1. *Logic-based dialects.* These are meant to include rule languages that are based on some form of logic; for example, first-order logic and various logic programming approaches with different interpretations of negation (answer-set programming, well-founded semantics, etc.). The concrete dialects developed so far under this branch are:

   - *RIF Core*, essentially corresponding to function-free Horn logic; and
   - *RIF Basic Logic Dialect (BLD)*, essentially corresponding to Horn logic with equality.

2. *Rules with actions.* These are meant to include production systems and reactive rules. The concrete dialect developed so far in this branch is:

   - *Production Rule Dialect (RIF-PRD).*

The RIF family was designed to be both *uniform* and *extensible*. Uniformity is achieved by expecting the syntax and semantics of all RIF dialects to share basic principles. Extensibility refers to the possibility of future dialects being developed and added to the RIF family. For the logic-based side, the RIF Working Group proceeded to support uniformity and extensibility by developing the *Framework for Logic Dialects (RIF-FLD)* which allows one to specify various rule languages by instantiating the various parameters of the approach. This framework is a major achievement, but goes beyond the scope of this book. In the following we will present the basic ideas of RIF-BLD.

Before doing so, it should be stated that a lot, if not most, of the work of the RIF Working Group was dedicated to semantic aspects. Of course, rule interchange takes place at the syntactic level (e.g., using XML) using mappings between the various syntactic features of a logic system and RIF. But the main objective is to interchange rules in a *semantics preserving way*.