This chapter discusses the concept of reference model as a structure of related Services and Protocols that together form a complex possibly standard Interaction System. These Services and Protocols are only globally defined in terms of their main characteristic or key functions. A reference model then acts as an intermediate specification to overview the total design process for a complex Interaction System based on separation of concerns. The key functions allocated to these Services and Protocols represent these concerns.

The structure of globally defined Services and Protocols can be used to distribute the design load amongst different groups of designers that can work relatively independently of each other to design and specify each of these Services and Protocols completely. The reference model also acts as a means to organise the communication amongst these groups of designers in the case of open questions, lack of clarity or conflicts. We show that the Service concept plays an important role in reference models as a means to stop the proliferation of design concerns amongst the various protocols and to allow flexibility of using various Protocol solutions, both above as well as below the Service boundary.

A reference model is of crucial importance as an intermediate to develop a set of related standards for Services and Protocols. This has been demonstrated with the development of the OSI Reference Model and the Internet Protocol Suite.

The concept of reference model can, mutatis mutandis, also be used for the design of complex systems in general.

### 12.1 Reference Model

The design of an Interaction System by representing it as a structure of a related Services and Protocols, sometimes called a Protocol Suite, can be carried out in two main phases. In the first phase, the functions of the Services and Protocols are not completely elaborated and defined, but only globally in terms of defining their main
characteristics and key functions. In the second phase, these incompletely defined functions are further elaborated and detailed to completely define the related Services and Protocols. In this case we call the design resulting from the first phase a Reference Model (RM), or a Reference Architecture (RA).

The example about the decomposition of a reliable end-to-end Service in Sect. 11.2 gives an example of such a global elaboration.

**Definition 12.1 Reference Model**

A Reference Model of an Interaction System is a structure of related Services and Protocols of which the functions are only globally characterised and defined. In later design stages, these functions can be further detailed and defined to get the complete definitions of these Services and Protocols.

A Reference Model, and its development, serves several purposes. Its main purpose is to provide a common framework of reference for different teams of designers to guide them when further detailing and completely defining the individual Services and Protocols without the need for these teams to interact on a regular basis. This is particularly necessary when the Services and Protocols are so complex that they can only be developed by large groups of designers, working together in different often internationally composed teams. Furthermore, while developing the RM, it also serves as a means to assess the technical and economic feasibility of the Services and Protocols under design, and the planning of their detailed design. Finally, it also serves as a means to communicate the objectives and the progress of the RM design effort amongst interested parties.

The development of a RM can thus be understood as an intermediate design step between the definition of the User requirements for the total Interaction System and the complete definition of each Service and Protocol from which it will be composed. The development of a RM requires high skills and is generally done by a special (international) team of experienced (senior) designers.

To serve its purpose, a RM should act as a stable reference. This means that the detailed elaboration of the Services and Protocols should not conflict with the RM and destabilise it by demanding redesigns of (parts of) the RM, while the detailed elaboration is still going on. It would mashup the design process of the IS totally.

We can also express this requirement by saying that the globally defined RM functions need to appear as correct abstractions of the detailed defined functions in the completely defined Services and Protocols. Vice versa, one can express this by saying that the completely defined Services and Protocols need to appear as correct refinements of the globally defined Services and Protocols. Since during refinements design details are added, and since in refinement one can choose amongst many possible design details, one can also characterise the functions defined by the RM as generic functional definitions demanded by generic functional requirements.

The above-described goal is quite difficult to achieve and can only be achieved by basing the development of the RM itself on the same sound design principles as discussed for the design of Services and Protocols in Chaps. 6–10.