# Grammar (BNF) of the Specification Language

*specification* ⟶ *spec-list*

*spec-list* ⟶ *spec* **.** *spec-list* | *spec* **.**

*spec* ⟶ *model-decl* | *system-decl* | *problem-decl*

*model-decl* ⟶ *comp-model-decl* | *net-model-decl*

*comp-model-decl* ⟶ **component model id is**

*event-decl*

*input-decl*

*output-decl*

*state-decl*

*transition-decl*

**end id**

*event-decl* ⟶ **event** *id-list* **;**

*id-list* ⟶ **id ,** *id-list* | **id**

*input-decl* ⟶ **input** *id-list* **;** | **ε**

*output-decl* ⟶ **output** *id-list* **;** | **ε**

*state-decl* ⟶ **state** *id-list* **;**

*transition-decl* ⟶ **transition** *trans-decl-list* **;**

*trans-decl-list*⟶ *trans-decl* **,** *trans-decl-list* | *trans-decl*

*trans-decl* ⟶ **id** = *event* **, id -> id , {** *opt-ref-list* **}**

*event* ⟶ *ref* | **( )**

*ref* ⟶ **id ( id )**

*opt-ref-list* ⟶ *ref-list* | **ε**

*ref-list* ⟶ *ref* **,** *ref-list* | *ref*

*net-model-decl* ⟶ **network model id is**

*component-section*

*input-decl*

*output-decl*

*link-section*

*pattern-section*

*initial-section*

*viewer-section*

*ruler-section*

**end id**

*component-section* ⟶ **component** *decl-list* **;**

*decl-list* ⟶ *decl* **,** *decl-list* | *decl*

*decl* ⟶ *id-list* **: id**

*link-section* ⟶ **link** *link-list* **;** | **ε**

*link-list*⟶ *link-decl* **,** *link-list* | *link-decl*

*link-decl* ⟶ *ref* **->** *ref*

*pattern-section* ⟶ **pattern** *pattern-list* **;** | **ε**

*pattern-list*⟶ *pattern-decl* **,** *pattern-list* | *pattern-decl*

*pattern-decl* ⟶ ref *pattern-op expr*

*pattern-op* ⟶ **=** | **==**

*expr* ⟶ *expr* '|' *term* | *term*

*term* ⟶ *term factor* | *term* **&** *factor* | *factor*

*factor* ⟶ *factor* ***** | *factor* **+** | *factor* **?** | **(** *expr* **)** | **~** *ref* | *ref*

*initial-section* ⟶ **initial** *ref-list* **;** | **ε**

*viewer-section* ⟶ **viewer** *map-list* **;** | **ε**

*map-list* → *map-decl* **,** *map-list* | *map-decl*
*map-decl* → *ref* **->** **id**
*ruler-section* → **ruler** *map-list* **;** | **ε**

*system-decl* → **system id is** *system-node-list emergence-section* **end id**
*system-node-list* → *system-node***,** *system-node-list* | *system-node* **;**
*system-node* → **node id: id is** *initial-section viewer-section ruler-section* **end id**
*emergence-section* → **emergence** link-*list* **;** | **ε**

*problem-decl* → **problem id is** *problem-node-list* **end id**
*problem-node-list* → *problem-node***,** *problem-node-list* | *problem-node* **;**
*problem-node* → **node id is** *initial-section viewer-section obs-section ruler-section* **end id**
*obs-section* → **obs [** *opt-id-list* **] ;**
*opt-id-list* → *id-list* | **ε**

Note: A specification file may contain several `#include` directives (not involved in the grammar of the specification language), to be managed by a preprocessor. The (single) file generated by the preprocessor is a phrase of the specification language.