Search-backed applications
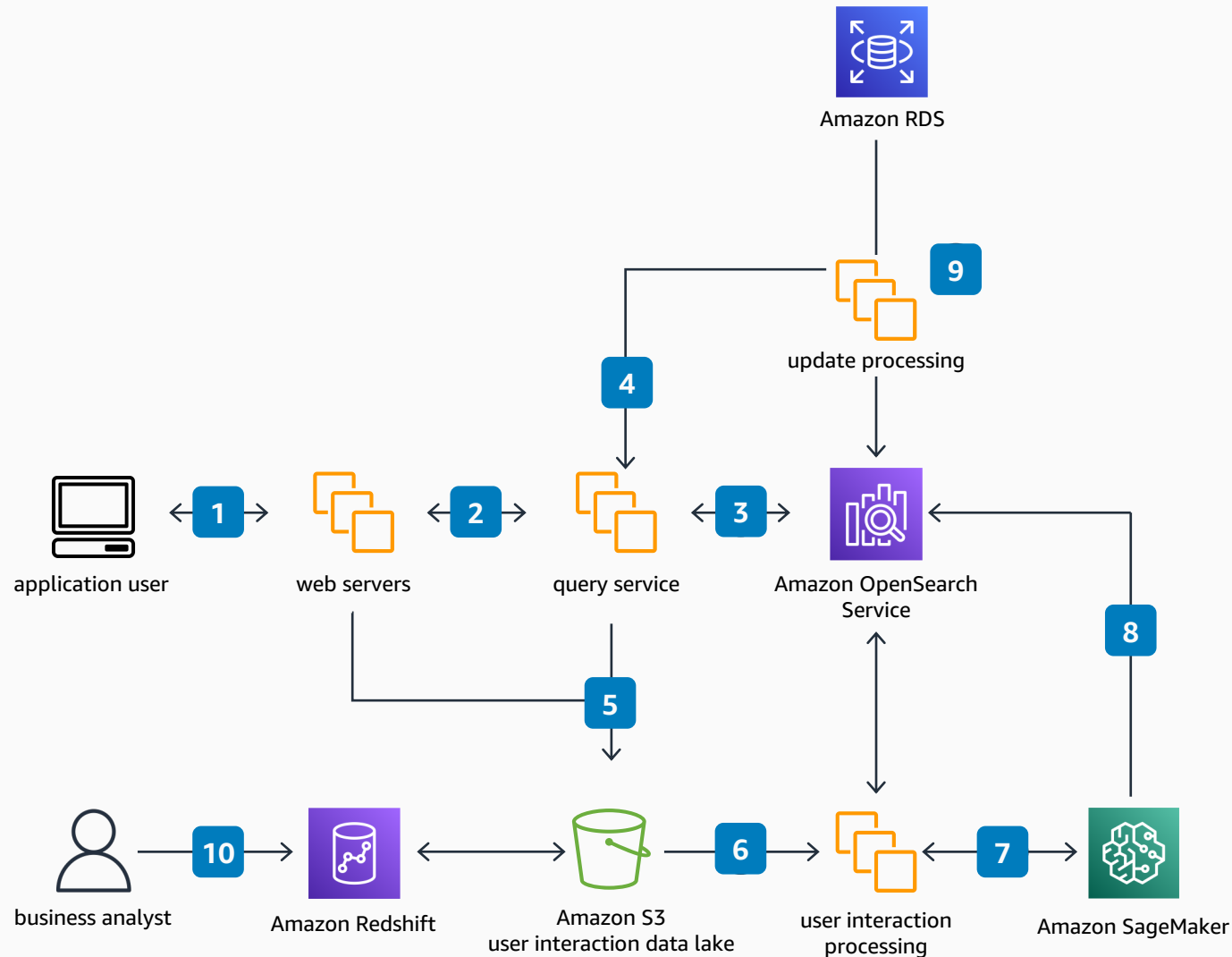
This reference architecture outlines the process to add or improve the search for an existing application.

AWS Reference Architecture

1 An application user sends a query.

2 The web servers deliver to the query service. At this point, the query service can employ machine learning (ML) models using **Amazon SageMaker** (arrow not shown) for user segmentation, concept and entity extraction, query-to-click, and other data to enrich the query.

3 The query service enriches or rewrites the query, based on user segmentation from **Amazon SageMaker** (arrow not shown), user preferences from **Amazon Relational Database Service** (Amazon RDS), and past query performance. It sends the augmented query to **Amazon OpenSearch Service**.

4 The user sends only searchable data to **Amazon OpenSearch Service**, employing a relational or NoSQL system as the system of record. The query service retrieves only keys in the search results. It retrieves the full record information from the system of record.

5 The web servers and query service send user interaction data back to an **Amazon Simple Storage Service** (Amazon S3) data lake or **Amazon Redshift**.

6 An offline process pulls user interaction from the data lake.

7 The offline process takes data (such as clicks) that it needs to augment the records in the catalog, and updates models in **Amazon SageMaker**.

8 Records are updated in **Amazon OpenSearch Service** as needed.

9 Either the web servers are sending catalog updates to **Amazon OpenSearch Service**, or the user is running a change data capture to bring those updates to **Amazon OpenSearch Service**. (There could also be a separate inventory or other system that holds data for result enrichment.)

10 Business analysts generate reporting, KPIs, and so on, from the processed user interactions.