

# Introducing Cloud CBR: A Cloud Based CBR Framework

Chamath Palihawadana<sup>1,\*</sup>, Ikechukwu Nkisi-Orji<sup>1</sup>, Nirmalie Wiratunga<sup>1</sup>,  
David Corsar<sup>1</sup> and Anjana Wijekoon<sup>1</sup>

<sup>1</sup>Robert Gordon University, Aberdeen, Scotland

## Abstract

CBR applications have been deployed in a wide range of sectors, from pharmaceuticals; to defence and aerospace to IoT and transportation, to poetry and music generation; for example. However, a majority of applications have been built using monolithic architectures which impose size and complexity constraints. Such applications have a barrier to adopting new technologies and remain prohibitively expensive in both time and cost because changes in frameworks or languages affect the application directly. To address this challenge, we introduce a distributed and highly scalable generic CBR framework, CLOUD, which is based on a microservices architecture. Microservices architecture splits the application into a set of smaller, interconnected services that scale to meet varying demands. Experimental results show that our CLOUD implementation retrieves cases at a fairly consistent rate as the casebase grows by several orders of magnitude and was over 3,700 times faster than a comparable monolithic CBR system when retrieving from half a million cases. Microservices are cloud-native architectures and with the rapid increase in cloud-computing adoption, it is timely for the CBR community to have access to such a framework.

Video Link: <https://youtu.be/CkuehJPEQyI>

## 1. Cloud CBR System Overview

Introduced in [1], CLOUD is the first cloud-based generic CBR framework developed for scalability using a microservices-oriented design. Prior to CLOUD, CBR frameworks were based on monolithic architecture which is restrictive for applications that require high scalability. CLOUD opened up a new avenue in the CBR landscape by enabling CBR applications (including case-based recommenders) to reach higher levels of scale. An empirical study showed that CLOUD can scale extensively without compromising performance (e.g. retrieval from a casebase of half a million cases was over 3,700 times faster than jCOLIBRI). Further, limited integration support (e.g. APIs) in the previous CBR frameworks has been an obstacle for adopting CBR for some real-world applications. The extensible and open source nature of CLOUD has facilitated its improvement and enhancement over time. Implementation using the microservices paradigm makes CLOUD more sustainable considering the rapid adoption of cloud computing and serverless computing, particularly in enterprise settings.

A complete CLOUD based implementation consists of the following:


---


ICCBR CBR Demos'22: Workshop on CBR Demos and Showcases. ICCBR-2022, September, 2022, Nancy, France

\*Corresponding author.

✉ c.palihawadana@rgu.ac.uk (C. Palihawadana)

🆔 0000-0002-9594-8683 (C. Palihawadana)

 © 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

- **Casebase** contains the cases of a CBR application and is implemented using a full-text search engine (e.g. Elasticsearch, OpenSearch).
- **CBR Functions** consists of the CBR cycle operations (retrieve, reuse, revise, retain), which were implemented using Serverless functions.
- **Similarity Functions** provides the mechanism to match and rank cases during case retrieval. The microservices based architecture in CLOUD is advantageous for extending similarity functions.
- **External Access** consists of a set of HTTP API endpoints that trigger the CBR functions. This external interface is implemented using cloud-native microservices that is available on most cloud service providers (e.g. AWS API gateway, Google API Gateway).
- **Data sources** are the persistence tools where applications store their data (e.g. MySQL, MongoDB) and these data sources can be synchronised with the Casebase through the external access APIs.
- **Client applications** are front facing components of any system such as the CLOUD dashboard. Client applications on any platform can use the external APIs to achieve CBR integration.

As an alternative to deployment on cloud-based services, an instance of CLOUD can be deployed in a containerised environment using docker. Docker is a software platform that allows you to build, test, and deploy applications quickly. The use of containerised deployment simplifies development and testing when using the CLOUD framework.

## 2. Cloud CBR Demonstration

The video presents a complete system overview of CLOUD and demonstrates how to create a sample project using the CLOUD dashboard. The sample project is based on a sample casebase that uses a pizza dataset and can determine the price of pizza.

The following parts are covered in the video demonstration.

- **New Project:** Create a new CBR application/project.
- **Configuration:** Each attribute of the case representation is configured and setup with local similarity metric and attribute weight.
- **Data Import:** CSV based sample cases are pre-loaded to the casebase.
- **Retrieval:** Demonstrates how to perform case retrieval and change attribute weight parameters at retrieve time.
- **Retain:** A recommended or revised case is retained in the casebase.

More information on CLOUD can be obtained by visiting [cloudcbr.com](https://cloudcbr.com) and project source code is available at <https://github.com/rgu-computing/cloud>

## References

- [1] I. Nkisi-Orji, N. Wiratunga, C. Palihawadana, J. A. Recio-García, D. Corsar, Cloud CBR: Towards microservices oriented case-based reasoning, in: International Conference on Case-Based Reasoning, Springer, 2020, pp. 129–143.