The constituents of the SLang software are called **units and standalone routines**. To keep them and your development organized, it is convenient to group **units and standalone routines** into **clusters**. By combining **units and standalone routines** from one or more clusters, you may build **systems**. The system is a general term for library or program (executable)

These three concepts provide the basis for structuring Slang software:

•A *unit and standalone routine* is a modular (compilation) unit.

•A *cluster* is a logical grouping of compilation units.

•A *system* results from the assembly of one or more compilation units to produce the system.

Of these, only “unit and standalone routine”, describing the basic building blocks, correspond directly to constructs of the language. To build clusters and systems out of compilation units, you will use not a language mechanism, but tools of the supporting environment.

Clusters provide an intermediate level between compilation units and systems, indispensable as soon as your systems grow beyond the trivial:

•At one extreme, a cluster may be a simple group of a few compilation units.

•At the other end, a system as a whole is simply a cluster too.

•In-between, a cluster may be a library consisting of several subclusters, or an existing system that you wish to integrate as a subcluster into a larger system.

Clusters also serve to store and group compilation units using the facilities of the underlying operating system, such as files, folders, and directories.

After the basic definitions, the language description will concentrate on compilation units, indeed the most important concept, which views software construction as an industrial production activity: combining components, not writing one-of-a-kind applications.