Configure JPA

To configure the JPA vendor, you typically need to specify the `jpaVendorAdapter` bean along with the `entityManagerFactory`. Here's how you can configure it along with the other JPA properties using Java-based configuration:

```java
import org.springframework.orm.jpa.vendor.HibernateJpaVendorAdapter;

@Configuration
@EnableTransactionManagement
public class JpaConfig {

    // DataSource and other beans...

    @Bean
    public LocalContainerEntityManagerFactoryBean entityManagerFactory() {
        LocalContainerEntityManagerFactoryBean em = new
LocalContainerEntityManagerFactoryBean();
        em.setDataSource(dataSource());
        em.setPackagesToScan("com.example.entity.package");

        // Set JPA vendor adapter
        em.setJpaVendorAdapter(jpaVendorAdapter());

        // Set JPA properties
        Properties jpaProperties = new Properties();
        jpaProperties.setProperty("hibernate.dialect",
"org.hibernate.dialect.MySQLDialect");
        jpaProperties.setProperty("hibernate.show_sql", "true");
        // Add other JPA properties as needed
        em.setJpaProperties(jpaProperties);

        return em;
    }

    @Bean
    public JpaVendorAdapter jpaVendorAdapter() {
        HibernateJpaVendorAdapter jpaVendorAdapter = new
HibernateJpaVendorAdapter();
        // Set Hibernate properties
        jpaVendorAdapter.setGenerateDdl(true); // Generate DDL schema scripts
        jpaVendorAdapter.setShowSql(true); // Show SQL statements
        // Add other Hibernate properties as needed
        return jpaVendorAdapter;
    }
```

```
    // TransactionManager and other beans...
}
```

In this configuration:

– The `jpaVendorAdapter` bean is created using `HibernateJpaVendorAdapter`, which is an implementation of `JpaVendorAdapter` for Hibernate.
– The `entityManagerFactory` bean is configured with the `jpaVendorAdapter` using the `setJpaVendorAdapter` method.
– Hibernate properties specific to the JPA vendor (such as `generateDdl` and `showSql`) are set using the `HibernateJpaVendorAdapter`.
– Other JPA properties (such as `hibernate.dialect` and `hibernate.show_sql`) are set separately using a `Properties` object, as shown in the previous example.