Why Spring is the hot favorite java framework among the developers ?

“Necessity is the mother of invention”, before Spring, application were developed using J2EE standards. J2EE was marginally good, however there were many problemds in it, such as

1) code became cumbersive as application scalled.

2) performance of the system got compromised due to the heaviness of the applications. Etc.

Because… Spring is lightweight :

Spring is very lightweight because of its POJO (Plain Old Java Objects) implementation, which doesn't force it to inherit any class or implement any interface.

Because… Spring follows AOP :

Aspect-Oriented Programming is used for separating cross-cutting concerns (ie. logging, security, transaction management, etc.) from the actuall business logic of the application.

Because… Spring supports Dependency Injection :

Spring supports Dependency Injection, this means Spring allows us to develop loosely coupled applications. Therefore, the unit testing of these loosely coupled applications becomes easier.

Because… Spring aims to promote code reuse :

Spring avoids the need of making some important hard decisions up front, like whether our application will ever use JTA or JNDI; Spring abstractions will allow us to deploy our code in a different environment if we ever need to.

Because... Spring aims to facilitate good programming practice, such as programming to interfaces, rather than classes :

Use of an IoC container such as Spring greatly reduces the complexity of coding to interfaces, rather than classes, by elegantly concealing the specification of the desired implementation class and satisfying its configuration requirements.

Because… Spring promotes pluggability :

Spring encourages us to think of application objects as named services. Ideally, the dependencies between such services are expressed in terms of interfaces. Thus we can swap one service for another without impacting the rest of our application.

Because... Spring is designed so that applications using it are as easy as possible to test :

As far as possible, application objects will be POJOs, and POJOs are easy to test; dependency on Spring APIs will normally be in the form of interfaces that are easy to stub or mock.

Because... Spring promotes architectural choice :

While Spring provides an architectural backbone, Spring aims to facilitate replaceability or each layer. For example, with a Spring middle tier, we should be able to switch from one O/R mapping framework to another with minimal impact on business logic code, or switch from, say, Struts to Spring MVC or WebWork with no impact on the middle tier.

Because... Spring does not reinvent the wheel :

Despite its broad scope, Spring does not introduce its own solution in areas such as O/R mapping where there are already good solutions exists. Similarly, it does not implement its own logging abstraction, connection pool, distributed transaction coordinator, remoting protocols, or other system services that are already well-served in other products or application servers. However, Spring does make these existing solutions significantly easier to use, and places them in a consistent architctural approach.