# JAR vs WAR

## 🔎 What is a JAR?

JAR (Java ARchive) is a package file format that bundles:  
✅ Compiled `.class` files  
✅ Configuration files  
✅ Resources (like images, property files, etc.)  
  
In Spring Boot, a JAR file contains an embedded web server (like Tomcat or Jetty) and can be executed directly using:  
java -jar myapp.jar

### ✅ Advantages of JAR

✔️ Self-contained – Includes an embedded server, no need for external configuration.  
✔️ Simplified deployment – Deploy directly as a standalone application.  
✔️ Faster startup – Optimized for microservices and cloud-based deployments.  
✔️ Easier to manage – Can be built once and run anywhere with `java -jar`.

### ❌ Disadvantages of JAR

❌ Less flexibility for deploying in existing application servers.  
❌ Larger file size due to embedded server.

## 🔎 What is a WAR?

WAR (Web Application Archive) is a package format designed for deployment in a servlet container (like Tomcat or JBoss).  
WAR files don’t include the server; they need to be deployed to an external server.

### ✅ Advantages of WAR

✔️ Flexible deployment – Can be deployed in existing servlet containers.  
✔️ Easier to manage – If you already have a centralized server, WAR is easier to manage.  
✔️ Better separation of concerns – Web server and application are separate.

### ❌ Disadvantages of WAR

❌ Requires an external servlet container.  
❌ More complex to deploy and manage compared to JAR files.  
❌ Not suitable for microservices architecture.

## 🔥 Key Differences

|  |  |  |
| --- | --- | --- |
| Feature | JAR | WAR |
| Contains | Application + Embedded Server | Application only |
| Execution | `java -jar` (standalone) | Deploy in external server |
| Startup Time | Faster (embedded server) | Slower (depends on server) |
| Flexibility | Independent deployment | Requires servlet container |
| Best For | Microservices, Cloud apps | Enterprise apps, Monolithic apps |
| Size | Larger (includes server) | Smaller |

## 🏆 When to Use JAR vs WAR

|  |  |
| --- | --- |
| Use Case | Best Choice |
| Microservices | JAR (due to independence and quick startup) |
| Cloud Deployment | JAR |
| Standalone Apps | JAR |
| Monolithic Apps | WAR |
| Existing Application Servers | WAR |
| Traditional Enterprise Deployment | WAR |

## 🚀 Conclusion:

👉 JAR is best for:  
✅ Microservices  
✅ Cloud-based and containerized apps (like Docker)  
✅ Independent and fast deployment  
  
👉 WAR is best for:  
✅ Traditional enterprise applications  
✅ Deploying to centralized or shared servlet containers  
✅ Apps that need to work with existing infrastructure  
  
👉 JAR is more common for Spring Boot projects due to its simplicity and flexibility. ✅