Chapter 4: Patterns (Part 2)

# Lab: Apply the Schema Versioning Pattern

Back to the Question

## Correct Options

Scenario A & B both describe situations for which the schema versioning pattern is designed.

- Scenario A

  Since your team has a strict **zero downtime** transition requirement, it is essential to minimize the volume of writes throughout the upgrade. As a result, applying the Schema Versioning Pattern is the right solution.

- Scenario B

  Even though the modifications to the schema are significant, it is likely that you can implement them without downtime using the Schema Versioning Pattern. Having additional information in the updated documents will probably translate to a new function that will make fewer requests to the database to gather what it needs. For the documents that were migrated, the application will call this new function. For the unmigrated documents, your application can still call the old function.

## Incorrect Option

- Scenario C

  Since both applications can handle both schemas, there is no need to use the schema versioning pattern. However, there is still a need to somehow distinguish one document shape from another. In this case, applying the Polymorphic Pattern, which will be covered later is the more appropriate approach.

  Once you decide to come up with a new schema that could be common to both current schemas, then you can use the schema versioning pattern to make the transition.

Proceed to next section