ApplicationContext

If you have multiple application contexts and you want to access them using the `ApplicationContextAware` interface, you need to make sure that the beans implementing this interface are associated with the respective application contexts you want to access.

Here's a general approach:

1. Implement the `ApplicationContextAware` interface in your bean class(es) where you want to access the application context.

2. Ensure that each application context is associated with the appropriate bean(s) by registering them accordingly.

3. In the `setApplicationContext()` method provided by the `ApplicationContextAware` interface, store the reference to the application context for later use.

Here's an example:

```java
import org.springframework.context.ApplicationContext;
import org.springframework.context.ApplicationContextAware;
import org.springframework.stereotype.Component;

@Component
public class MyBean implements ApplicationContextAware {

    private static ApplicationContext context;

    @Override
    public void setApplicationContext(ApplicationContext applicationContext) {
        context = applicationContext;
    }

    // Method to access the application context
    public static ApplicationContext getApplicationContext() {
        return context;
    }
}
```

In this example, `MyBean` implements `ApplicationContextAware`, allowing it to access the application context. When the application context is set, it's stored in the `context` variable. You can then access this application context statically using the `getApplicationContext()` method.

You would need to repeat this process for each application context and associated bean(s) if you have multiple application contexts in your application. Each bean should store the reference to its corresponding application context for later use.