



## Blackboard Architectural Pattern

College of Engineering and Architecture

SWE202: Introduction to Software Engineering

Course Instructor: Dr. Marwa Altunusi

Team members: Hala Alnadeem, Basma Alhajji, Raghad Saffaf,  
Dinah Aburahmah, Haneen Hamed

November 27, 2022

# Contents

---

- 3- Introduction.
- 3- Blackboard Architectural Pattern.
- 4- Models Illustrate Blackboard Architecture.
- 6- Real Life Examples of Blackboard.
- 8- Benefits of Blackboard.
- 9- References.

## 1. Introduction

The blackboard architectural pattern is a task solving strategy uses separated independent components called knowledge-sources, along with a database called blackboard, and a controller system to achieve effective solution to an ambiguous problem.<sup>(2)</sup> It is widely used in systems include Artificial intelligence and machine learning. For instance, speech recognition is a widely known problem solved using the blackboard pattern. Blackboard pattern supports maintainability and reusability, which makes it a more efficient and valuable pattern.

## 2. Blackboard Architectural Pattern

### 2.1. Definition:

The Blackboard Architectural Pattern considers problems that does not have achievable solution (non-deterministic problems). Blackboard pattern employs different specialized subsystems, each of which provide partial solution, to eventually deliver an approximate solution to the problem.<sup>(1)</sup>

### 2.2. When is it used?

The Blackboard Architectural Pattern is adopted for large systems that do not have an ultimate solution. This behavioral design pattern is frequently used to create frameworks for dynamic systems that involve artificial intelligence or machine learning, speech recognition, and computer games. <sup>(3)</sup>

### 2.3. Structure & Components:

The Blackboard pattern consists of three main components:

#### 1- Blackboard:

Is a central data storage that manages data. It operates as an interface for knowledge-sources (2<sup>nd</sup> component) to read from or write to it.<sup>(1)</sup> Blackboard is a comprehensive available database that represents the current state of the system as it stores inputs, partial solutions and partial results.<sup>(2)</sup>

Class: Blackboard	Collaborators:
Responsibility: -Manages central data	

#### 2- Knowledge-sources:

Are independent sub-systems that exist separately. Each sub-system contributes to find partial solution to the problem and updates the blackboard. Then the results of each knowledge-source are assembled

Class: Knowledge-source	Collaborators: Blackboard
Responsibility: - Updates Blackboard - Computes a result	

to solve the entire problem. (1) Each knowledge-source is triggered when some conditions are satisfied.(2) Knowledge-sources are scheduled, triggered, and arranged by a controller.(2)

3- Controller:

Is a system that repeatedly keeps track of changes implemented on the Blackboard (database).(1)  
It identifies the suitable knowledge-source to be enabled and executed after analyzing the required changes to enhance the progress of the system state.(2)  
it Arranges the execution of each knowledge source. (1)

Class: Controller	Collaborators: - Blackboard  - Knowledge-source
Responsibility: - monitors blackboard. - schedule knowledge-source execution.	

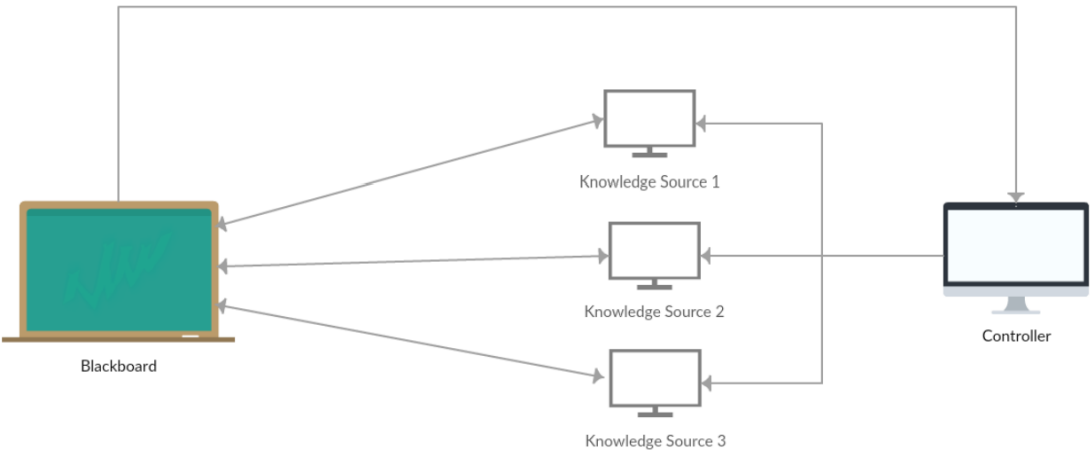
2.4. Why called Blackboard?

“Blackboard” architecture gets its name from the way people work together around a blackboard. Everyone would gather around the board while a problem is written on the board. Each individual starts to independently add a solution he knows that can solve part of the problem. This process is then repeated until a comprehensive solution is found.(2)

3. Models Illustrate the Blackboard Architecture

The following diagrams represent general blackboard architecture models:

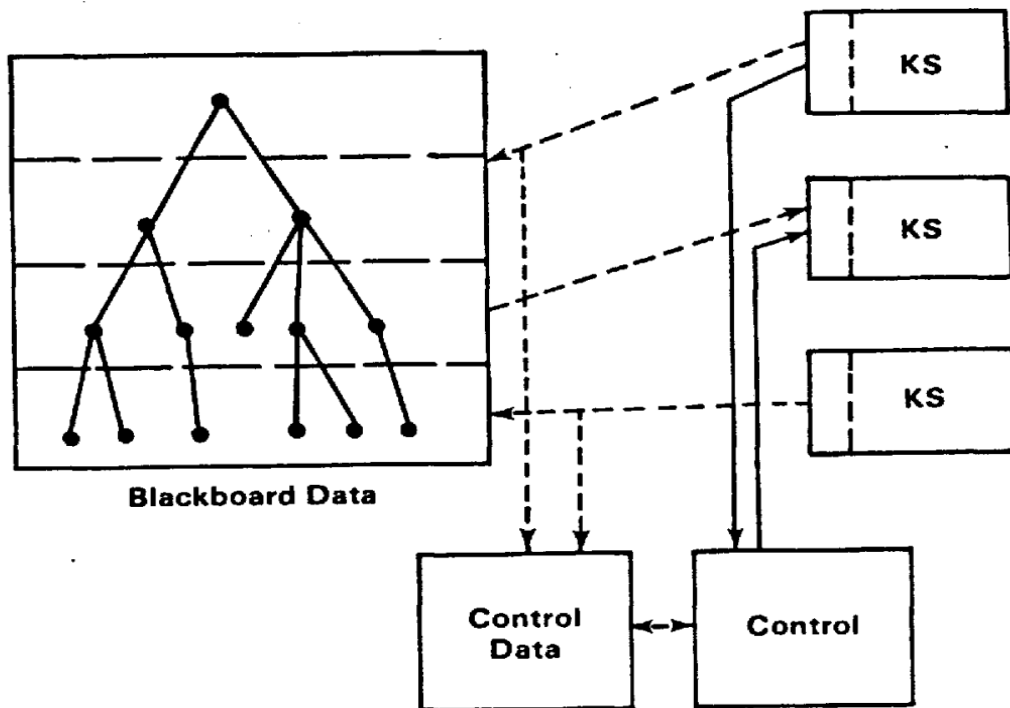
3.1.



The diagram illustrates:

- Three elements: a single controller, one chalkboard, and many knowledge sources.<sup>(2)</sup>
- Direct communication between knowledge sources is not possible. A blackboard is the primary means of communication.<sup>(2)</sup>
- Knowledge sources can only modify the blackboard.<sup>(2)</sup>
- A controller can only read from a blackboard, and based on the state of the blackboard, it will select the next best knowledge source to use.<sup>(2)</sup>
- Access to Blackboard is available to both controllers and knowledge sources.<sup>(2)</sup>

### 3.2.



The diagram illustrates:

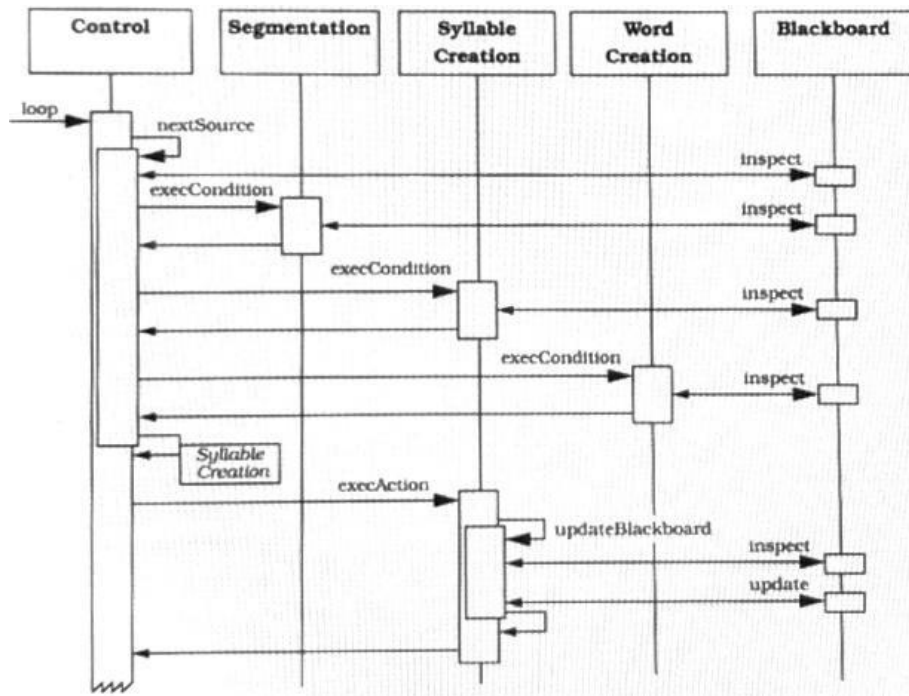
- The information on the blackboard is arranged hierarchically.
- The knowledge sources are self-selecting, logically independent modules.
- Changes to the blackboard may only be made by the knowledge sources.
- A control module chooses which knowledge source to use next and then starts it.

## 4. Real life examples of Blackboard

### 4.1. Speech Recognition

The ability of a machine or program to recognize words spoken aloud and translate them into legible text is known as *speech recognition*, often known as speech-to-text. Basic speech recognition software can only pick out and identify phrases and vocabulary that are pronounced clearly, using computer algorithms to translate spoken speech into text after processing and interpretation. A computer program converts the audio that a microphone records into text that both computers and people can understand. More advanced software can handle diverse languages, accents, and natural speech<sup>(7)</sup>. For instance, one way separates the waveform into parts that make sense in the context of speech, such as phones and another step examine the candidate phrases syntax at the other side of the processing sequence, both techniques are effective in various fields. What defines this problem are the ambiguities of spoken language, noisy data, and the unique characteristics of speakers, such as vocabulary, and pronunciation<sup>(6)</sup>.

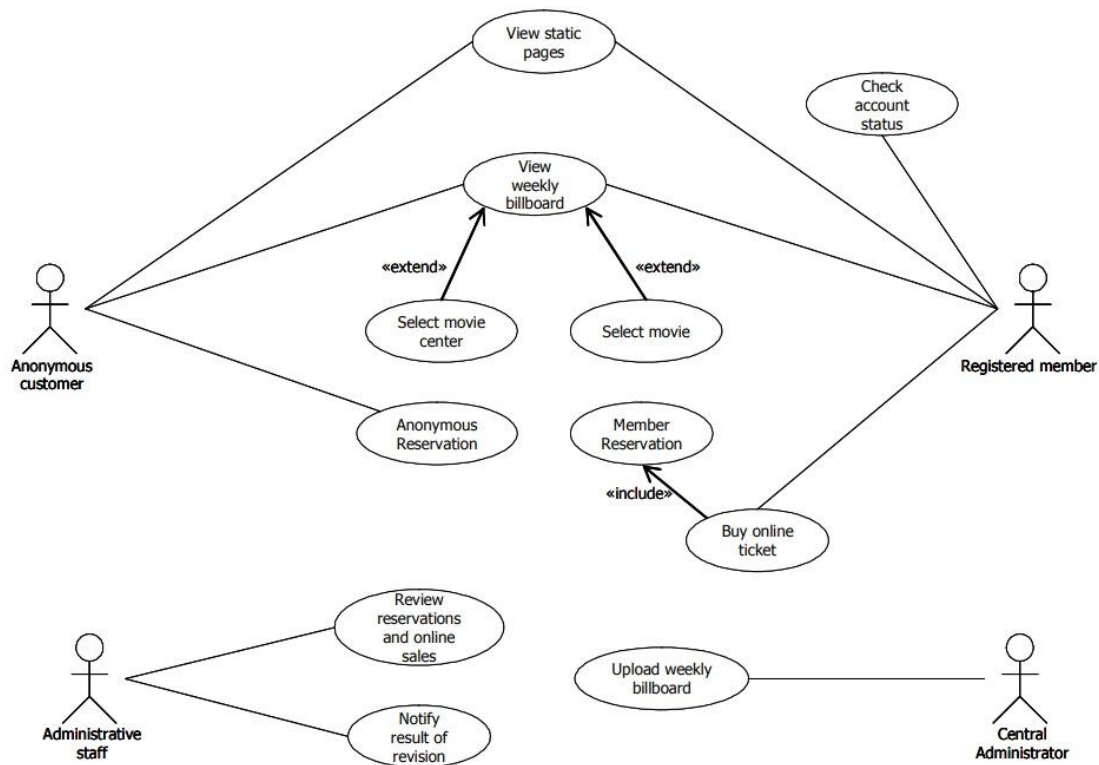
Blackboard architectural pattern was an appropriate solution to organize the system architecture and solve this problem by using a set of independent programs(knowledge-sources) cooperating on the same data structure(blackboard), each program operates individually of each other and designed to handle a specific aspect of the entire task. Also, there is a central control component(controller) which play the role of evaluating the current situation and coordinates the specialized programs and to share results, programs communicate using a Blackboard data structure<sup>(6)</sup>.



## 4.2. Web Applications

Although Blackboard was defined for solving heuristic and artificial intelligence problems, its flexibility allows adaptation to web applications.

For instance, Development of web applications in the field of movie chains. It develops web components for an integrated movie management system that contains ticket sales, accounting, and administration processes in movie chains. The web component is intended to provide web customers an improved services for showing information, ticket purchases, and online reservations. Reserved shows and exhaustive information about the movie can be found on the Billboard page. Tickets can be purchased by credit card or debit against a member account set and paid for during registration.<sup>(8)(9)</sup>



## 5. Benefits of blackboard:

### 5.1. Maintainability & Changeability:

Because knowledge sources, control components and blackboards are separated, it is easy to change/maintain.<sup>(1)</sup>

### 5.2. Reusability:

Knowledge sources are not restricted to a particular issue, They are intendent solutions that can be integrated with large-scale problems.<sup>(1)</sup>

### 5.3. Experimentation:

It is a beneficial pattern/strategy in fields that do not have deterministic solutions.<sup>(1)</sup>

### 5.4. It is advantageous when problems can be divided into subproblems.<sup>(4)</sup>

### 5.5. When there are multiple sources of input data, this method is appropriate.<sup>(5)</sup>



## 5.6. Concurrency:

Knowledge sources are independent of each other, which allows to work in parallel.<sup>(8)</sup>

### References

- (1) Aishwarya, G. (2016, May 4). Blackboard Architecture Pattern. [Blackboard architecture pattern \(slideshare.net\)](#).
- (2) Sheng, E., Lon, L., Yu, H., & Norman, Y. (2018, February 12). Blackboard Architecture. [Microsoft Word - 375481-119565 - Complete-Group 2 - Elgin Wei Sheng Lee - Feb 12, 2018 716 PM - Blackboard.docx \(uwaterloo.ca\)](#).
- (3) Understanding the Blackboard Pattern. [Understanding the blackboard pattern | Hands-On Design Patterns with Java \(packtpub.com\)](#).
- (4) Long, E., Nguyen, B., Pitropov, M., & Torres, E. Blackboard Architecture Style. [Blackboard.pdf \(uwaterloo.ca\)](#).
- (5) Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P. & Stal, M. (1996). Pattern-Oriented Software Architecture: A System of Patterns. John Wiley & Sons. [\\*Blackboard Architecture \(concordia.ca\)](#).
- (6) Bulka, A. (2004, August 5). Blackboard Architectural Pattern. [Blackboard Pattern | Andy Bulka \(abulka.github.io\)](#).
- (7) Lutkevich, B. (2021, September). Speech Recognition. [What is Speech Recognition? \(techtargget.com\)](#).
- (8) Ngan, N., Hieu, N., & Hung D. (2010, November 17). Blackboard Architecture. [blackboard architecture \(slideshare.net\)](#).
- (9) Metzner, C., Cortez, L., & Chacin, D. (2005, January). Using a Blackboard Architecture in a Web Application. [Microsoft Word - I58f73Metz.doc \(informingscience.org\)](#).