**Introduction of Java**

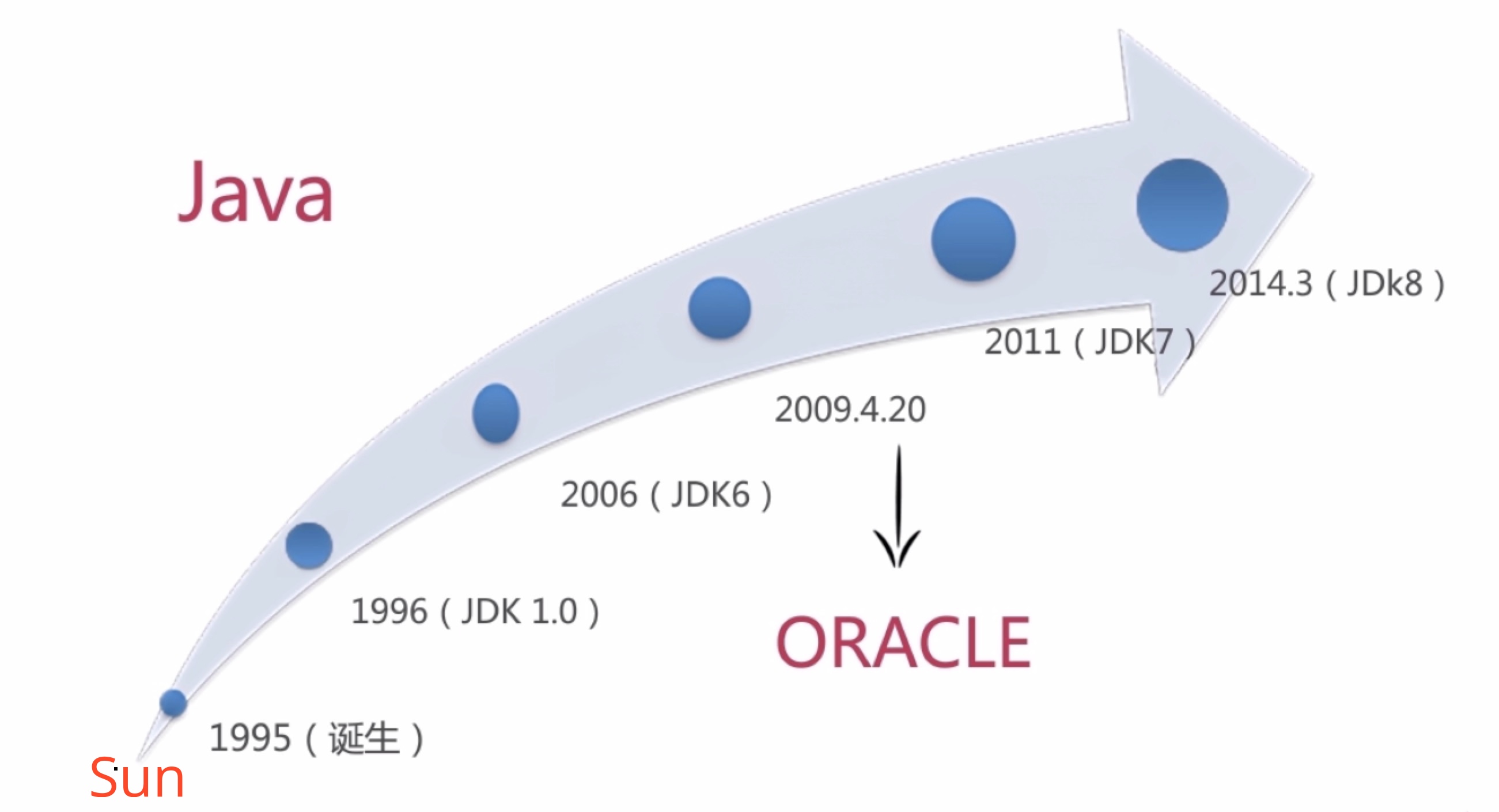
Java refers to the dynamic and object oriented programming language and platform developed and launched by Sun Microsystems on May of 1995. **Java have three editions**

* JavaSE（Java Platform Standard Edition）
* JavaEE (Java Platform,Enterprise Edition)
* JavaME (JME)(Java Platform Micro Edition).

Focusing on JavaSE and JavaEE, this course will help you master the basic grammars of Java and lay a solid foundation for the future.

**History of Java**

* On May 23rd, 1995, Java was born.
* On Jan. 1st, 1996, the first JDK-JDK1.0 was born.
* On June, 1996,Sun Microsystems launched three editions for Java.
* On May 29th, 2000, JDK1.4 was launched.
* On Feb. 26th , 2002, J2SE1.4 was launched, which marked a significant upgrade on Java's computing capcabilities.
* In June of 2005, JavaOne was held as an annual conference for Java developers and JavaSE 6 was released.
* On April 20tth, 2009, Oracle acquired Sun Microsystems for 7.4 billion US dollars, thus obtained the copyright of Java.
* Up to date the latest Java edition is javaSE 11.



To learn more about the history of Java, please visit [here](http://oracle.com.edgesuite.net/timeline/java/)

**Main Features of Java**

1. **Easy to learn:** Java is a programming language evolved from C++. Besides, Java has taken avay many elements from C++ which are difficult to learn, such as pointer, which makes the learning experience relatively easy.
2. **Object-oriented:** Java is known as an advanced object oriented programming language.Object oriented is its key feature and also the soul of this language.
3. **Cross-platform：** Java is cross platform in the sense that a compiled Java program runs on all platforms and can be migrated between different platforms.
4. **Interpretation：** Java is a language which needs interpretation.The source code written in java is first interpreted into byte code. And then the byte code will be interpreted to machine code through a Java virtual machine before it can be implemented by a computer.
5. **Robust：** java has a robust system which can avoid serious bugs, such as the type checking and exception handling during compling.
6. **Multi-tread：** Multi-threading contributes much to Java's high performance, which enables maximum utilization of CPU.

Above are some of the common features listed in this course. For details on other features, please read the following articles:

* [Reading Notes on Core Java, Volume II--Advanced Features(<https://www.jianshu.com/p/15187621a73c>)
* [Outlines of Java Features](https://www.jianshu.com/p/dcbe1fdc69e5)

**Some basic concepts of Java:**

[**JVM**](https://en.wikipedia.org/wiki/Java_virtual_machine)

Java virtual machine Java virtual machine(JVM) is a mechanism requsite for running Java programme. It is this mechanism that makes Java independent from its platforms, which is also a key feature of Java.JVM can shield the information related with a certain platform. By using the target bytecode(.class) produced by a Java compiler, migration can be done between different platforms without any revisions. To execute a bytecode, a Java virtual machine will first interprete a bytecode into a machine instruction on a certain platform. In this way, Java can be run indepenent of its platforms.JVM is not onlly the key to enable Java to be migrated as a whole between different patforms, but also a security inspection engine for Java programme(conduct security check). In other words, JVM can be regarded as a connection between interpreted Java programme(.class files) and hardware system.

[**JDK**](https://en.wikipedia.org/wiki/Java_Development_Kit)

Java Development Kit(JDK). It is mainly used on Java application programmes of mobile devices and embeded device. Acting as the core in Java development, JDK provides the runtime environment, necessary tools and foundation class libraries for Java. In orther words, as JDK is the SDK facing Java developers, to develop a Java programme you must install JDK. Details on JDK installation will covered later in this course.

[**JRE**](https://en.wikipedia.org/wiki/Java_virtual_machine)

Java Runtime Environment. It provides a collection of environments requsite for Java programme, including JVM Standard Implementation and Java core class libraries. Only when a Java source code is interpreted to bytecode and JVM receives the bytecode together with the JRE, can JVM be executed on a certain paltform. Therefore, JVM is included in JRE and JDK has its own JRE.

Below is an outline of the relationship between the three:

