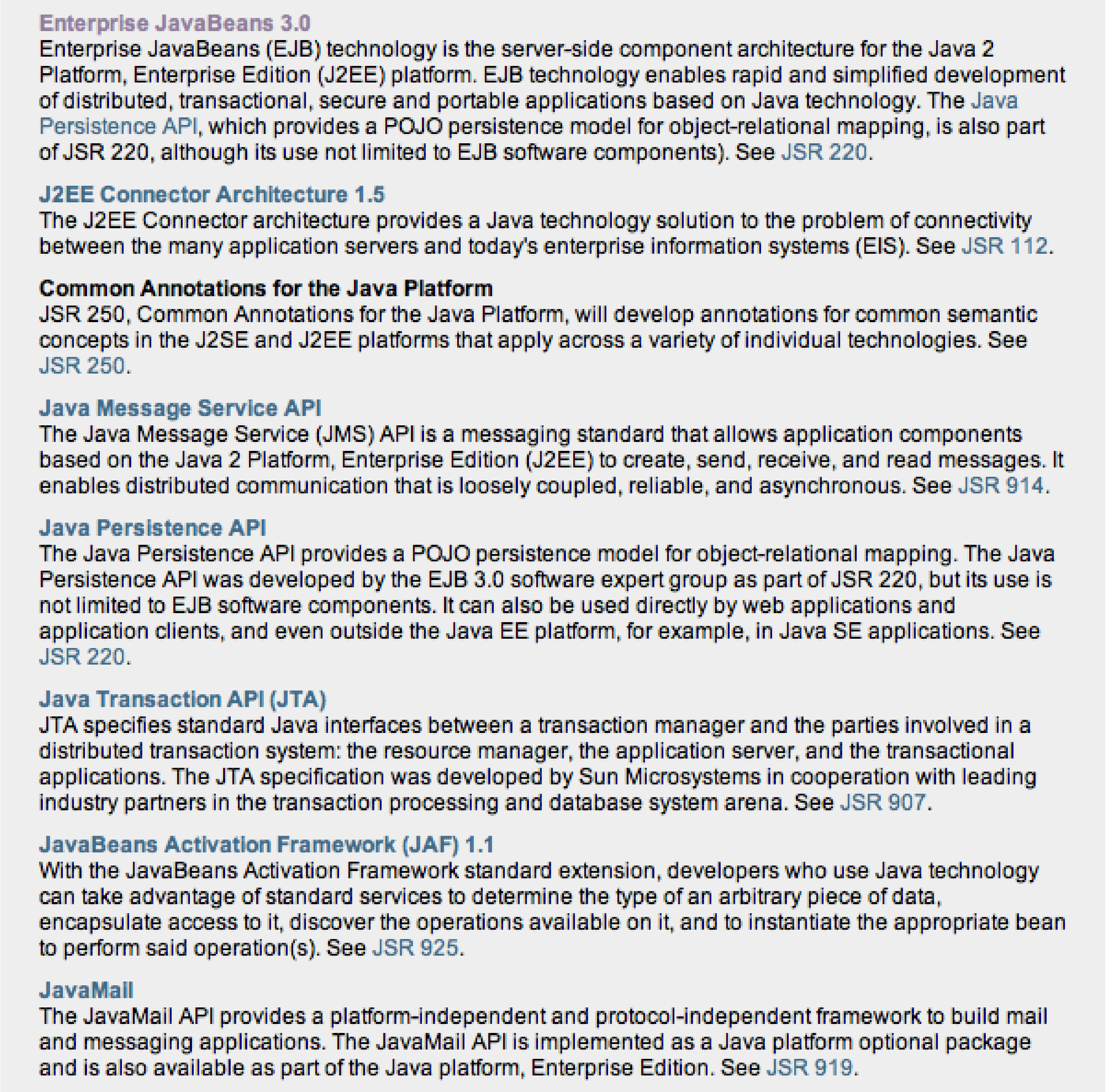

Enterprise JavaBeans 3.0

Enterprise JavaBeans (EJB) technology is the server-side component architecture for the Java 2 Platform, Enterprise Edition (J2EE) platform. EJB technology enables rapid and simplified development of distributed, transactional, secure and portable applications based on Java technology. The Java Persistence API, which provides a POJO persistence model for object-relational mapping, is also part of JSR 220, although its use not limited to EJB software components). See JSR 220.

J2EE Connector Architecture 1.5

The J2EE Connector architecture provides a Java technology solution to the problem of connectivity between the many application servers and today's enterprise information systems (EIS). See JSR 112.

Common Annotations for the Java Platform

JSR 250, Common Annotations for the Java Platform, will develop annotations for common semantic concepts in the J2SE and J2EE platforms that apply across a variety of individual technologies. See JSR 250.

Java Message Service API

The Java Message Service (JMS) API is a messaging standard that allows application components based on the Java 2 Platform, Enterprise Edition (J2EE) to create, send, receive, and read messages. It enables distributed communication that is loosely coupled, reliable, and asynchronous. See JSR 914.

Java Persistence API

The Java Persistence API provides a POJO persistence model for object-relational mapping. The Java Persistence API was developed by the EJB 3.0 software expert group as part of JSR 220, but its use is not limited to EJB software components. It can also be used directly by web applications and application clients, and even outside the Java EE platform, for example, in Java SE applications. See JSR 220.

Java Transaction API (JTA)

JTA specifies standard Java interfaces between a transaction manager and the parties involved in a distributed transaction system: the resource manager, the application server, and the transactional applications. The JTA specification was developed by Sun Microsystems in cooperation with leading industry partners in the transaction processing and database system arena. See JSR 907.

JavaBeans Activation Framework (JAF) 1.1

With the JavaBeans Activation Framework standard extension, developers who use Java technology can take advantage of standard services to determine the type of an arbitrary piece of data, encapsulate access to it, discover the operations available on it, and to instantiate the appropriate bean to perform said operation(s). See JSR 925.

JavaMail

The JavaMail API provides a platform-independent and protocol-independent framework to build mail and messaging applications. The JavaMail API is implemented as a Java platform optional package and is also available as part of the Java platform, Enterprise Edition. See JSR 919.