Software licensing for cloud computing is an enduring problem without a universally accepted solution

at this time. The license management technology is based on the old model of computing centers with

licenses given on the basis of named users or as site licenses. This licensing technology, developed for

a centrally managed environment, cannot accommodate the distributed service infrastructure of cloud

computing or of grid computing.

Only very recently IBM reached an agreement allowing some of its software products to be used on

EC2. Furthermore,MathWorks developed a businessmodel for the use ofMATLABin grid environments

[63]. The Software-as-a-Service (SaaS) deployment model is gaining acceptance because it allows users

to pay only for the services they use.

There is significant pressure to change the traditional software licensing model and find nonhardware-

based solutions for cloud computing. The increased negotiating power of users, coupled with

the increase in software piracy, has renewed interest in alternative schemes such as those proposed by

the SmartLM research project ( www.smartlm.eu). SmartLM license management requires a complex

software infrastructure involving SLA, negotiation protocols, authentication, and other management

functions.

A commercial product based on the ideas developed by this research project is elasticLM, which

provides license and billing for Web-based services [63]. The architecture of the elasticLM license

service has several layers: coallocation, authentication, administration, management, business, and

persistency. The authentication layer authenticates communication between the license service and the

billing service as well as the individual applications; the persistence layer stores the usage records. The

main responsibility of the business layer is to provide the licensing service with the licenses prices, and

the management coordinates various components of the automated billing service.

When a user requests a license from the license service, the terms of the license usage are negotiated

and they are part of an SLA document. The negotiation is based on application-specific templates and

the license cost becomes part of the SLA. The SLA describes all aspects of resource usage, including

the ID of application, duration, number of processors, and guarantees, such as the maximum cost and

deadlines. When multiple negotiation steps are necessary, the WS-Agreement Negotiation protocol

is used.

To understand the complexity of the issues related to software licensing, we point out some of

the difficulties related to authorization. To verify the authorization to use a license, an application

must have the certificate of an authority. This certificate must be available locally to the application

because the application may be executed in an environment with restricted network access. This

opens up the possibility for an administrator to hijack the license mechanism by exchanging the local

certificate.