SOA and REST are the most popular today in distributed integration and the ones we want to compare, between them and against the Structural Services style. The first aspect to realize is that comparing architectural styles in abstract terms is not at all the same thing as comparing their practical instantiations. SOA is easy to grasp, since it constitutes a natural evolution of the object-oriented style, which analysts and programmers are already familiar with. However, Web Services, with WSDL, XML Schema and SOAP, are complex to use, especially without adequate tools that usually automate or hide most of the development process. On the other hand, REST is comparatively harder to master, in particular for people with an object-oriented mindset, who tend to slide to the probably more familiar RPC (Remote Procedure Call) style (Fielding, 2008), with operation parameters instead of dynamic resource references. On the other hand, its use with HTTP is relatively straightforward, even without elaborate tools, since what is involved is essentially plain HTTP messaging.

* SOA appeared in the context of enterprise integration, with the main goal of achieving *interoperability* between existing enterprise applications with the emerging XML-based technologies,