (RESULTS)

Manuel vs Automatic Measurement

- Calculated Functional Points
- Data Movement Type: Entry, Exit, Read, Write
- Blue line: Automatic Measurement
- Red line: Manuel Measurement
CONCLUSION & FUTURE STUDIES
CONCLUSION & FUTURE STUDIES

CONCLUSION

COSMIC functional size has been calculated for a sample application by using Measurement Library and automatic measurement value has been obtained at the end of this calculation. This value has been compared with manual measurement value that is also calculated by using same application.

We have observed that automatic and manual measurement values are 92% convergent. These values enable us to rely on applicability of this method. The ratio of additions to the total code length is 2%.
Within the scope of future studies, we will

• apply Measurement Library on relatively large applications and observe the results.

• find a better solution for time info constraint.

• consider the triggering frequencies of functionalities for application. By the consideration of this frequency, both efficiency of applications can be evaluated and performance of functional processes can be increased.

• prepare a guideline for making code additions in the pre-calculation phase. And try to make the code addition process automatically by making code analysis.
QUESTIONS
THANK YOU

AHMET ATA AKCA
Steps of Using Measurement Library

Step 1: Importing Measurement Library
In this step, Measurement Library is imported from a three-tier Java business application.
Step 2: Making Necessary Code Additions
In this step, necessary code additions must be done to the specific areas of the code where GUI and DB movements take place. For GUI actions we need to add NotifyEntryMovement() and NotifyExitMovement() method calls. For DB actions we need to add NotifyDBMovement() method call. For the calculation of the total functional size we need to add CalculateCosmic() method call to the exit point of the code.
Step 3: Recording of data movements:
In this step, four main data movements (Entry, Read, Write, Exit) on which COSMIC method is based are recorded with the information of “realization time” which we will use for the discovery of functional processes afterwards.
Step 4: Discovery of a functional process:
In this step, by analyzing the previously recorded time information and related movements, it is detected whether the movements create a functional process according to COSMIC method. For COSMIC Method, a functional process consists of at least an Entry data movement, and a following Write or Exit data movement.
Steps of Using Measurement Library

Step 5: Counting of data movements:
Firstly, all related data movements have been counted for each previously recorded functional process and each data movement is calculated as 1 CFP. After functional sizes are calculated one by one for all functional processes, calculated values are added and thereby functional size value for the whole application is obtained in CFP.