A Theory of User-interaction Objects

Fabio Paterno

Centro Nazionale Universitario di Calcolo Elettronico, Institute of CNR

August 28, 2019

Table of Contents

1. Problem Definition
2. Motivation
3. Previous Works
4. What is a UIS?
5. Building block of a UIS
6. Architectural Model of an Interactor
7. How it works?
8. Formal Definition of an Interactor UIS

Problem Definition

How to define user interface systems so that

- They can be described formally using precise mathematical notation
- Their behavior and properties can be methodically evaluated

User Experience

User Interfaces are an important part of the user experience.
They directly affect the users ability to analyze and understand the information presented.

But

They are way too complicated!

Motivation

Taking a mathematical approach to UI design helps us:

- Provide useful indicators and guidelines to UI designers during design refinement phase

Motivation

Taking a mathematical approach to UI design helps us:

- Provide useful indicators and guidelines to UI designers during design refinement phase
- Create better automated tools for UI design and testing

Motivation

Taking a mathematical approach to UI design helps us:

- Provide useful indicators and guidelines to UI designers during design refinement phase
- Create better automated tools for UI design and testing
- Create more reusable UI elements

Previous Works

B. A. Myers, ACM Transactions on Information Systems 8, 289-320

A new Model for handing input

J. Foley et al. (1991) UIDE

An Intelligent User interface design environment

L. Cinque et al. (1990)

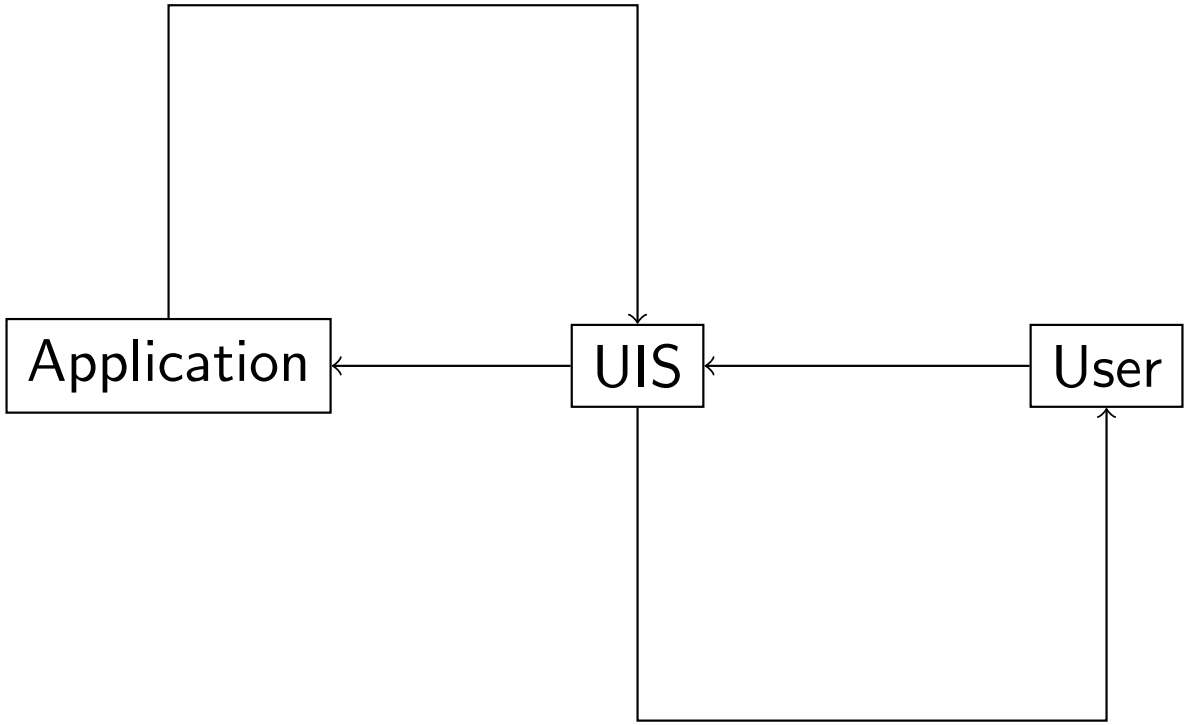
Towards a formal specification methodology for iconic interface design

What is a UIS?

Every Application has 3 major components.

What is a UIS?

Every Application has 3 major components.



Building Block of a UIS : Interactor

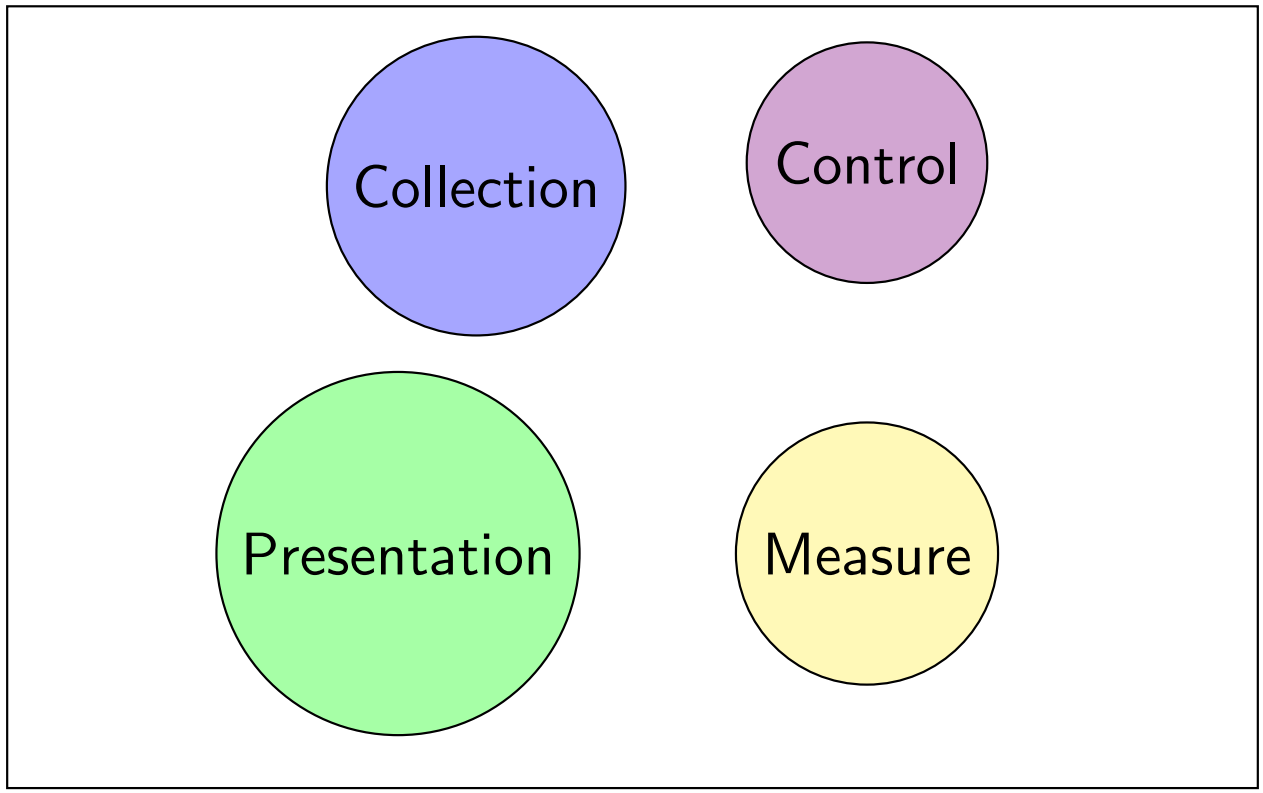
As we can see UIS is the component that communicates between the user end and the application end.Each UIS is basically a composition of a much smaller components.
And we are calling it the **'Interactor'**

Architectural Model of an Interactor

An Interactor consists of 4 architectural components.

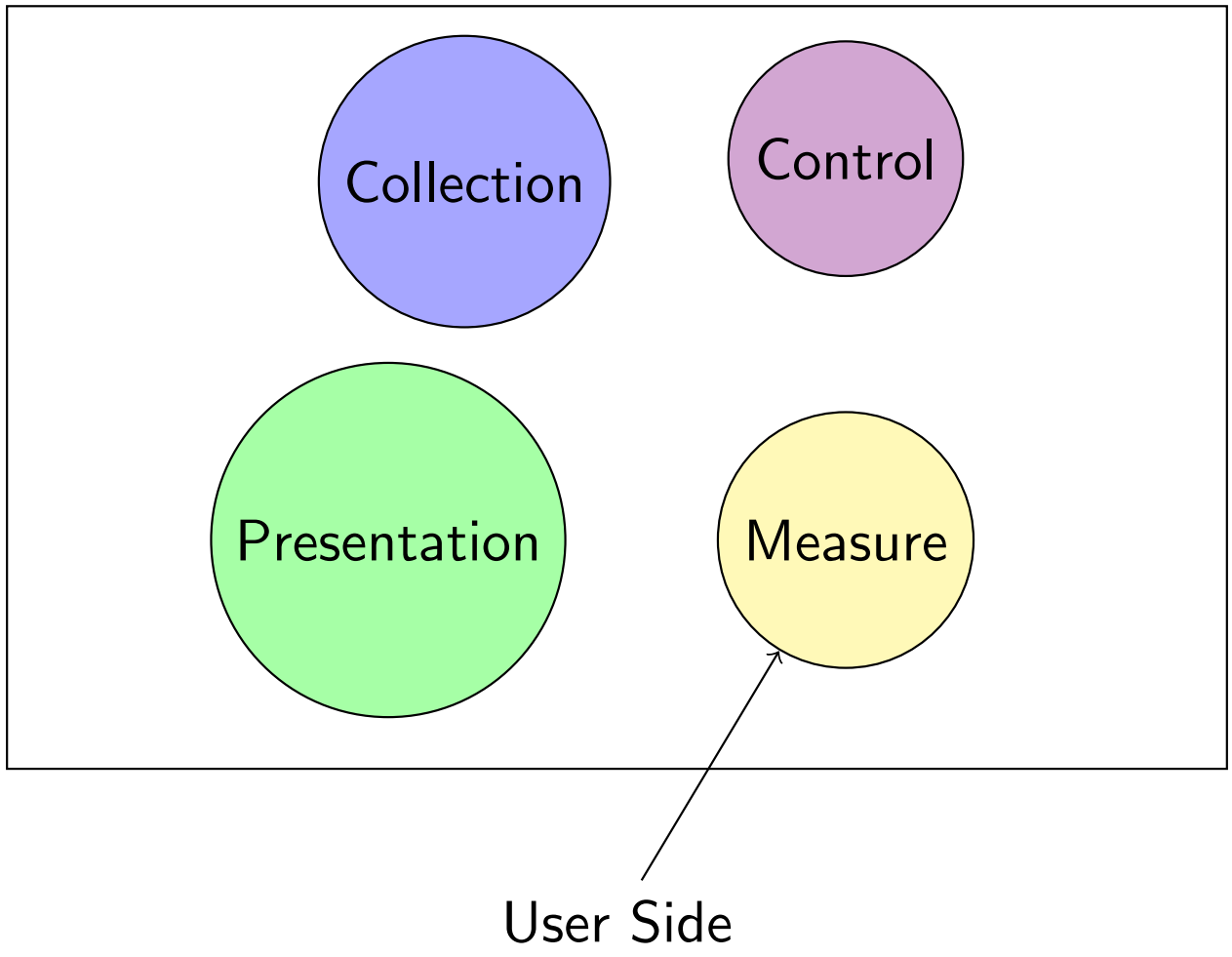
- Measure
- Control
- Collection
- Presentation

Application Side

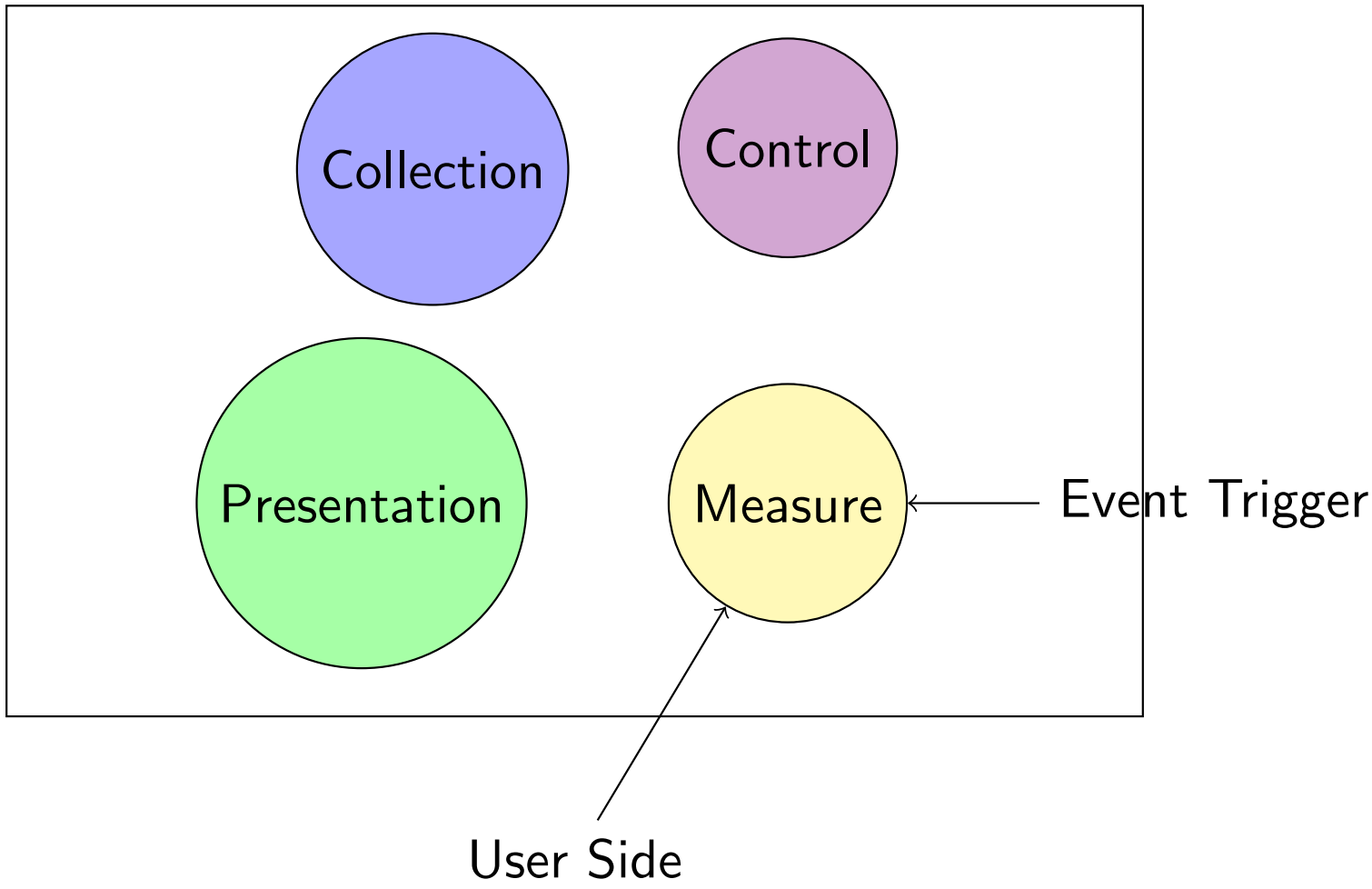


User Side

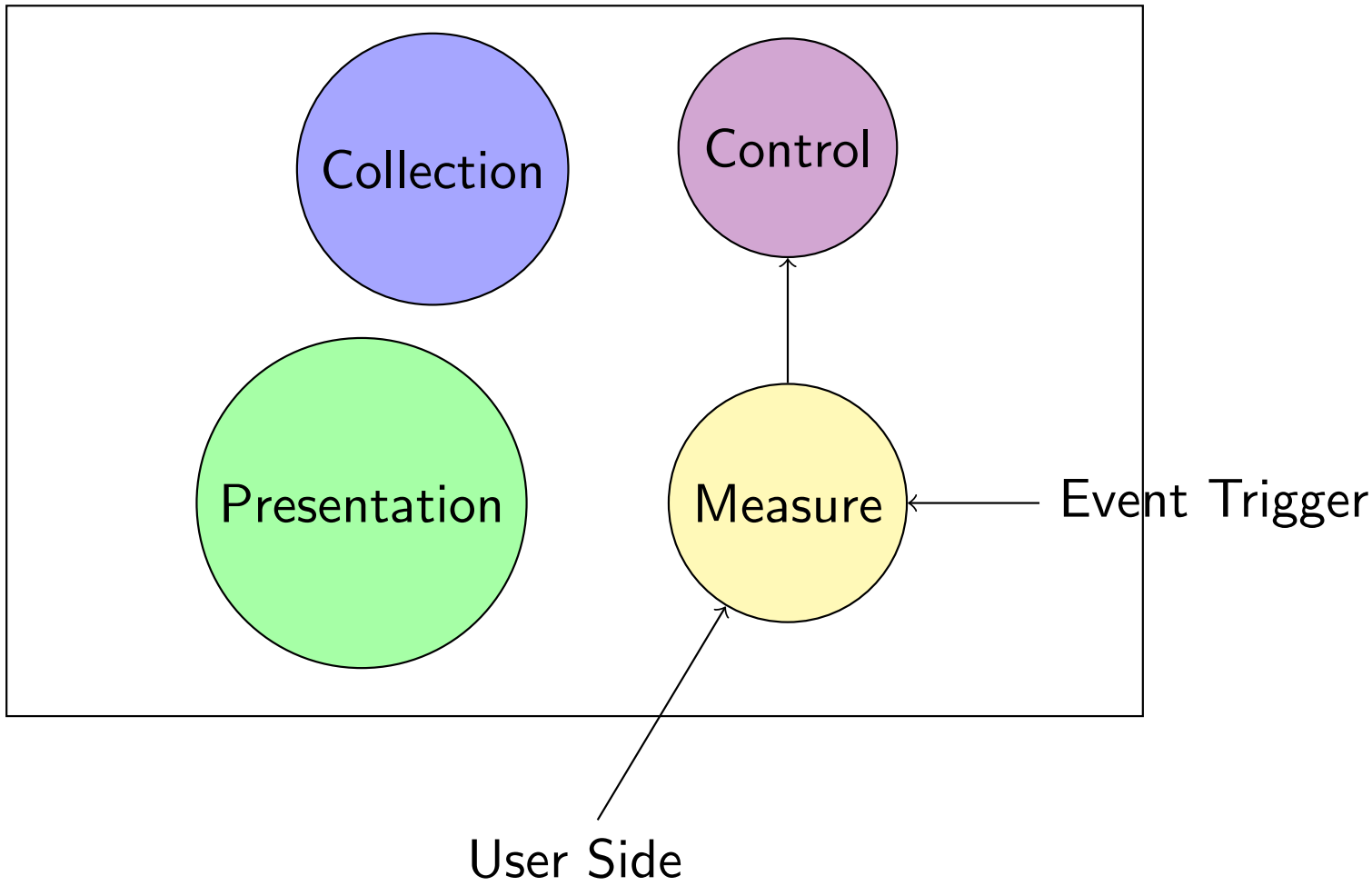
Application Side

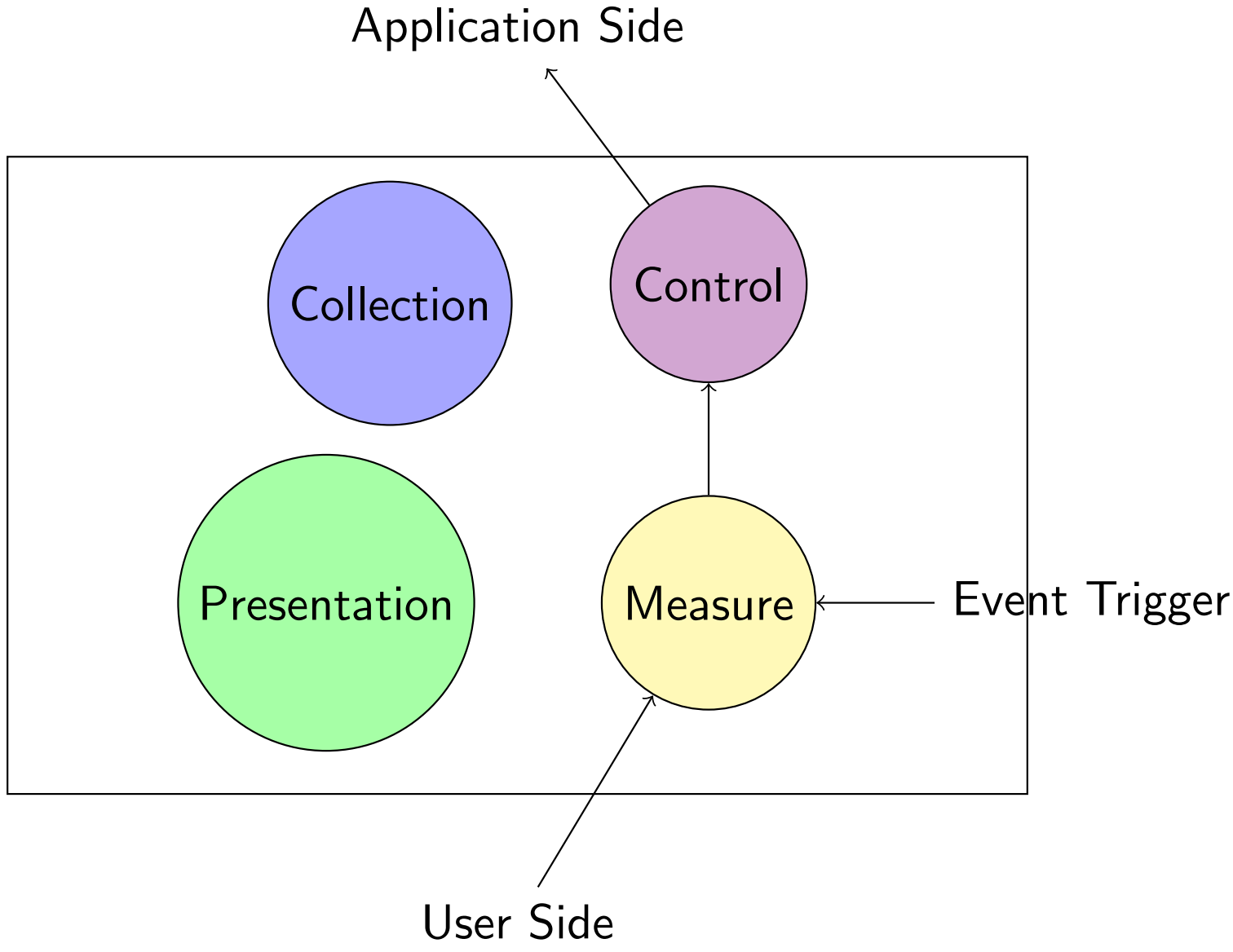


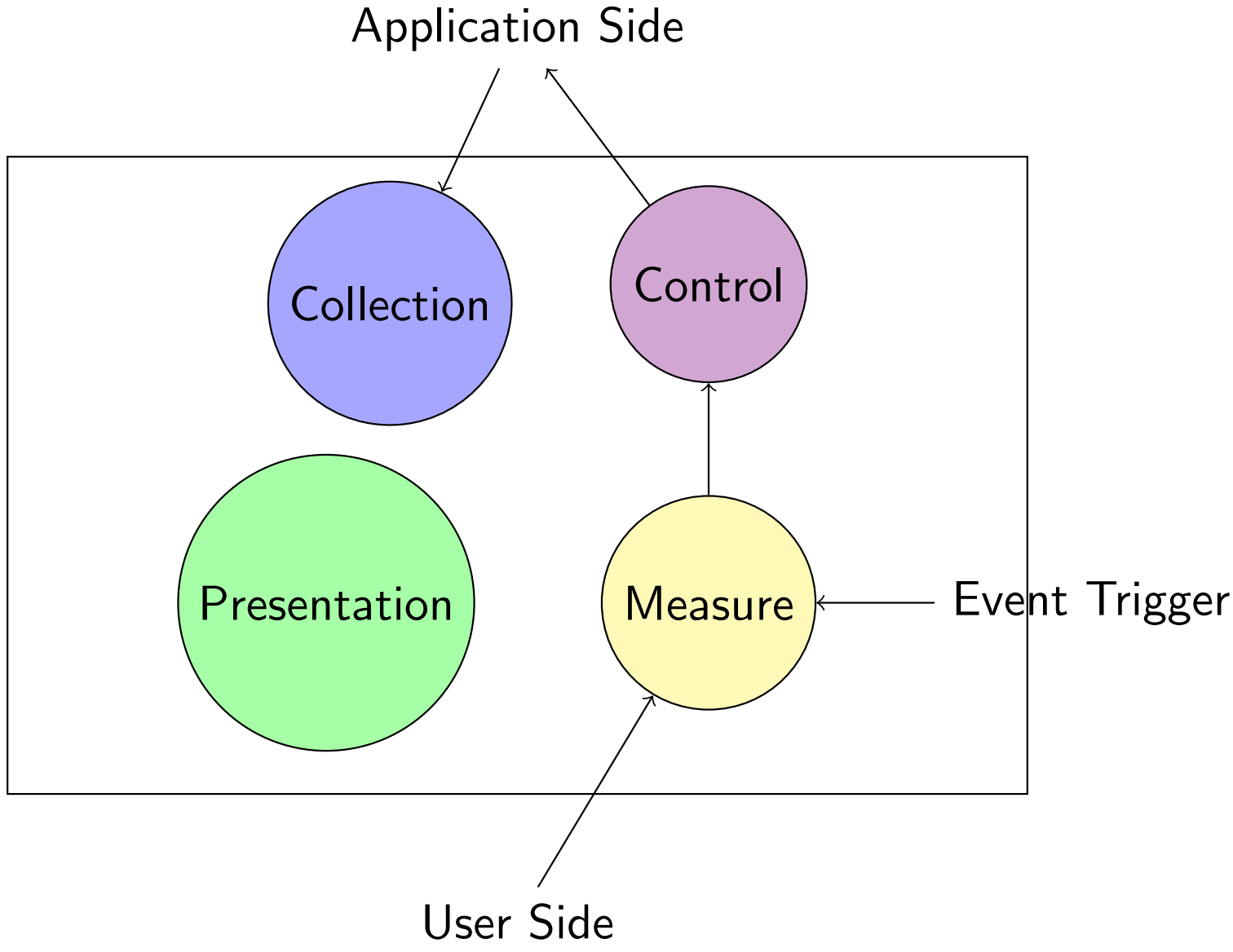
Application Side

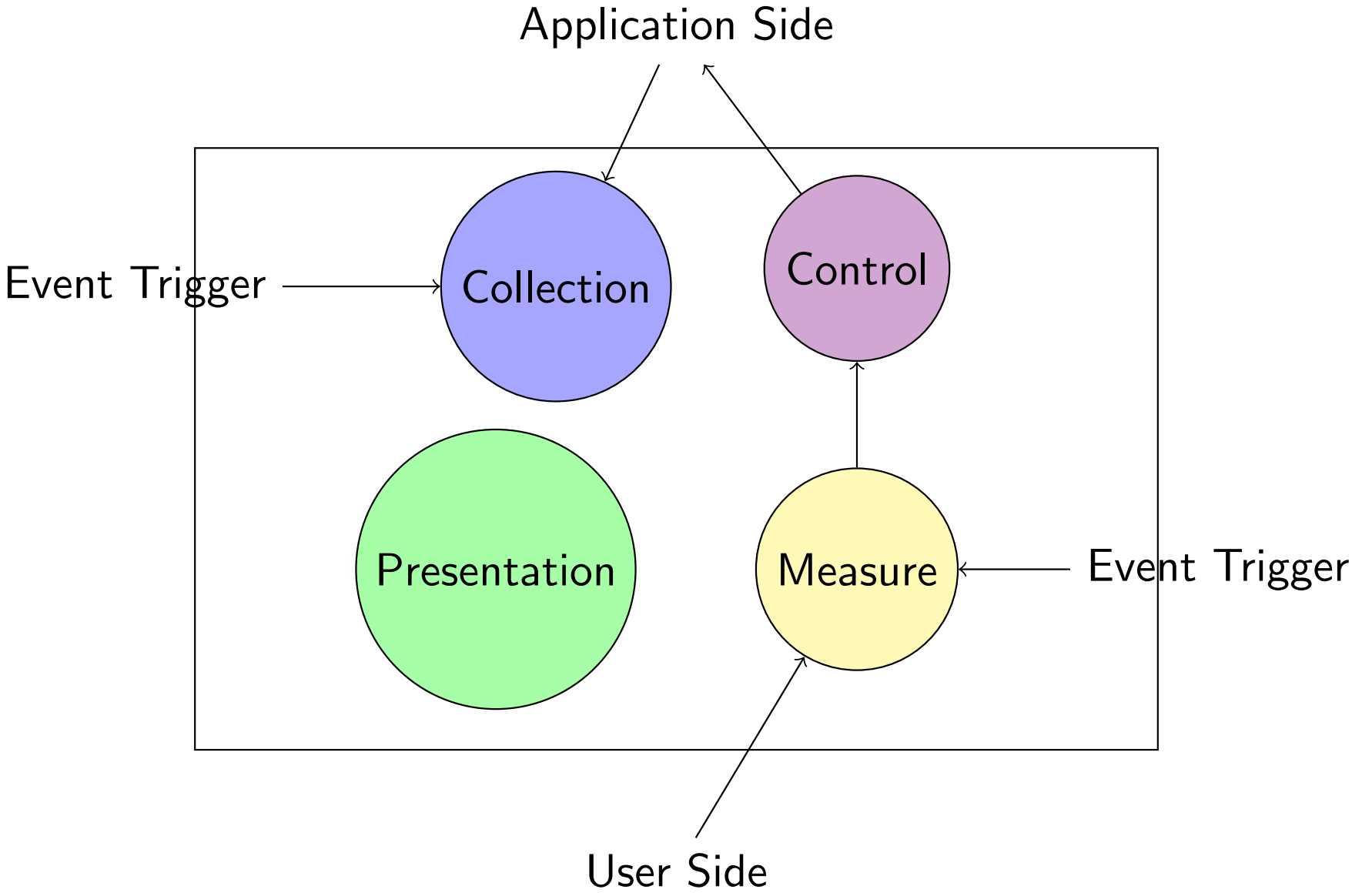


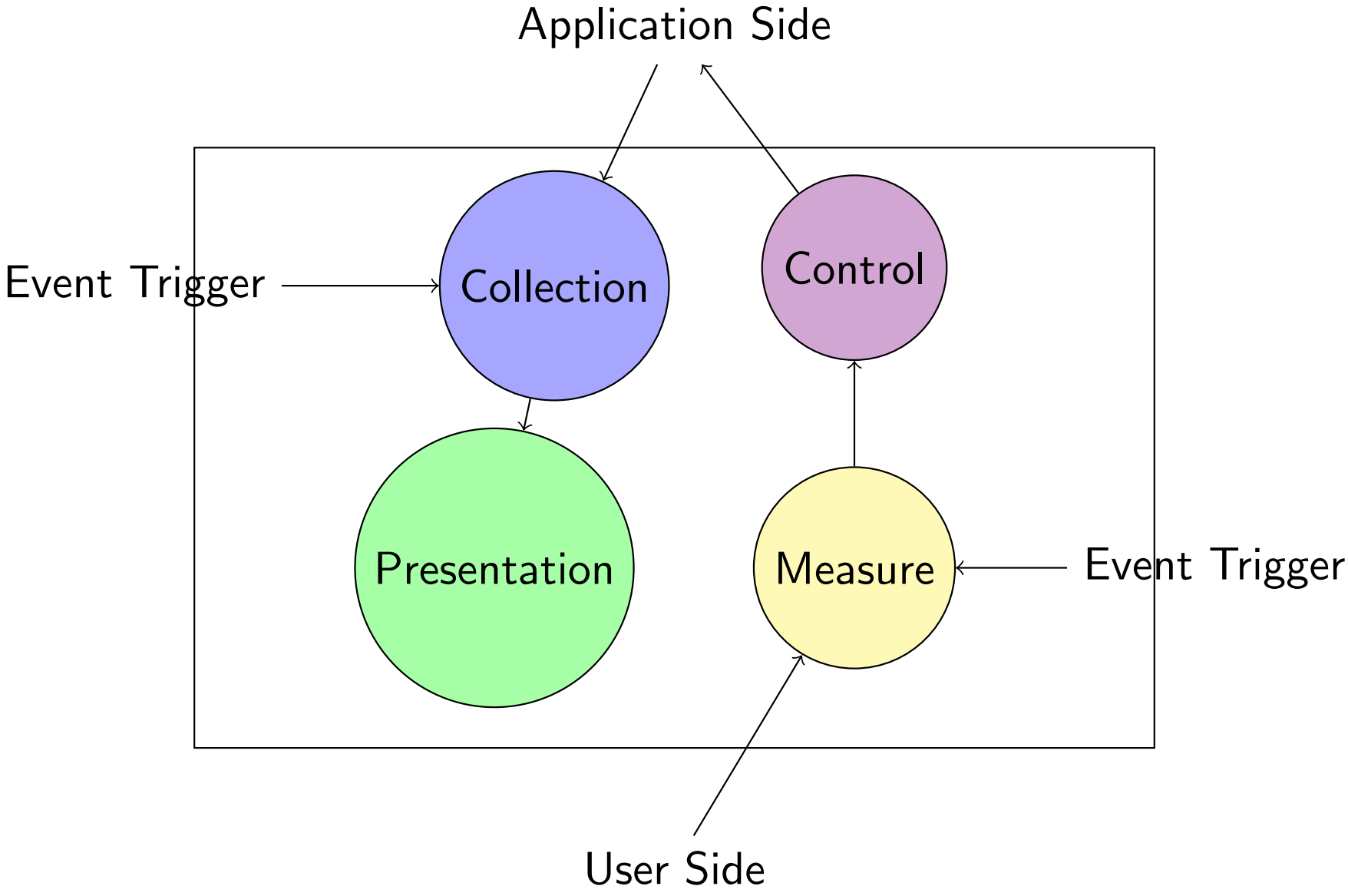
Application Side

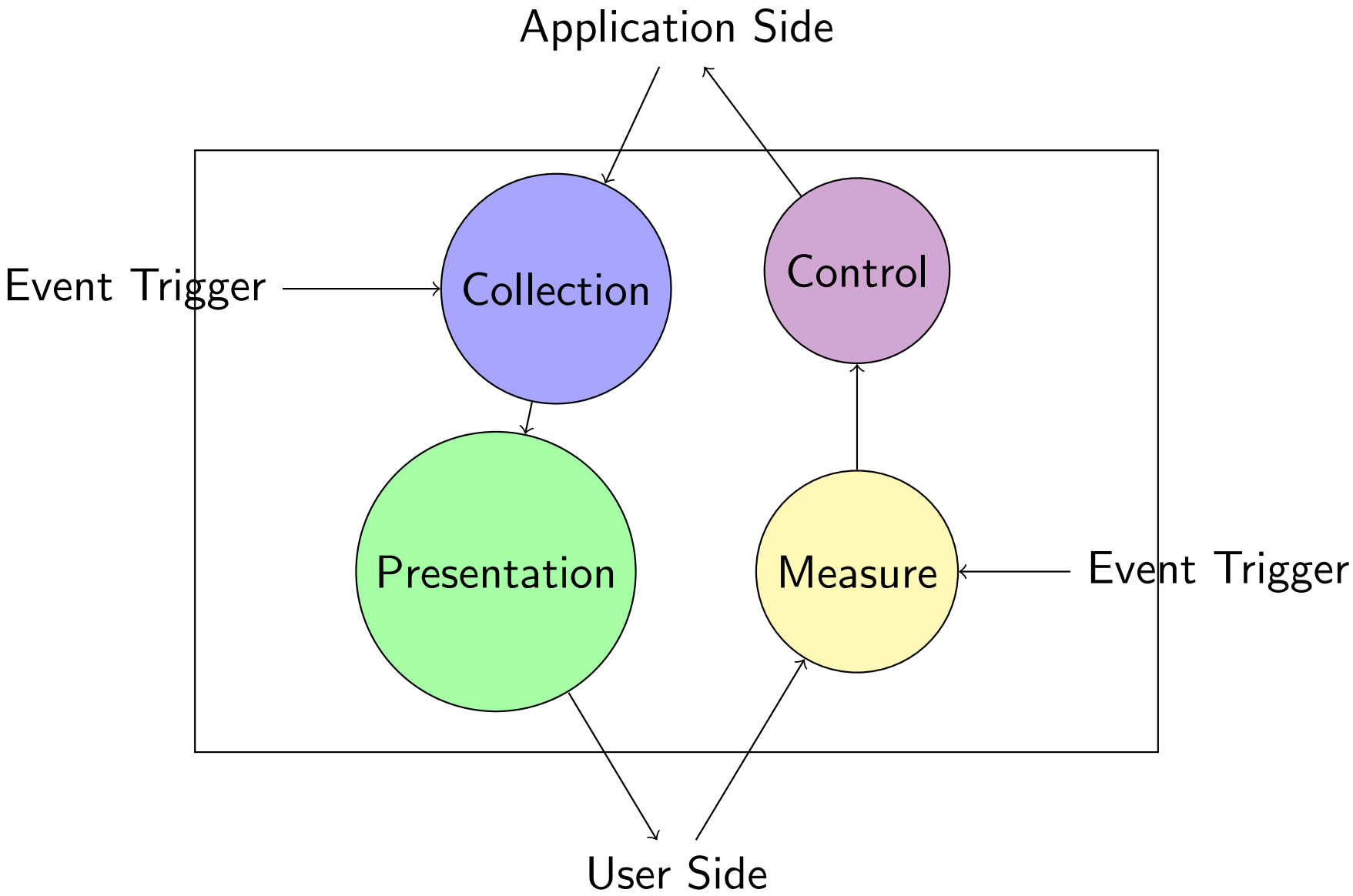



Application Side
Collection
Control
Presentation
Measure
Event Trigger
User Side


Application Side
Collection
Control
Presentation
Measure
Event Trigger
User Side

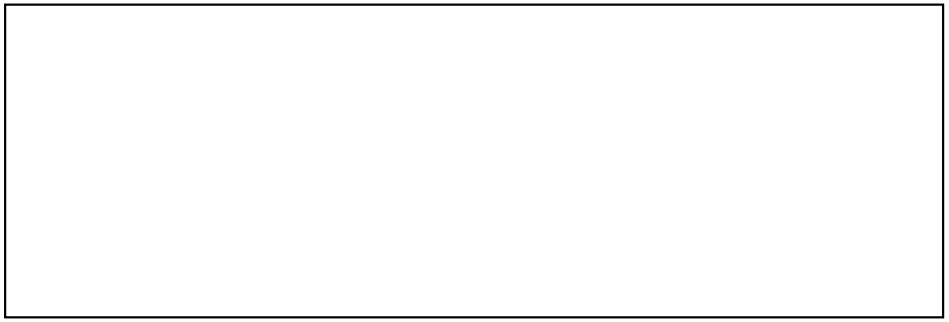

Application Side
Event Trigger
Collection
Control
Presentation
Measure
Event Trigger
User Side


Application Side
Event Trigger
Collection
Control
Presentation
Measure
Event Trigger
User Side


Application Side
Event Trigger
Collection
Control
Presentation
Measure
Event Trigger
User Side

How it works?

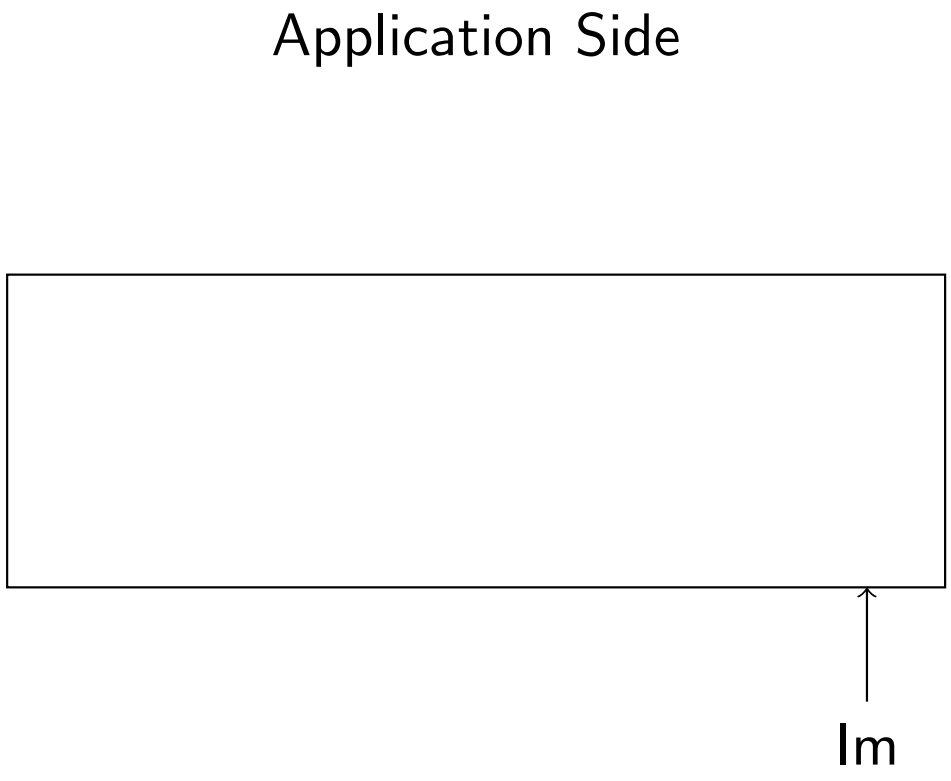
Application Side



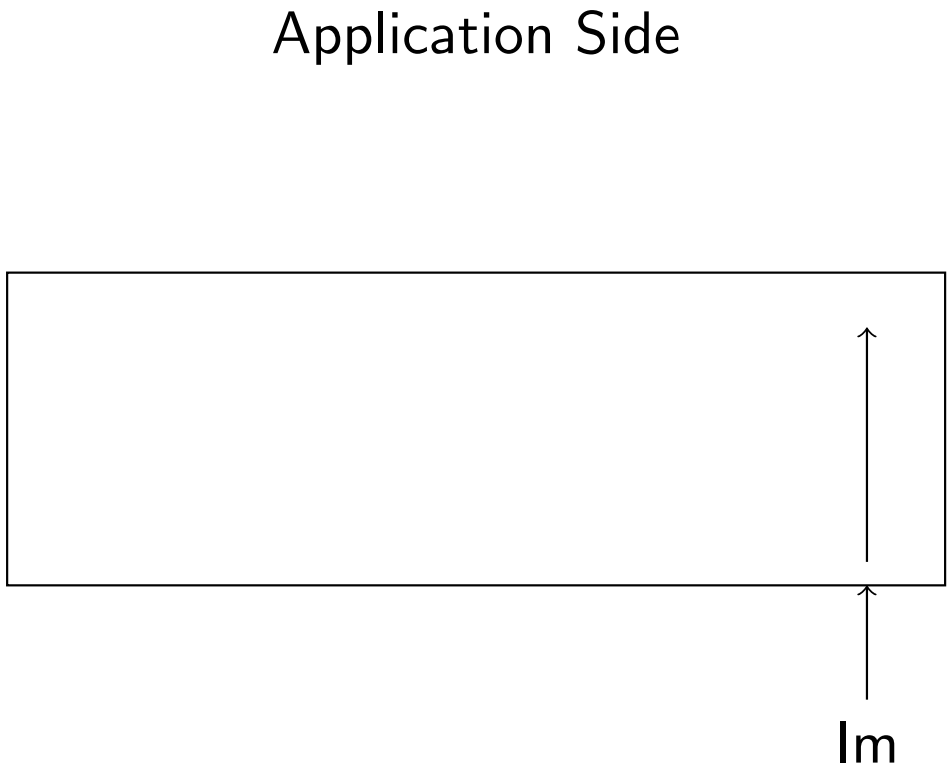
Im

User Side

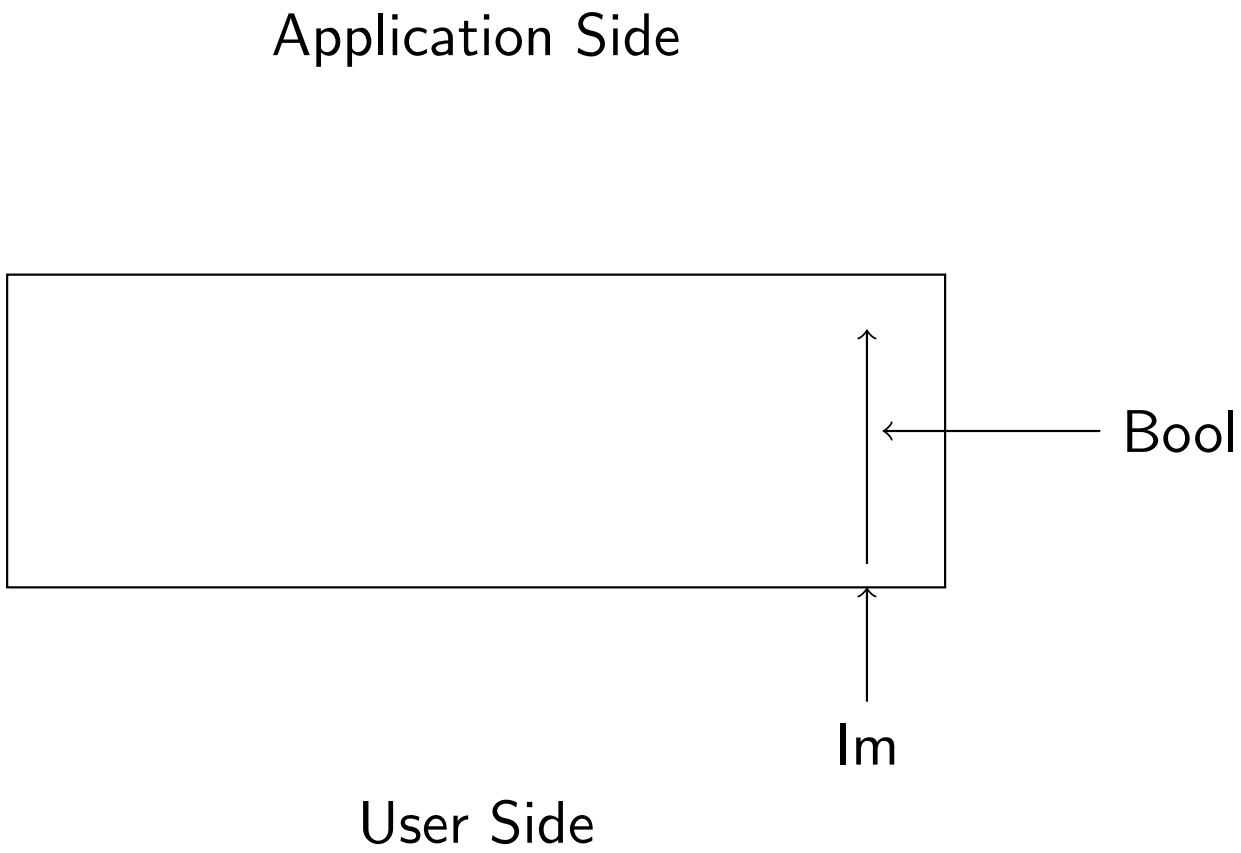
How it works?



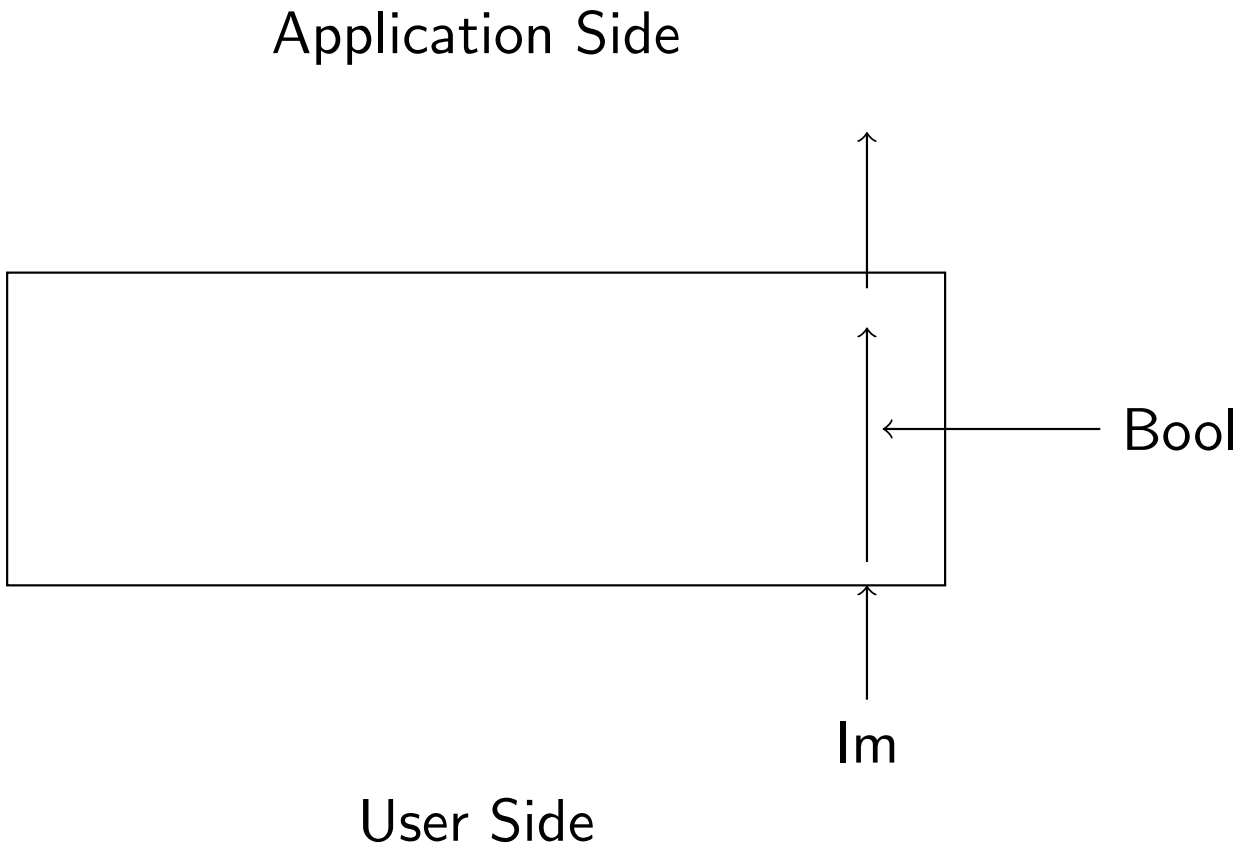
How it works?



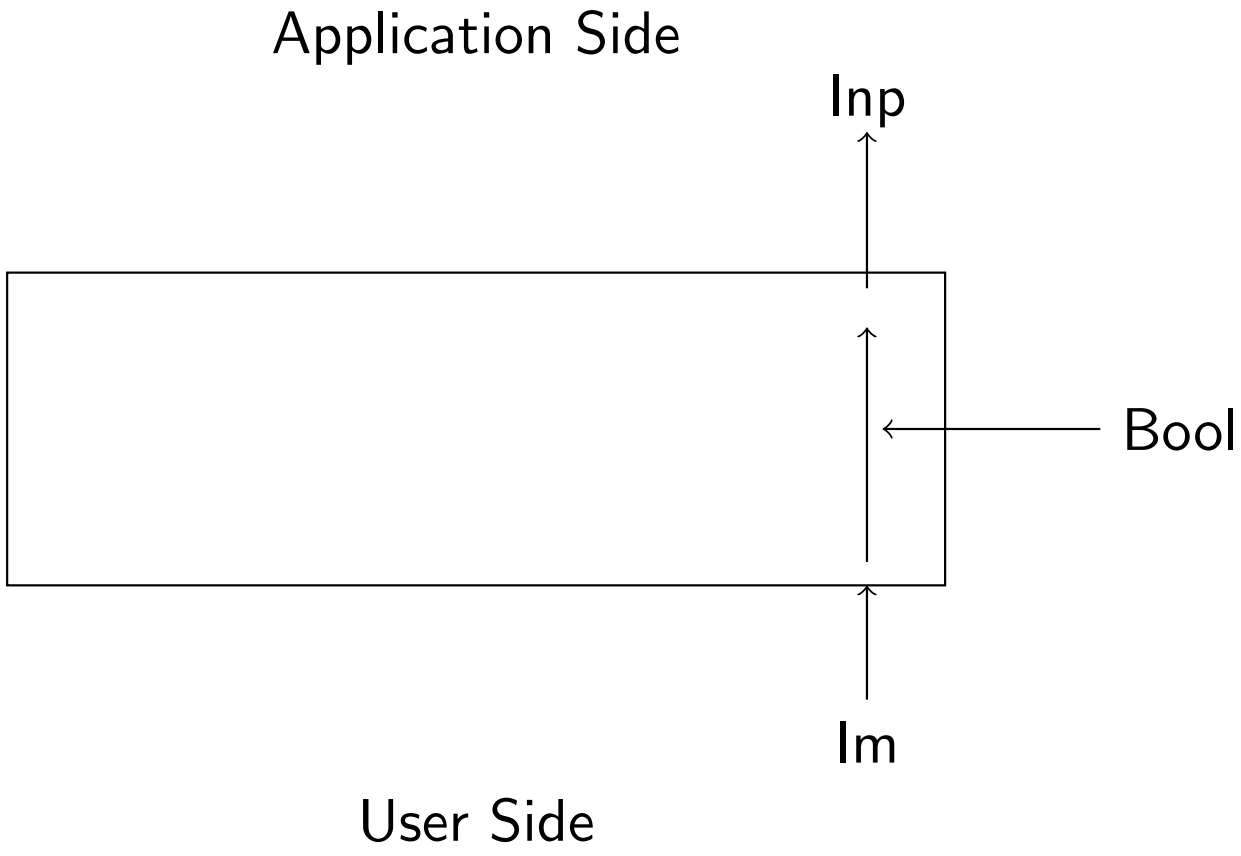
How it works?



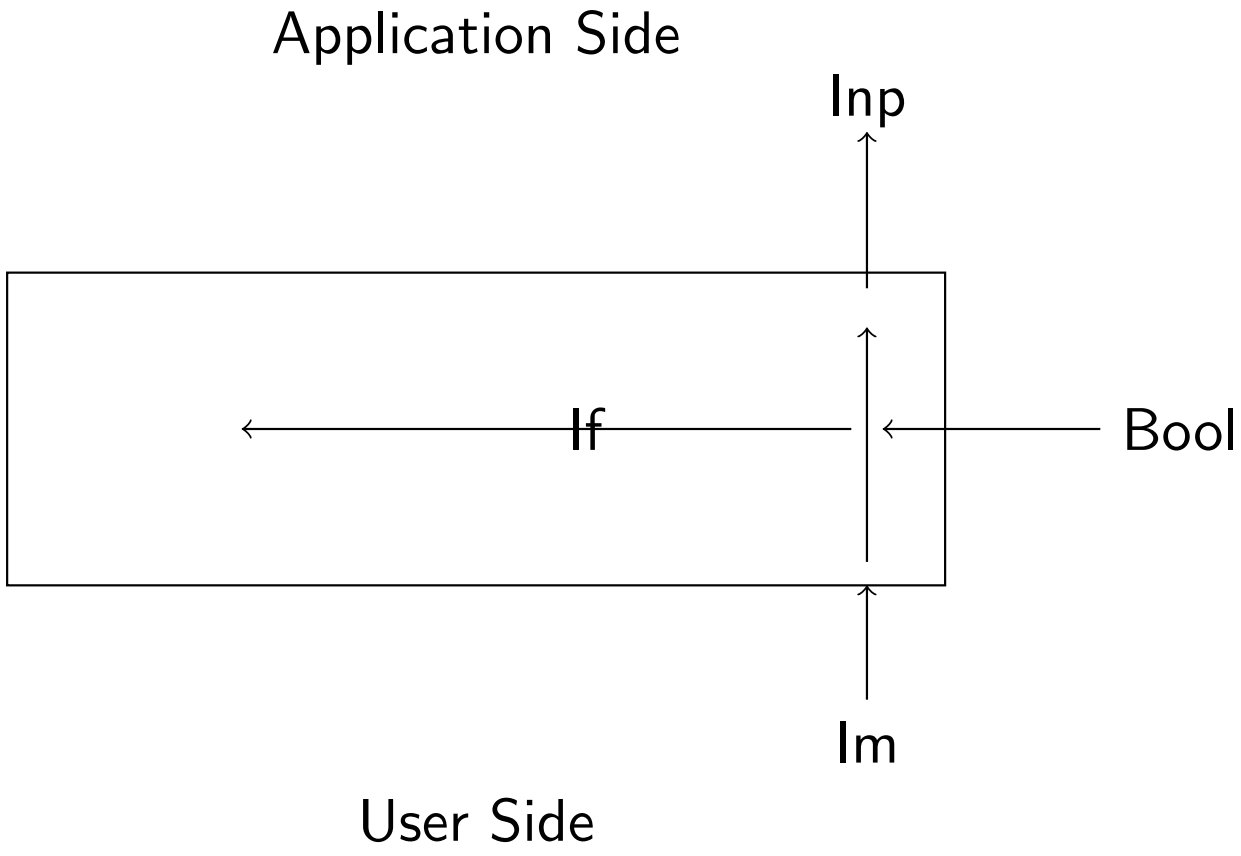
How it works?



How it works?

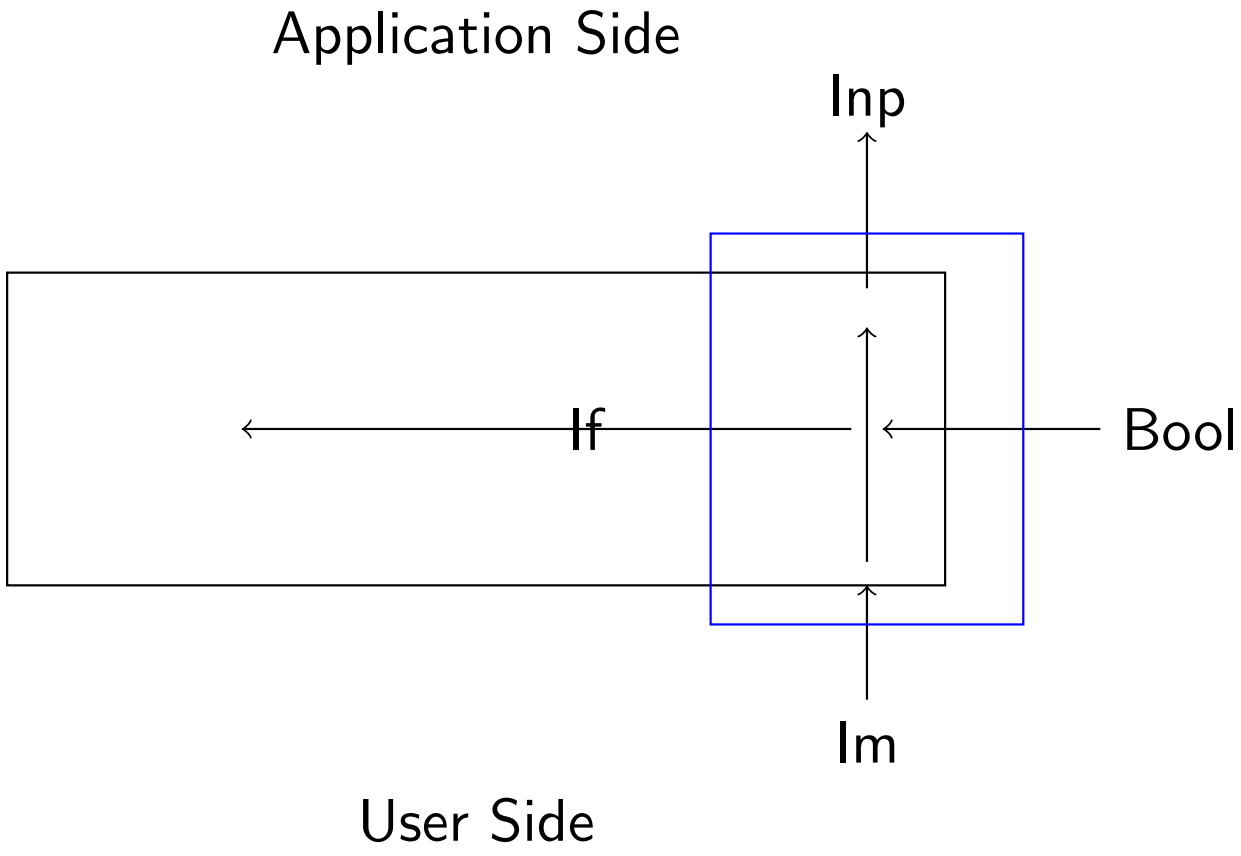


How it works?



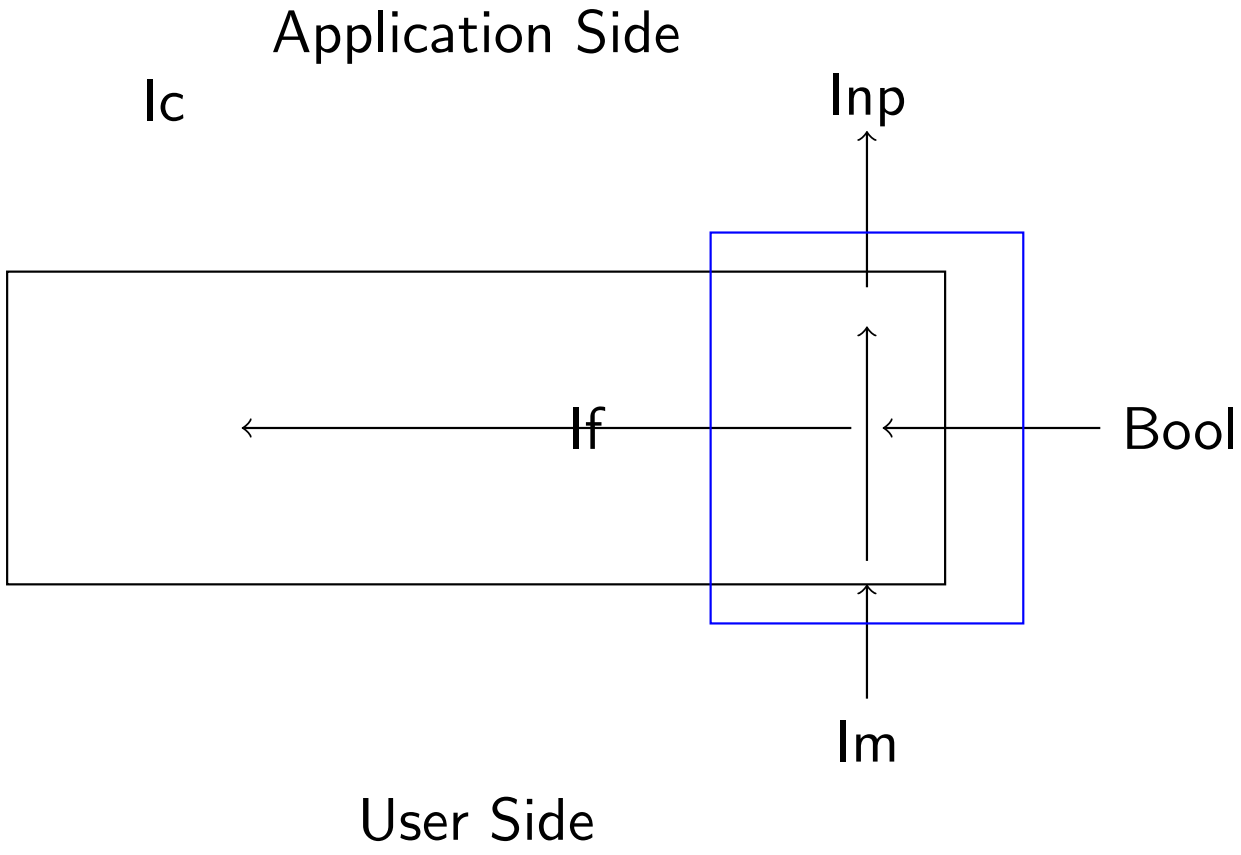
How it works?

- Input Function



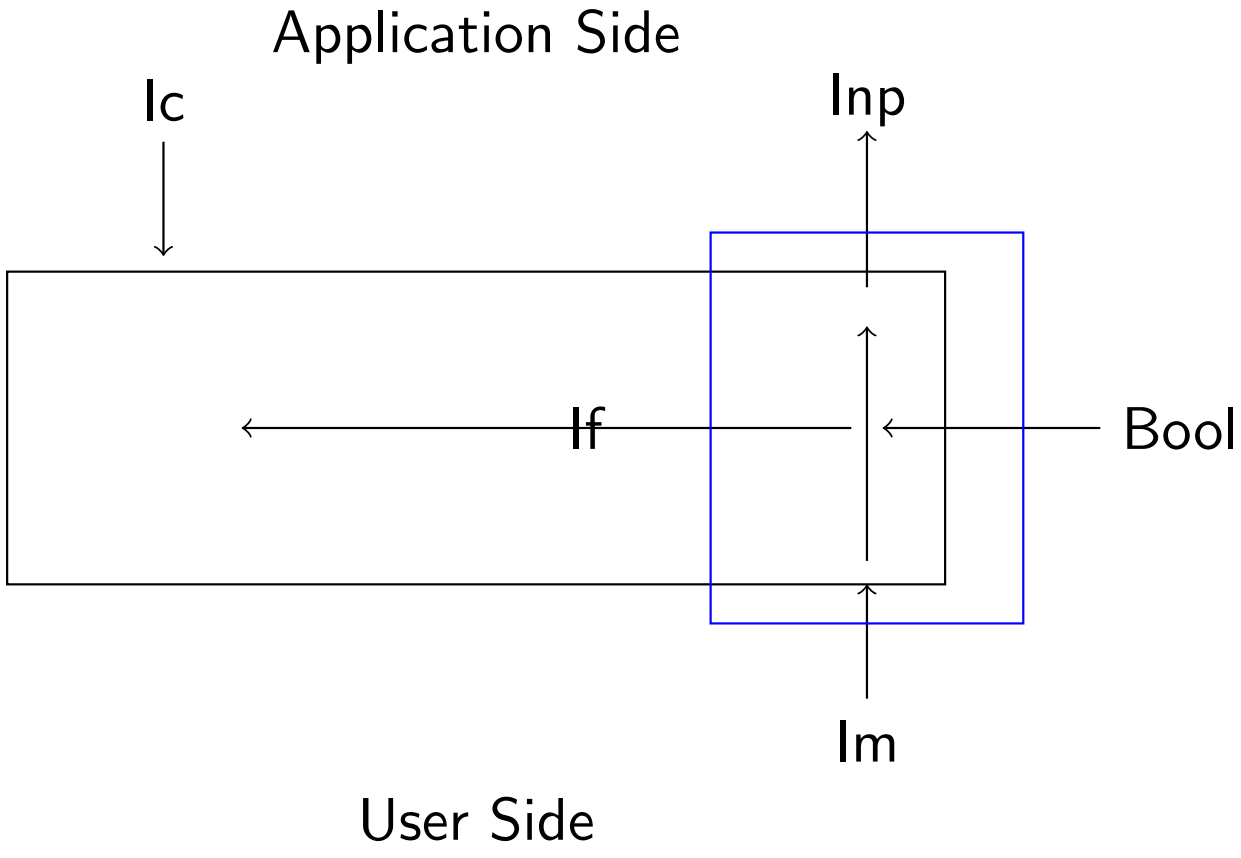
How it works?

- Input Function



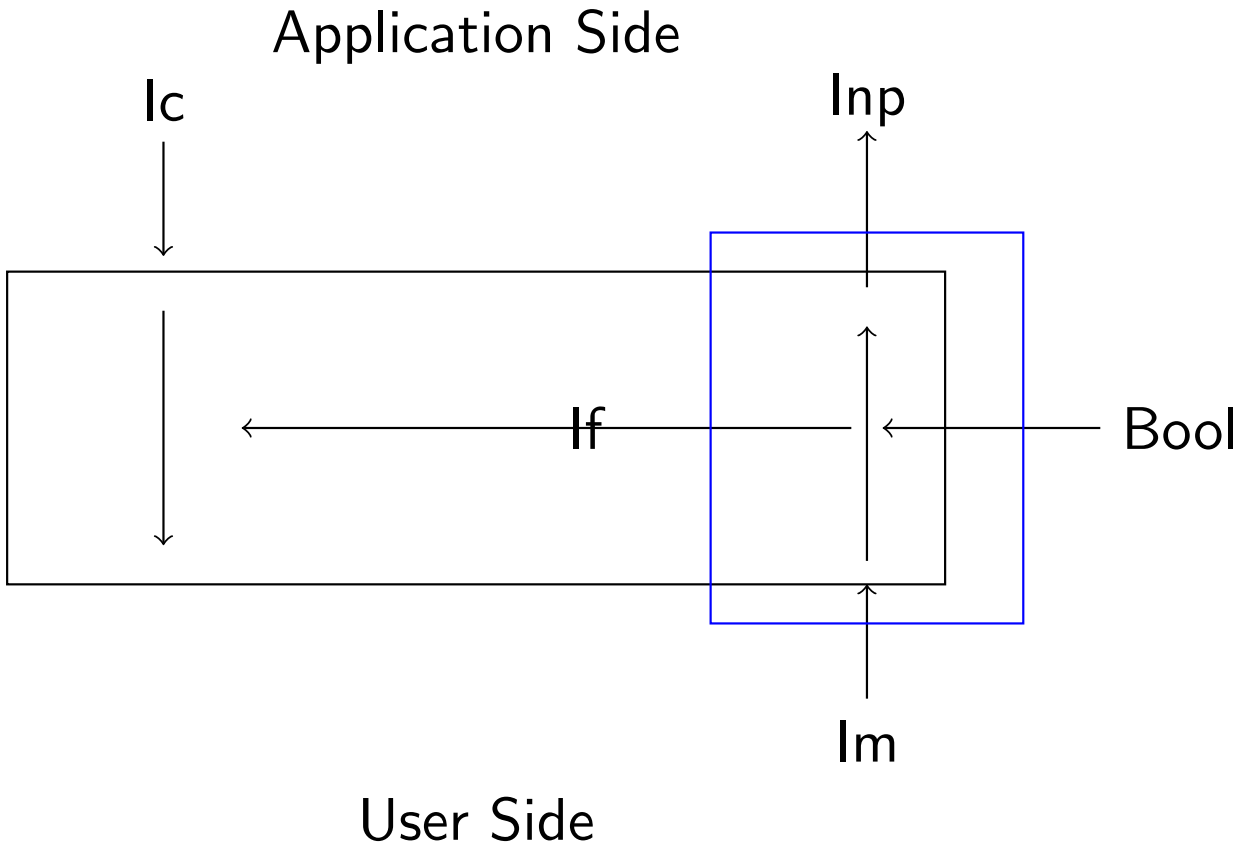
How it works?

- Input Function



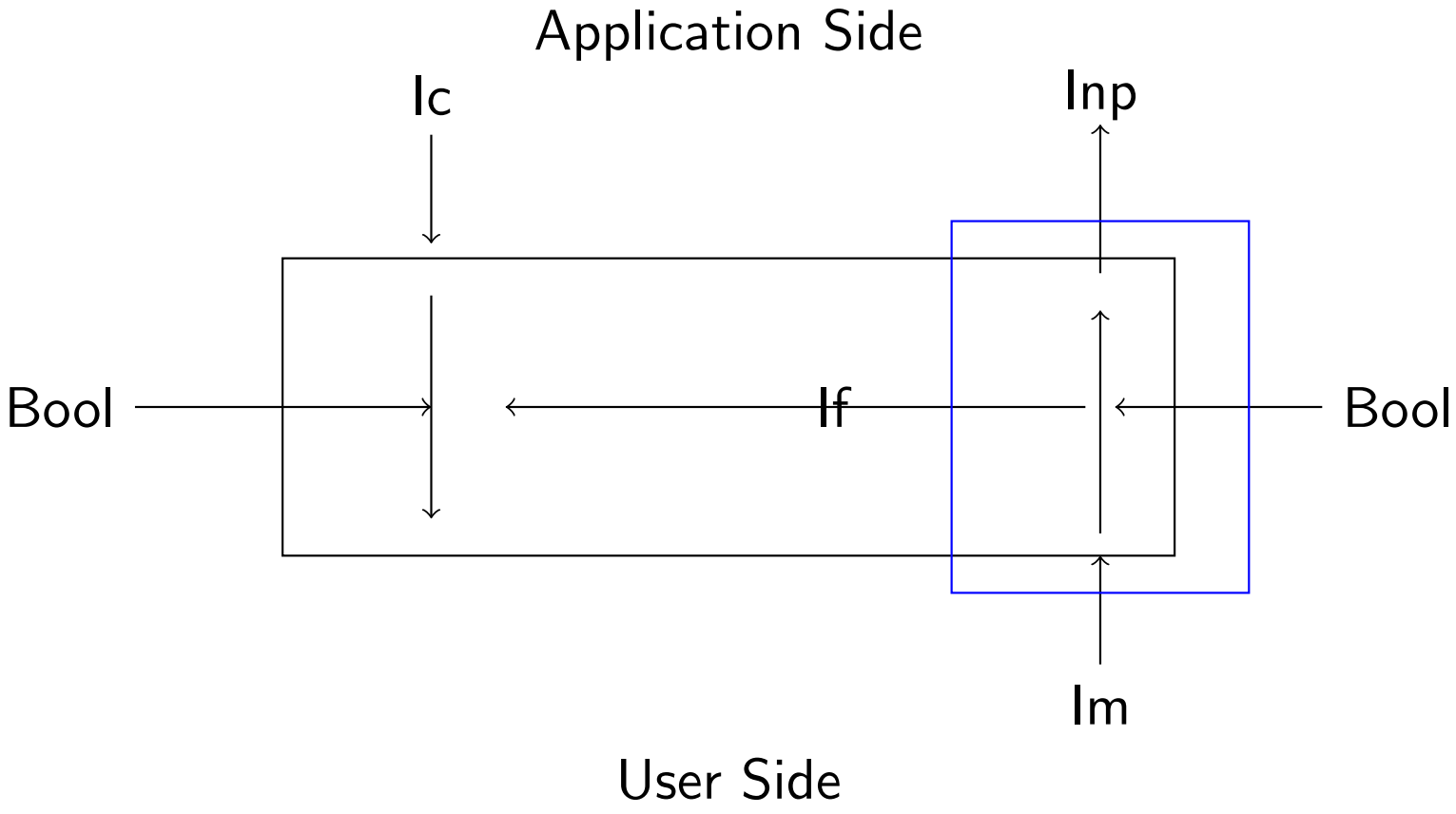
How it works?

- Input Function



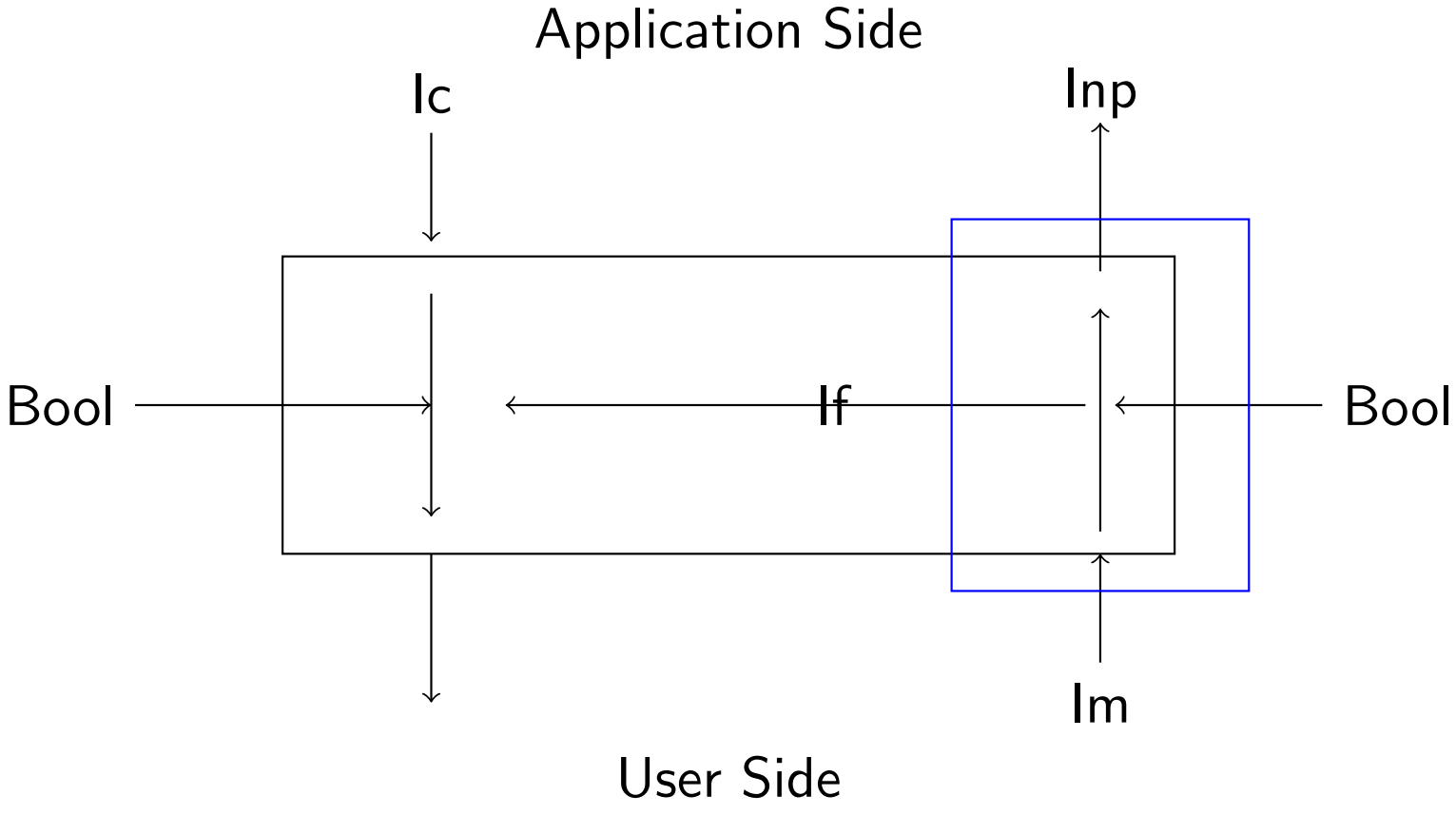
How it works?

- Input Function



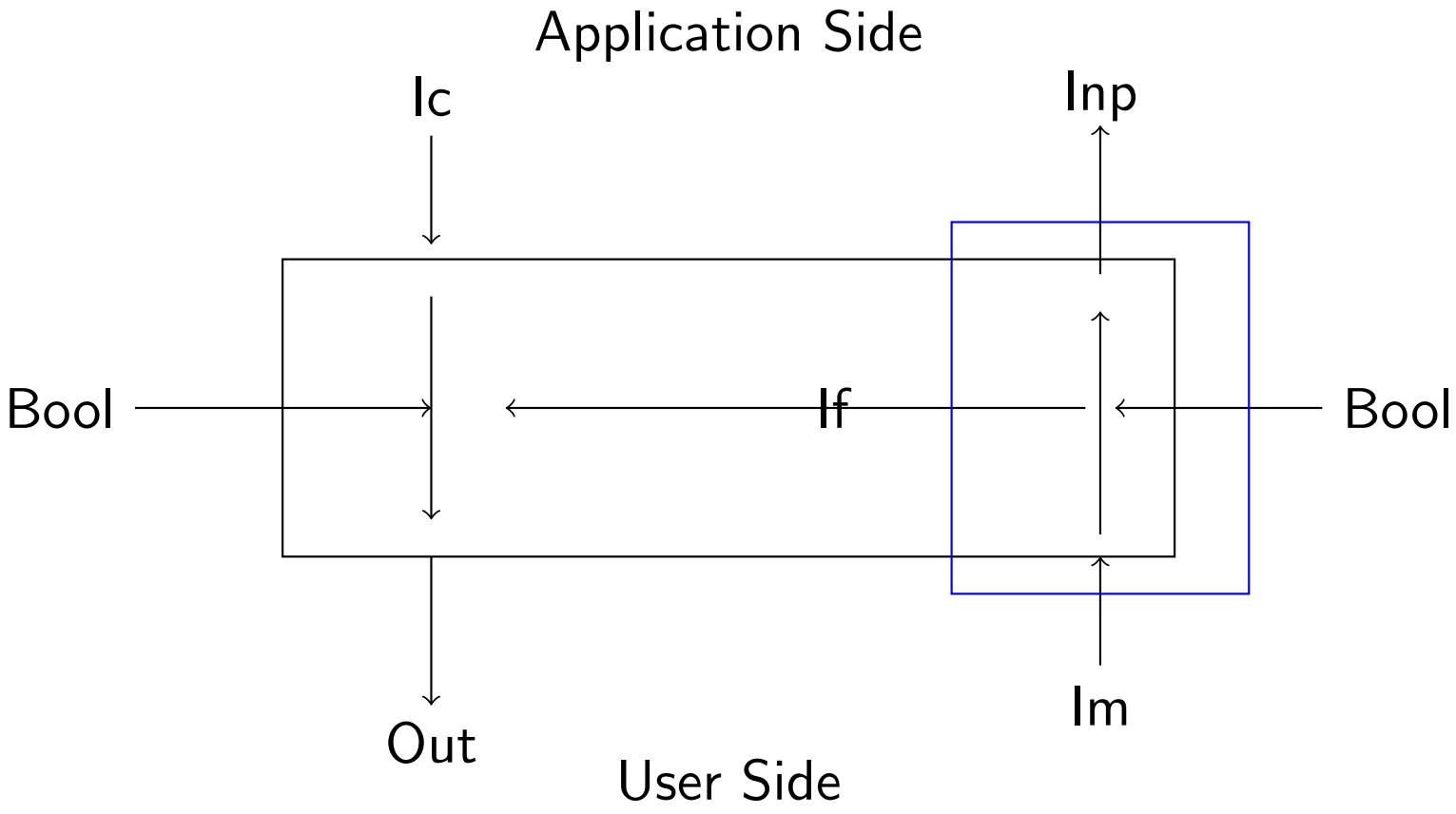
How it works?

- Input Function

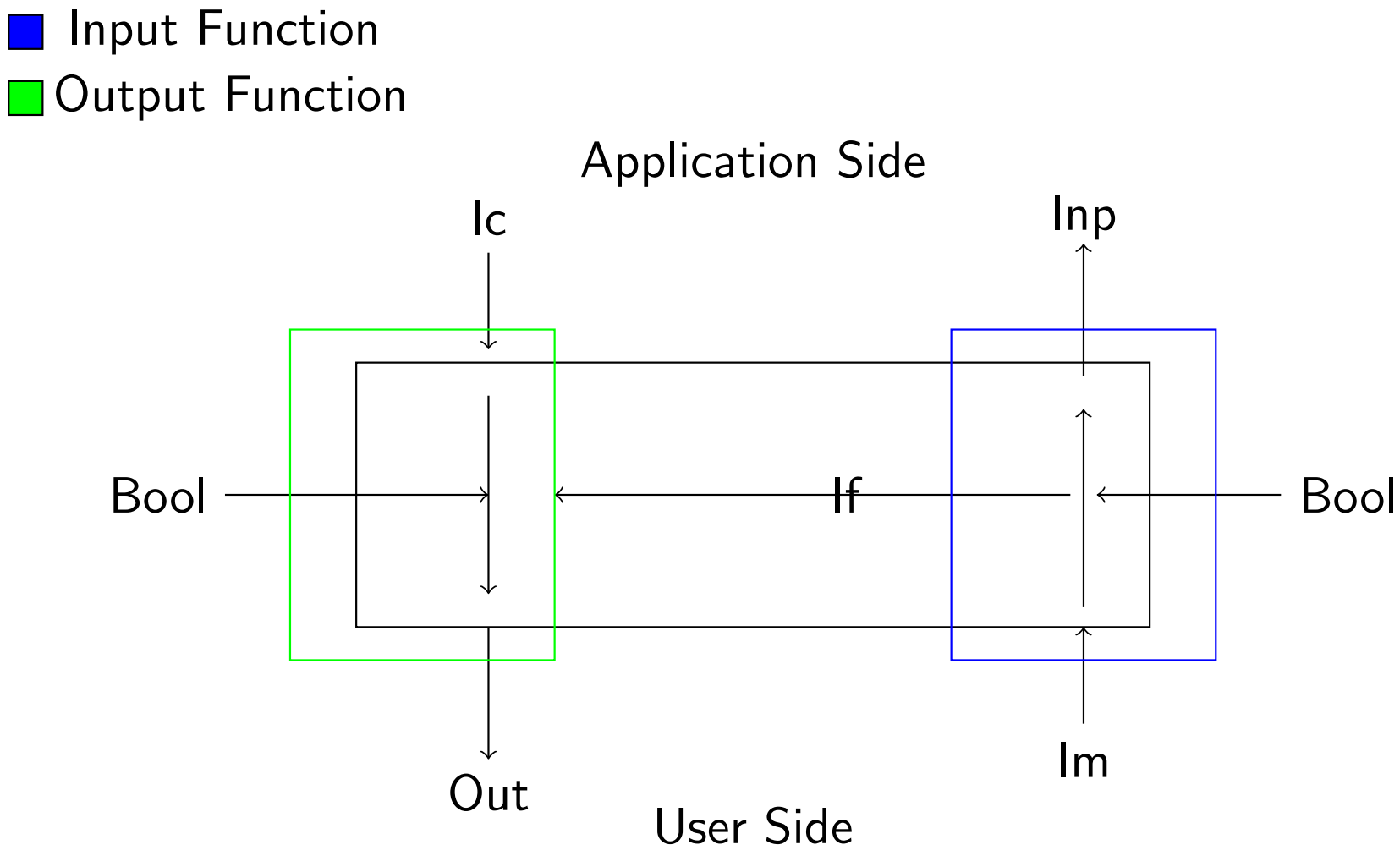


How it works?

- Input Function



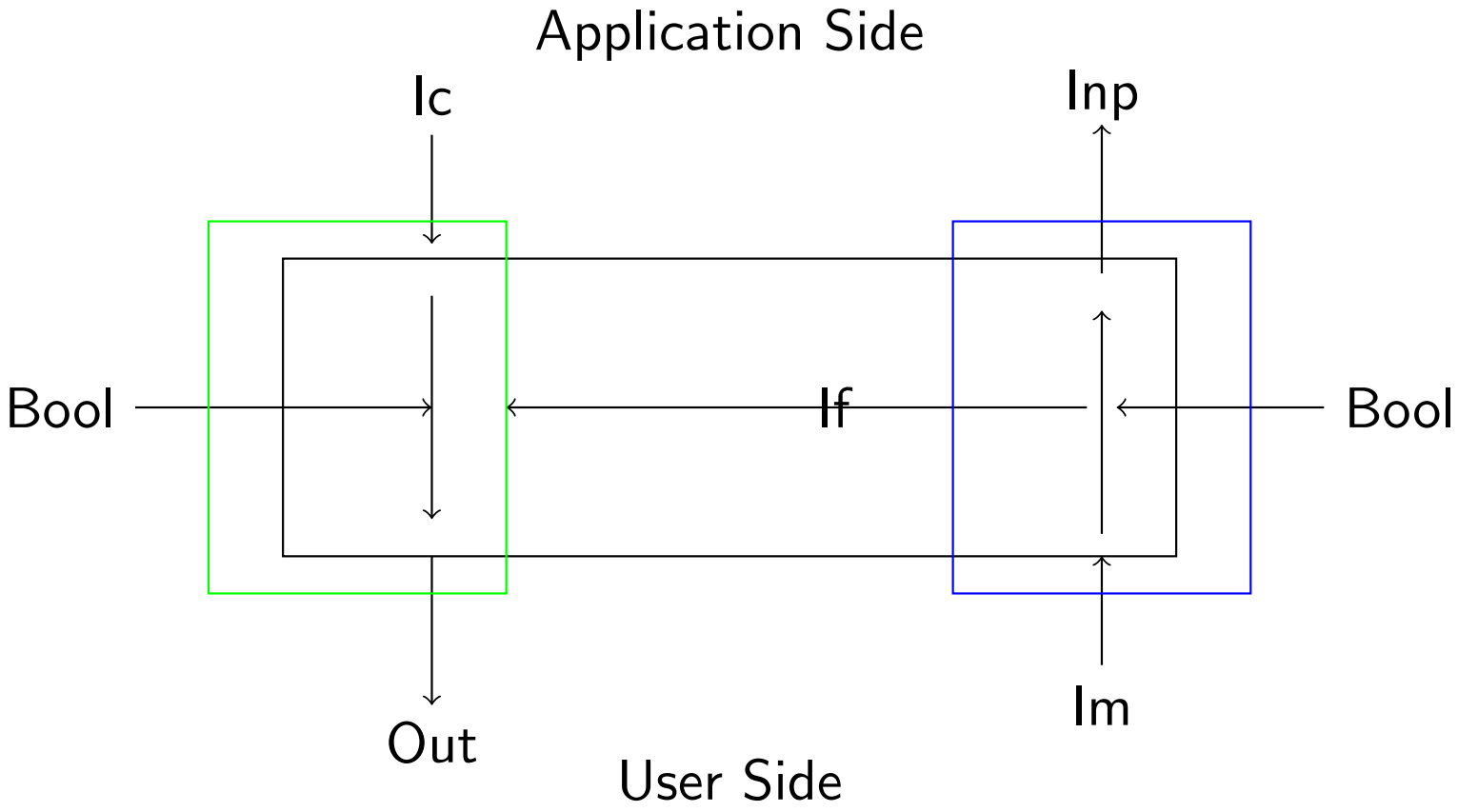
How it works?


Input Function
Output Function
Application Side
Ic
Inp
Bool
If
Bool
Out
Im
User Side

How it works?

Input Function

Output Function



Definition of an Interactor

Now, we can finally define an Interactor mathematically.

An Interactor is a pair of functions

$$I = (FI, FO)$$

Where,
FI = Input Function FO = Output Function

Definition of an UIS

So, We can define an UIS as,

An UIS is a composition of of Interactors

$$UIS = I1, I2, I3, ...$$