

Fusion Registry — Client installation readiness (Spring 7)

Short guide for technical teams preparing to install or upgrade to the **Spring Framework 7**–based Fusion Registry. For dependency detail and migration notes, see [docs/client-facing/README.md](https://docs.spring.io/spring-framework/docs/current/README.html) (Spring 7 client technical brief).

1. What you must provide

Area	Requirement
Java	JDK 17 or newer (LTS such as 21 recommended). JDK 8 and 11 are not sufficient for this release line.
Application server	A servlet container that implements Jakarta Servlet 6.1 . Apache Tomcat 10+ is the typical choice. Tomcat 9.x (and other servers stuck on <code>javax.*</code> / older Servlet APIs) must be upgraded or replaced before deployment.
JDBC drivers	Vendor JDBC drivers compatible with Java 17+ , matching your database major version. Align driver versions with your DBA — older drivers packaged for JDK 8 often cause runtime issues on newer JDKs.

2. Tomcat upgrade (high level)

The Spring 7 stack uses the **Jakarta EE 11/Servlet 6.1** APIs (`jakarta.servlet.*`), rather than the legacy Java EE `javax.servlet.*` .

1. Plan migration from **Tomcat 9** → **Tomcat 10+** (or another **Servlet 6.1**–certified server).
2. Verify **JVM options** (`JAVA_HOME` , memory, encoding, timezone) on the new instance.
3. Deploy only the **new** Registry WAR; do not mix **old** Spring 5 libraries on the shared Tomcat classpath.
4. Re-test **TLS**, **reverse proxy** (paths, headers, WebSockets if used), and **JNDI** datasource configuration if applicable.

3. Database connectors (JDBC)

Supported database platforms are those your contract and release notes specify (commonly **MariaDB**, **Oracle**, **Microsoft SQL Server**). For installation:

1. Use a **current JDBC driver** from the database vendor, certified for **JDK 17+** and your **supported** database server version.
2. Install the driver where your deployment model requires it (`$CATALINA_BASE/lib` for Tomcat-wide use, or follow your vendor runbook).
3. **Do not** downgrade the driver to an old “JRE 8” artefact unless your support team explicitly approves it — classpath and TLS/cipher requirements often break on modern JDKs.
4. Confirm **connection URLs**, **schemas**, **collation/charset**, and **pool sizing** after the upgrade; Hibernate **7.x** / JPA **3.2** ride the same datasource configuration as in the technical brief.

4. Checklist — environment ready before installing the new Registry

Use this before first start in each environment (dev → test → prod).

Java and server

- ☐ **JDK 17+** installed; `java -version` matches the planned production JDK.
- ☐ **Tomcat 10+** (or equivalent **Servlet 6.1** server) installed and hardened (user, file permissions, logging).
- ☐ **Tomcat service** configured with sufficient **heap** and **metaspace**; garbage collector settings reviewed if you had custom tuning on JDK 8/11.
- ☐ **File system** paths for logs, temp, and Registry properties exist and are writable by the Tomcat user.

Database and JDBC

- ☐ **Database version** supported for this Registry release (confirm with release notes / your vendor).
- ☐ **JDBC driver** upgraded and tested against **JDK 17+**.
- ☐ **Backup and restore** procedure verified; a **rollback** snapshot exists before migration.
- ☐ **Dedicated schema/user** permissions unchanged or re-applied after any DBA maintenance.
- ☐ **Character set / encoding** aligned (e.g. UTF-8) end-to-end.

Security and networking

- ☐ **TLS certificates** valid on proxies and connectors; cipher suites acceptable to your security policy.
- ☐ **Firewall** rules allow app → DB, and clients → Registry as today.
- ☐ **LDAP / SSO / OAuth** (if used) endpoints and trust stores updated if URLs or certs changed during infra moves.

Application configuration

- ☐ `RegistryProperties` (or equivalent) points to the correct DB, LDAP, and paths for this environment.
- ☐ **Secrets** not embedded in old formats incompatible with the new image or systemd/tomcat layout.
- ☐ **Smoke test plan** agreed (login, structure browse, critical data/API flows).

Regression

- ☐ **Staging** run completed on JDK 17+ and Tomcat 10+ with production-like data volume where possible.
- ☐ **Integration partners** notified of cutover window if URLs or TLS fingerprints change.

5. After installation

- Monitor startup logs for `NoClassDefFoundError`, `javax/` packages, or **duplicate Spring/Hibernate** JAR warnings (usually means classpath contamination).
- Keep the [Spring 7 client brief](#) bookmarked for dependency versions (Spring **7.0.x**, Spring Security **7.0.x**, Hibernate **7.2**, Jakarta Servlet **6.1**, JPA **3.2**).