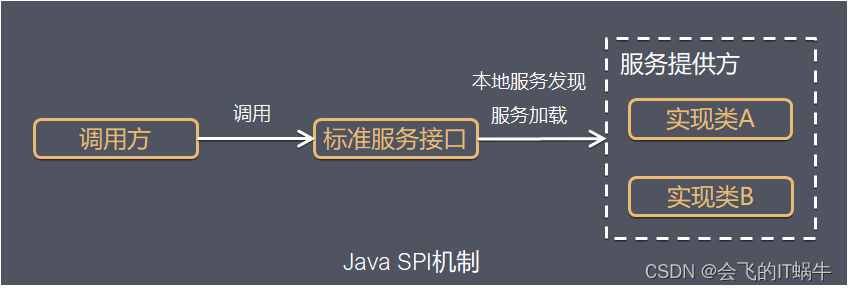
### **synchronized是Java中的关键字，是一种同步锁。它修饰的对象有以下几种：**

　　1. 修饰一个代码块，被修饰的代码块称为同步语句块，其作用的范围是大括号{}括起来的代码，作用的对象是调用这个代码块的对象；   
　　2. 修饰一个方法，被修饰的方法称为同步方法，其作用的范围是整个方法，作用的对象是调用这个方法的对象；   
　　3. 修改一个静态的方法，其作用的范围是整个静态方法，作用的对象是这个类的所有对象；   
　　4. 修改一个类，其作用的范围是synchronized后面括号括起来的部分，作用主的对象是这个类的所有对象。

**SPI机制是什么？**

* spi全称为 (Service Provider Interface)，是JDK内置的一种服务提供发现机制。SPI是一种动态替换发现的机制，一种解耦非常优秀的思想。
* spi的工作原理： 就是ClassPath路径下的META-INF/services文件夹中， 以接口的全限定名来命名文件名，文件里面写该接口的实现。然后再资源加载的方式，读取文件的内容(接口实现的全限定名)， 然后再去加载类。
* spi可以很灵活的让接口和实现分离， 让api提供者只提供接口， 第三方来实现。



优点：

使用Java SPI机制的优势是实现解耦，使得第三方服务模块的装配控制的逻辑与调用者的业务代码分离，而不是耦合在一起。应用程序可以根据实际业务情况启用框架扩展或替换框架组件。

缺点：

虽然ServiceLoader也算是使用的延迟加载，但是基本只能通过遍历全部获取，也就是接口的实现类全部加载并实例化一遍。如果你并不想用某些实现类，它也被加载并实例化了，这就造成了浪费。获取某个实现类的方式不够灵活，只能通过Iterator形式获取，不能根据某个参数来获取对应的实现类。

多个并发多线程使用ServiceLoader类的实例是不安全的。