

Architectural Thinking for Intelligent Systems

Assignment 10

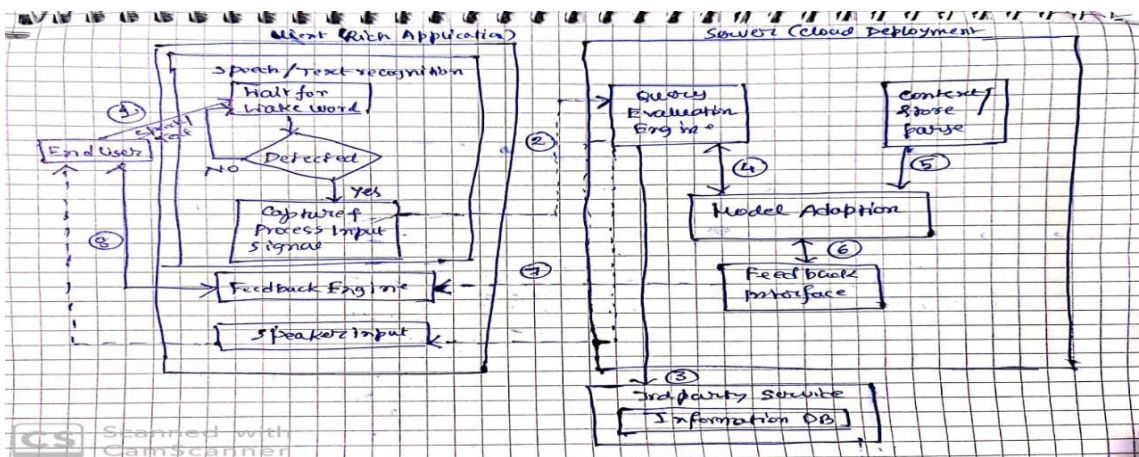
Team 11

1. Think about the couplings that you need internally in your system and to connect the system to its environment. What are the key quality attributes these couplings need to meet?

The key quality attributes that the couplings in our system needs to meet are:

- Performance/Efficiency
 - Latency
 - Bandwidth
- Reliability/Availability
 - Fault Tolerance
 - Recoverability
- Maintainability
 - Modularity
 - Analyzability
 - Testability
- Interoperability (with external components)
- Modifiability
- Portability
 - Adaptability
 - Replicability

2. Use the views you created in Assignment 9 and annotate them with the EAI pattern(s) that is best suited to achieve the desired coupling quality attribute. Give an argument why a specific EAI pattern is the best choice. Base the argument among others on properties of the data and the desired communication quality that you need to achieve.



[Above figure shows the View for Business Owner's viewpoint]

In this view we will take remote procedure call and messaging as the EAI patterns which are best suited as per the system.

1. Here the data will flow from end users to speech/text recognition component including the wake words in case of speech and normal text in case of test mode. So, messaging pattern will be appropriate here.
2. Speech/text recognition component then will send the data to query evaluation through the message bus asynchronously. Here also messaging can be implemented as appropriate pattern.
3. After evaluating the data send by speech/text recognition component, any information from 3rd party services can be obtained according to the necessity of the query evaluation engine. Therefore this 3rd party component will be invoked as remote procedure invocation.
4. For annotations 4, 5, 6 & 7, Query evaluation engine, Message store/parser, Model adaption and Feedback engine can access data internally by putting the message in the message bus by implementing message pattern
8. Feedback engine invokes the GUI driver in the Operating system in the external environment. Simultaneously, the data is sent to the driver space. Now, the GUI driver triggers the display panel of the device and prints the content on the screen. At this moment, user enters his/her input which is then collected and sent back to the Feedback Interface synchronously using the same pattern

3. Is there a difference in the EAI pattern selected for the communication between the internal system components and those for the external interfaces with the environmental systems? Why or why not?

Yes, we have selected different EAI patterns for communication between the internal system components and those for the external interfaces:

- For communication between Internal components: **Messaging Pattern**
- For communication with external interfaces:

Client Side	Server Side
Messaging	Remote Procedure Invocation

Reason:

As we have selected Client-Server architectural style to implement our system, we delegate all the computationally intensive tasks to the server and just handle input/output through a rich client in Android/IOS. Because of this reason, we need **Remote Procedure Invocation** pattern to send user input collected in the app to the server and invoke the response function of the intelligent system. Now, internally the system uses **Message** pattern to handle all the operations among the modules such as Query Evaluation engine, ML Model

Adaption, Context Parser etc. With respect to the user query received from the client. In brief, The query and it's corresponding request is sent to the server, where we need Remote Procedure Invocation to communicate between server's dedicated application port (ex: Port 555) & the Query Evaluation engine interface. The query is usually sent by the client to the Server's public/open port and this request is transferred from the server environment to the intelligent assistant application. After the successful invocation, our intelligent assistant system processes the query, trains the ML model and then return the output to the server interface (external environment), which is then forwarded back to the rich client through TCP/IP and HTTPS.

Whereas, we use Message pattern for communication between internal components. Let's understand this with an example: The Query Evaluation engine, parses the query and sends the final result to the Message bus synchronously and these <query, results> pairs are taken by the Model Adaption component to train the learning model and vice versa.

Thus it is evident that, **it would be very tough/inefficient** to use same EAI patterns for communication between internal components & external interfaces with environmental systems.

4. Think about a potential further evolution of the system. Do you see a possible change in the EAI pattern between selected components or to the environment?

Yes, perhaps if we switch to "**Edge Computing**" architectural style, we can switch from **Remote Procedure Invocation** to **Message (Asynchronous)** pattern in the external interfaces with environment to reduce latency issues, improve reliability & compatibility with legacy platforms (i.e. problem of legacy interfaces) and finally promote loose coupling between the Intelligent Assistant application and the environment (RHEL, Windows - server or Docker Container).