# THE MAIN LOGICAL DIFFERENCE BETWEEN THE AGENTIC APPROACH, THE OBJECT-ORIENTED APPROACH, AND THE BUSINESS PROCESS APPROACH

By KORAY YITMEN, Turkey


**KORAY YITMEN**
Turkish Testing Board, Turkey


Koray Yitmen is the president of Turkish Testing Board, a member of International Software Testing Qualifications Board (ISTQB), the member of ISTQB general assembly, the president of User Experience Professionals' Association (UXPA) Istanbul chapter, and the board member of the International Institute of Business Analysis (IIBA) Istanbul chapter.

As a business partner, Koray has led many multi-national, cross-functional teams in the banking, telecommunications, insurance, media, and information technology (IT) industries for more than 25 years. Challenged by the diversity of his projects' cultural, domain, and people issues, Koray has succeeded by putting the needs of his projects' target users first.

In 2000, he started his career as a technology consultant at Accenture (formerly Arthur Andersen). Having worked at Accenture, he has been the business partner of Keytorc Software Testing Services, BA-Works Business Analysis Services, UXservices UX Design Company, Continium Enterprise DevOps Services, and Explori Employee Experience Design Company since 2005. He has a B.S. in Computer Science and an MBA in Entrepreneurship. He is a strong advocate of the business motto "Quality is not tested, but built" and tries to implement this motto in every IT project in which he is involved. His greatest happiness of life is giving back to the community.

His published books are "Bilgi Teknolojilerinin Başarısı için Yazılım Testi – İş Analizi – Kullanılabilirlik", "Business Analysis, Software Testing, Usability : A Quick Guide Book for Better Project Management and Faster IT Career", and "Yazılım Test Teknikleri".

## QUESTION

What is the main logical difference between the agentic approach, the object-oriented approach, and the business process approach?

a) Their approach to functional requirements

b) Their approach to non-functional requirements

c) Their approach to system requirements

d) Their approach to user requirements

e) There is no difference

## THE AGENTIC APPROACH, THE OBJECT-ORIENTED APPROACH, AND THE BUSINESS PROCESS APPROACH

All of the agentic, the object-oriented, and business process approaches are methodologies used to understand the context of the business problem or opportunity. In the business process approach the system is viewed as a collection of processes where data and processes are handled separately. Flowchart, workflow, and entity-relationship diagrams are the most common diagramming types used in this approach. In the object-oriented approach the system is viewed as a collection of objects where data and processes are handled together. Use case, activity, and class diagrams are the most common diagramming types used in this approach.

The following ATM case study will outline the main differences between the business process approach and the object-oriented approach.

In this case study, which describes the analysis and design of an ATM, the requirements listed below are modeled by using both the business process and object-oriented approaches:

- Business Requirement
- User Requirements
- Functional Requirements
- System Requirements

## BUSINESS REQUIREMENT

In both approaches, the business requirement can be defined as follows:

*„To provide automated banking services to customers without the customer having to make a trip to the bank."*

## USER REQUIREMENTS

User requirements are modeled as follows in both approaches:

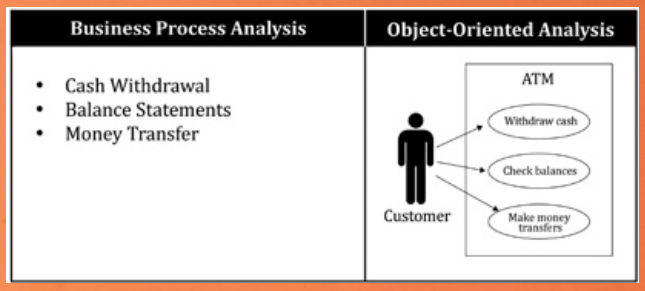


*Figure 1: Modeling user requirements in business analysis and object-oriented analysis*

As seen in the figure above, while use case diagrams are used in object-oriented analysis, simple language is used to describe the user requirements in business process analysis. Another difference between the two approaches is that the object-oriented approach is more user-centric. For instance, in object-oriented analysis, "withdraw cash" is used instead of "cash withdrawal". Those two differences might be thought as the main differences between the business process approach and the object-oriented approach, but they are not; they are only the symbolic differences between the two approaches.

## FUNCTIONAL REQUIREMENTS

Now, the "cash withdrawal" user requirement will be analyzed in order to show the difference between business process analysis and object-oriented analysis in terms of functional requirements. Functional requirements are the steps to be executed in order to achieve a specific goal. While business process analysis uses flowcharts to model the functional requirements, object-oriented analysis uses activity diagrams, as shown in the following:

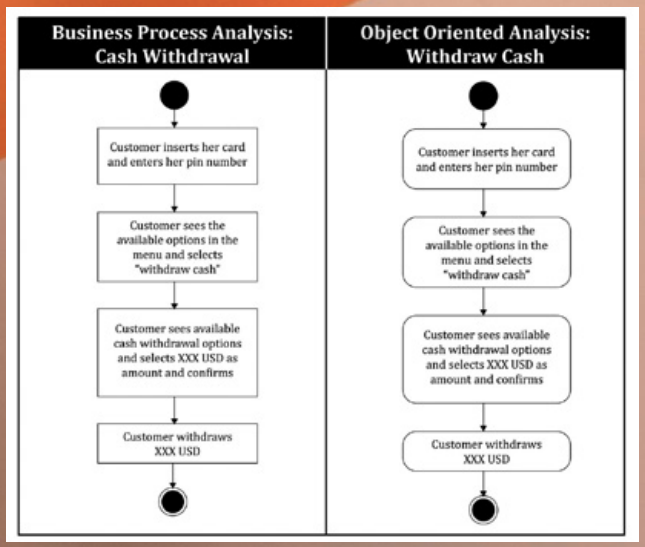


*Figure 2: Modeling functional requirements in business process analysis and object-oriented analysis*

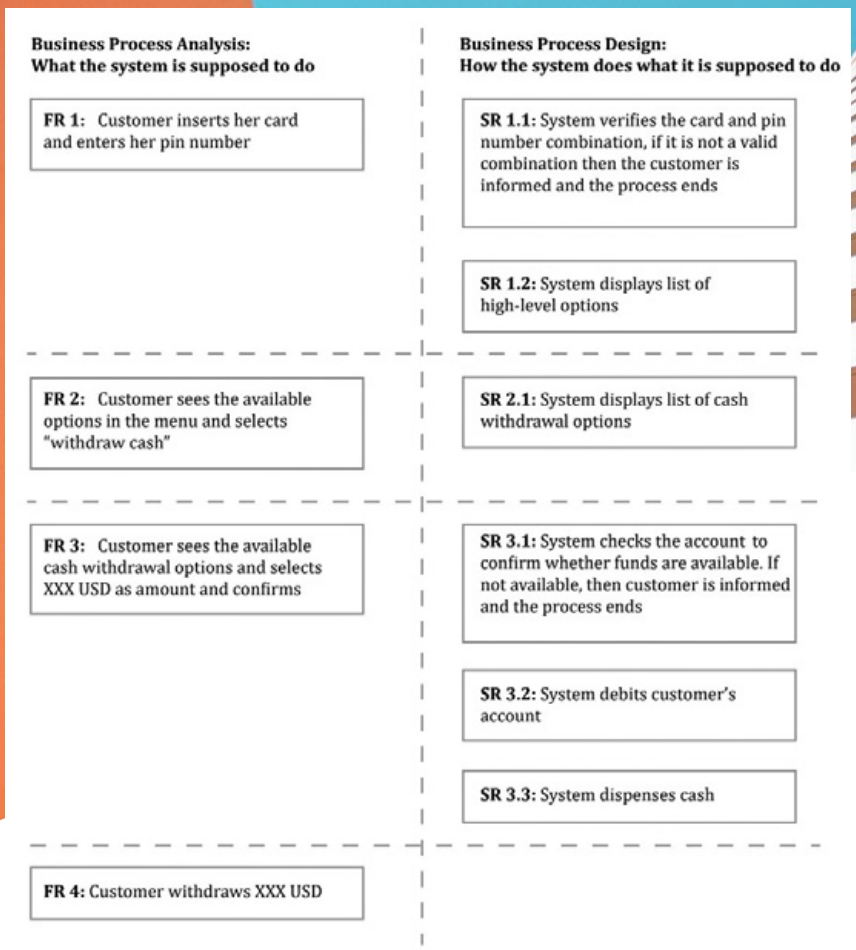


*Figure 3: Business process analysis and design*

So far, both approaches seem almost identical, except for the minor differences between the symbols used (e.g., while the functional requirement is represented by a rectangle in the business process approach, a rounded rectangle is used in the object-oriented approach).

## SYSTEM REQUIREMENTS

While designing system requirements, the focus should be shifted from "what the system is supposed to do" to "how the system does what it is supposed to do", indicating a shift from the analysis phase to the design phase. The following examples show how analysis and design are integrated in each approach:

As shown in the above table, in business process analysis and design, there are one or more corresponding system actions for each user action. Those system actions are then converted into system functions, and then recursively decomposed into sub-functions until simple elements that can be directly represented using programming languages are obtained. See the following functional decomposition as an example:

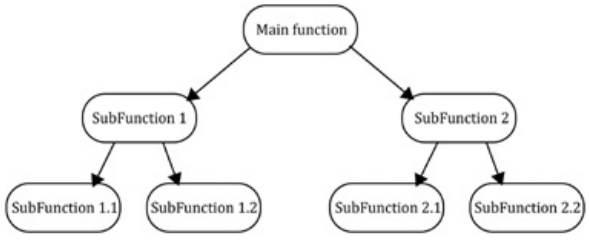


*Figure 4: Functional decomposition of system actions in business process design*

## OBJECT-ORIENTED DESIGN

Contrary to business process design, object-oriented design views the system as a collection of objects. In the object-oriented design approach, functions are represented as types of collaborations between the objects that comprise the system as shown in the following example design of a simple car system:

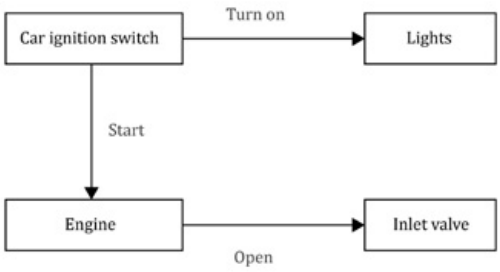


*Figure 5: Modeling of a car system in object-oriented design*

In the object-oriented approach, objects are built from classes. A class is an abstraction that statically captures the behavior of objects. A class describes the state of objects in terms of attributes (data), and behavior in terms of methods (functions and procedures). A class can be regarded as the DNA of the corresponding object. Different from the business process design approach, the object-oriented design approach handles data and functions together in one class.

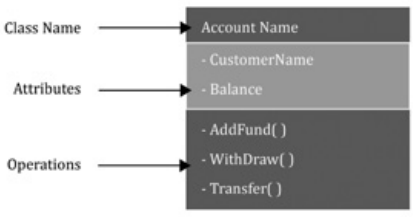


*Figure 6: Class diagram example*

Since the object-oriented approach models the design by using objects instead of functions, the modeling of the design is completely different from that of the business process approach.

## OBJECT-ORIENTED ANALYSIS AND DESIGN

The ATM case study further illustrates how design is integrated with analysis in an object-oriented approach. Assume that for the cash withdrawal user requirement, the system requires the classes below in order to implement customer actions:

- ATM
- Bank
- Account
- Customer

By using the above classes, the system can be designed and integrated with analysis as follows:

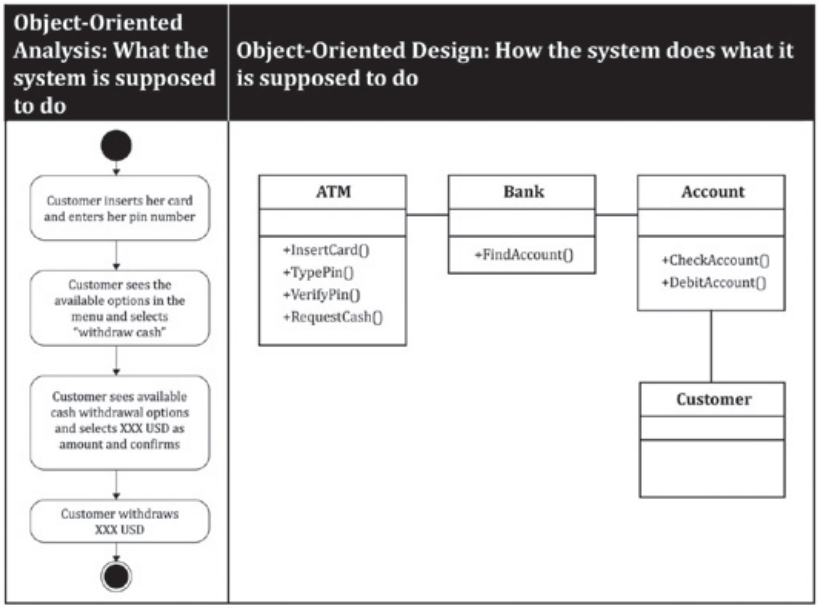


*Figure 7: Object-oriented analysis and design*

As seen from the above model, it is not very clear how the integration between analysis and design is achieved. The answer lies within another diagram (called a sequence diagram) of the object-oriented approach. Sequence diagrams are temporal representations of objects and their interactions to describe one path (scenario) through a use case. Sequence diagrams establish the connection between the process flow and the objects – thus, the connection between analysis and design. For example, the sequence diagram for the cash withdrawal user requirement is as follows: /Figure 8/

The sequence diagram, clearly illustrates the integration of process flow with objects, as well as the integration of analysis and design.

So, the main difference between the business process approach and the object-oriented approach is their approach to the system requirements and thus design.

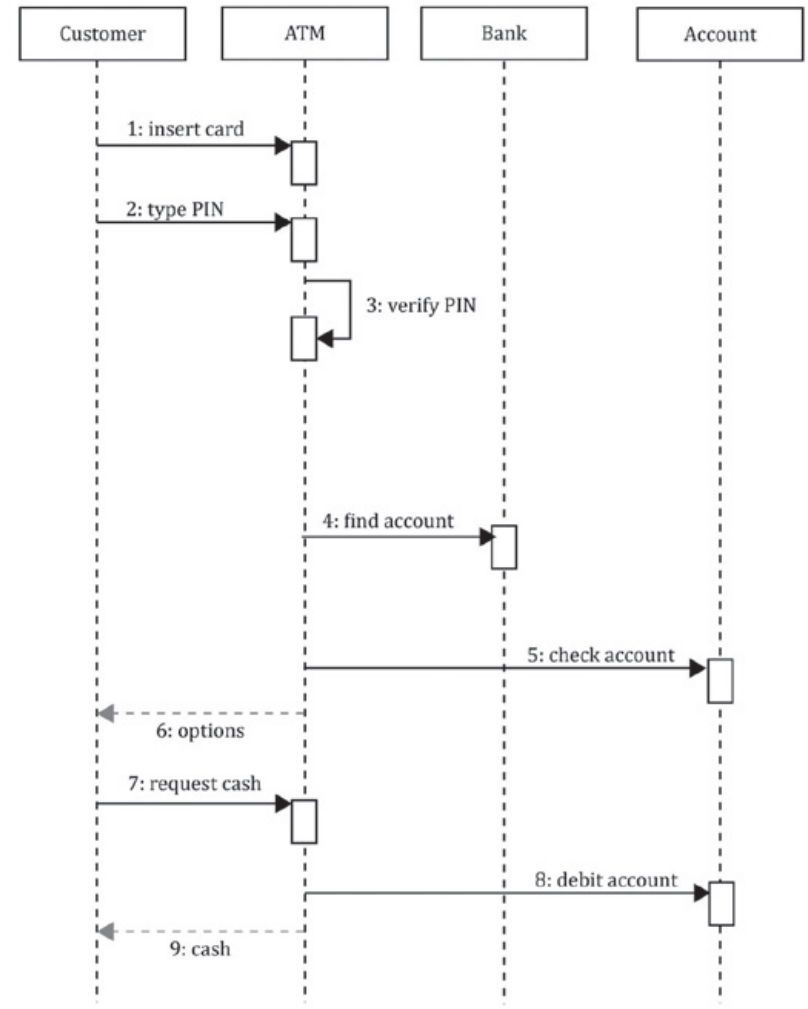


*Figure 8: Sequence diagram for the cash withdrawal user requirement*

## THE DIFFERENCE BETWEEN THE OBJECT-ORIENTED APPROACH AND THE AGENTIC APPROACH

Now, let's discuss the unique properties of the agentic approach. While the object-oriented and agentic approaches share similarities, the agentic approach differs due to its autonomous nature. Below is an illustration of the agentic approach applied to the system requirements of an ATM case study: /Figure 9/

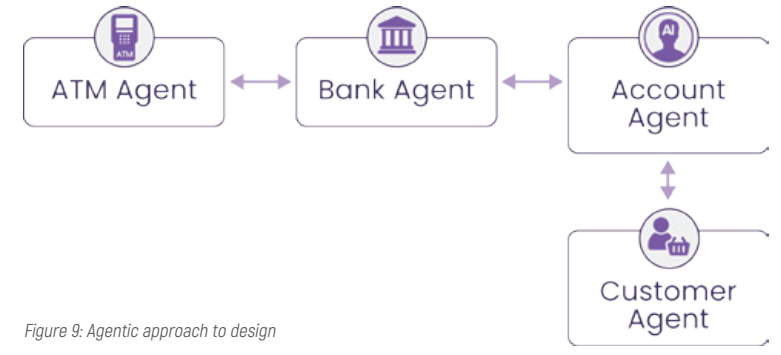


*Figure 9: Agentic approach to design*

Agents function similarly to objects in the object-oriented approach but are not strictly confined to predefined definitions. Instead, their roles, goals, backstories, and guardrails are established, allowing them to act autonomously. The following is a comparison between the object-oriented and agentic approaches: /Figure 10/

| Object-oriented Approach | Agentic Approach |
|---|---|
| **Encapsulation:** Data (attributes) and behaviors (methods) are bundled into objects. | **Autonomy:** Agents act independently without direct external control. |
| **Inheritance:** Objects can inherit properties and behaviors from other objects (classes). | **Goal-Oriented:** Agents are designed to achieve specific objectives. |
| **Polymorphism:** Objects can take on different forms, allowing for flexible code reuse. | **Perception & Action:** Agents interact with their environment through sensors and actuators. |
| **Message Passing:** Objects communicate with each other by sending messages (method calls). | **Decision-Making:** Uses AI techniques such as rule-based reasoning, machine learning, or reinforcement learning. |
| **State-Oriented:** Objects maintain state (data) that changes over time. | **Multi-Agent Interaction:** Agents communicate and collaborate in distributed systems. |

*Figure 10: Comparison between object-oriented approach and the agentic approach*

Here are their key differences: /Figure 11/

| Feature | Object-Oriented Approach | Agentic Approach |
|---|---|---|
| **Core Concept** | Objects encapsulating data & behavior | Autonomous agents making decisions |
| **Control** | Explicit method calls and state changes | Autonomous agents making decisions |
| **Goal** | Code reuse, modularity, and maintainability | Achieving intelligent behavior and adaptability |
| **Communication** | Objects interact via method calls | Achieving intelligent behavior and adaptability |
| **State Management** | Objects maintain state but rely on external calls | Agents make decisions based on internal and environmental state |

*Figure 11: Key differences between object-oriented approach and agentic approach*

In conclusion, the key difference between the business process approach, the object-oriented approach, and the agentic approach lies in how they address system requirements and, consequently, system design. Therefore, when designing intelligent process automation (IPA), this should be kept in mind.