

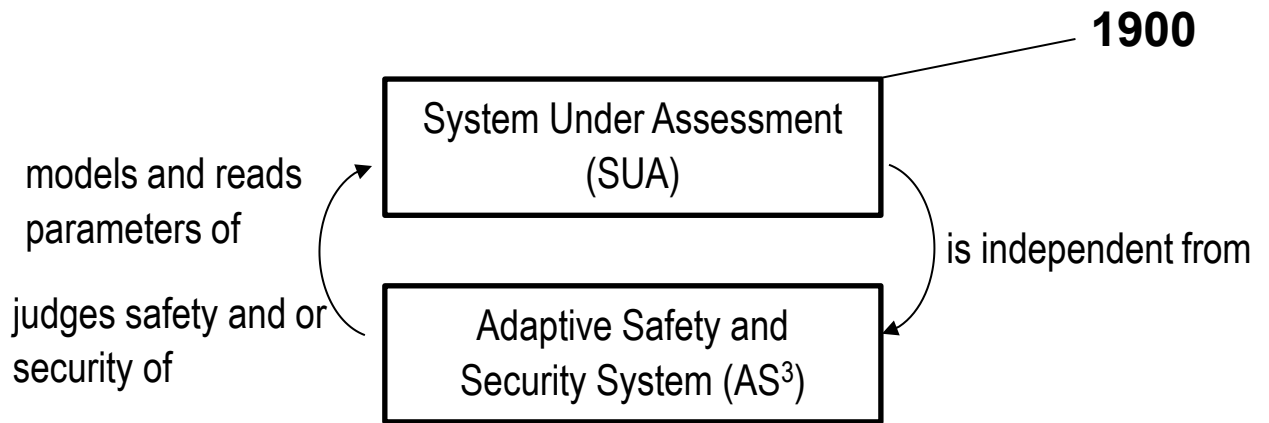


Fig. 1

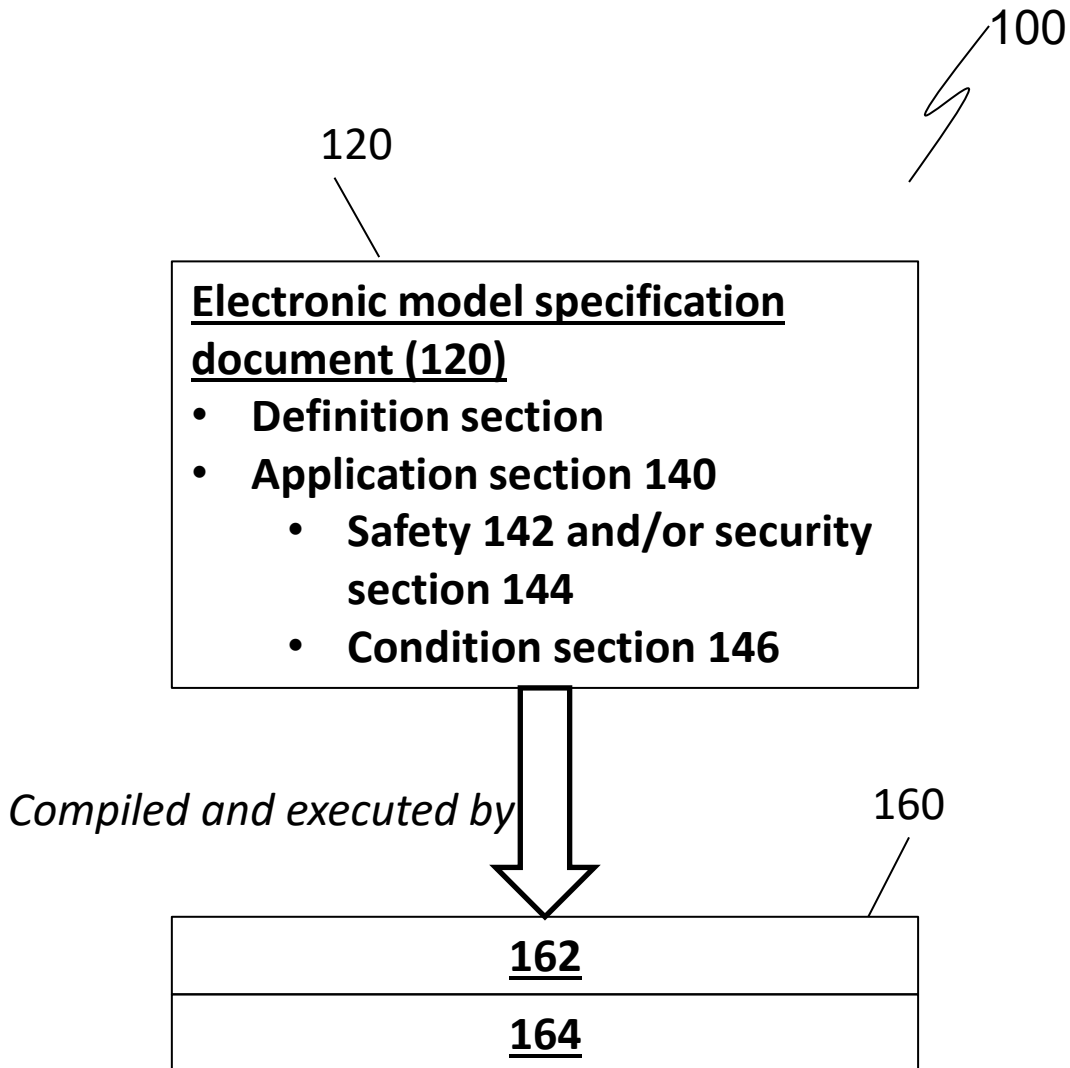


Fig. 2

3/14

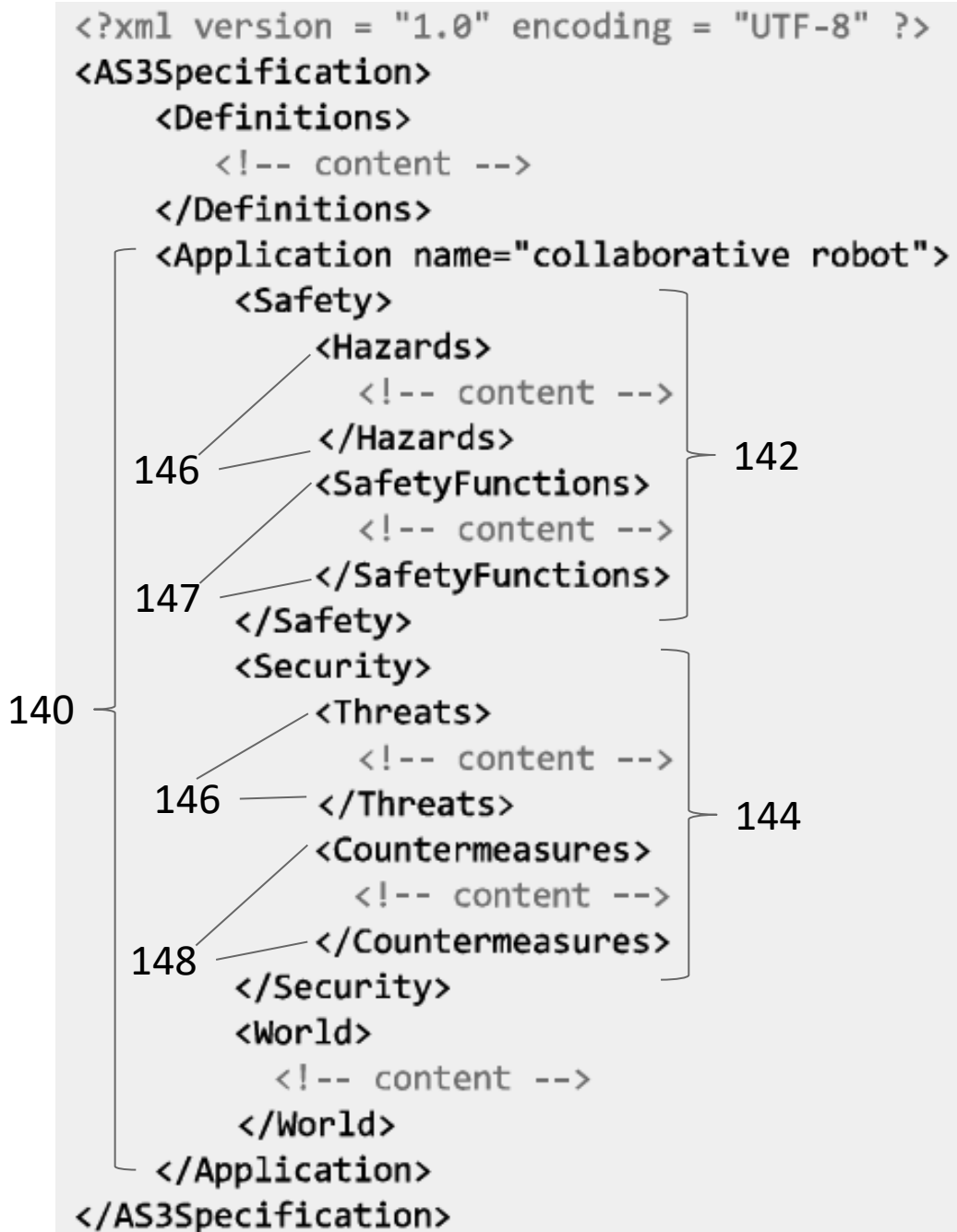


Fig. 3a

AS ³ Language Element	Description
Specification	Root element marking AS ³ content. AS ³ specifications may be embedded in other documents.
Definitions	Section of an AS ³ specification document that introduces default and user defined constructs. For example, AS ³ allows to define basic elements such as types for values and composite elements such as a new type of a robot.
Application	Section of an AS ³ specification document that instantiates elements relevant of the SUA. An application is a generalization of a SUA as it includes both the system components and the use case. For example, a SUA might include three robots of a given type and a manufacturing task. Each application also includes elements relevant to the safety and security assessment that might not belong to any piece of equipment. It would be valid to define more than one application.
Safety	Section of an AS ³ specification document that lists safety relevant properties, including hazards and protective measures. (see AS ³ safety profile in section 3.4.5)
Security	Section of an AS ³ specification document that lists security relevant properties, including threads and countermeasures (see AS ³ security profile in section 3.4.5)
World	Section of an AS ³ specification document that lists any other relevant properties of an SUA for the assessment that are neither uniquely risk nor mitigation.
Condition	Section of an AS ³ specification document that specifies a condition, for example the conditions that need to come true for a hazard to materialize. Furthermore, it forms the base element to model and import conditions of acceptability from standards and regulations.

Fig. 3b

5/14

Language Element	Description
Import	A statement of the AS ³ language to reference external fragments. These could be stored alongside the containing document or accessible over network infrastructure. Each language realization defines its own syntax and semantic of resource identification (e.g. using URI and hyperlinks). It is important to note that it is valid to reference individual fragments of an external AS ³ Specification Document.
Atom	Instantiates a value of a given type. This could be as simple as base types listed in Table 3 (e.g. approximation of PI as 3.14159265359), types with a physical dimension (e.g. a velocity of 2 m/s) or any other composite item (e.g. an instance of a machine type).
Script Variable	A variable that is resolved to a value during the “script” evaluation phase (see section 3.4.2)
Solver variable	Solver variables do not receive a value, but they are constituents of conditions. Hence, they are search parameters for judging the acceptability of an SUA.
Function	Functions are first class objects in AS ³ and even allow polymorphism. Operators such as “equal” are nothing more than a function that takes to parameters and resolves to a value of type Boolean. It is important to note that AS ³ optimizes evaluation of functions and allows to execute them either at “script” or at “solver” stage. For this they need to be treated as parameterizable objects.
Bind	The AS ³ bind construct can link variables to single values and complex expressions outside the scope of the current AS ³ fragment. This allows cross-referencing other parts of the document and helps to keep an AS ³ model in sync as it avoids repeating of fragments.

Fig. 3c

6/14

AS ³ Type	Description
AS3Type	The base type from which all other types are derived. Each value is at least of type “AS3Type”.
Undefined	A type with the single value “undefined” to mark the absence of a valid value.
Enumeration	An enumeration allows the definition of new value tokens (e.g. to define a type “Performance Level” as the set of the tokens “A”, “B”, “C”, “D”)
Boolean	Implements the semantics of a classical Boolean type in AS ³ .
Integer	Implements the semantics of a natural numbers type in AS ³ . The value range can be configured with an AS ³ option.
Real	Implements the semantics of floating-point type in AS ³ . The value range can be configured with an AS ³ option.
String	Implements the semantics of a string type. Of note is the mapping into the domain of constraint solvers as enumeration. Each instantiation of a string value will automatically be added to the value space of an internally managed enumeration.
Array	A type to represent arrays of elements of a given other type.
Bitvector	Implements an optimization specialization of an array of Booleans.
Composite	Base type to define any other composite types (e.g. a complex number type with a real and imaginary part).
Asset	Base type definition of a physical assets, which is derived from composite type. Users may use this base class to define their own asset types. A specification may instantiate as many objects of a type as required.
User	A specialization type derived from composite to implement a user type according to standard requirements (e.g. having a head, arms, torso, etc.) The user type is instrumental to define various types of hazards.

Fig. 3d

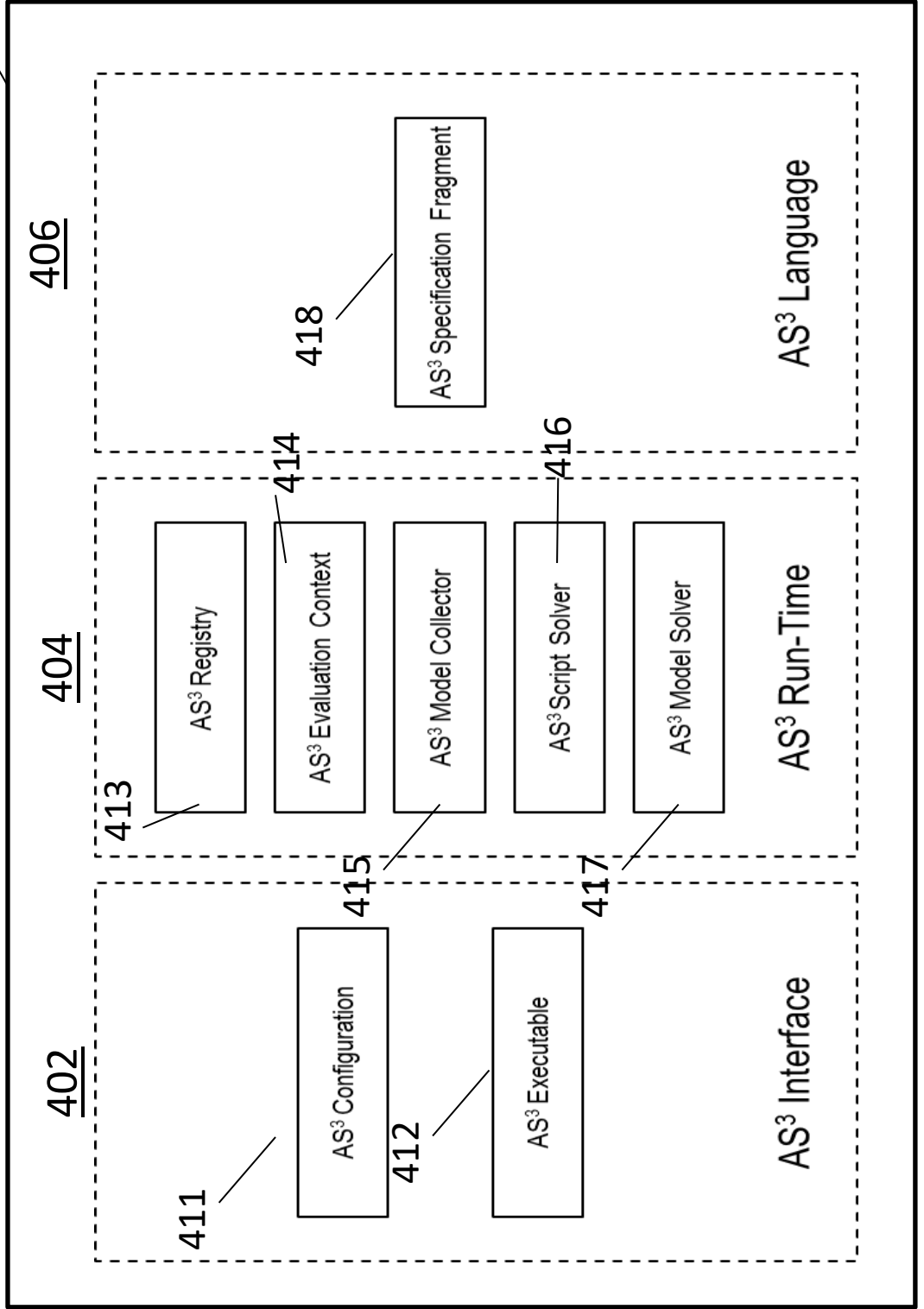


Fig. 4

8/14

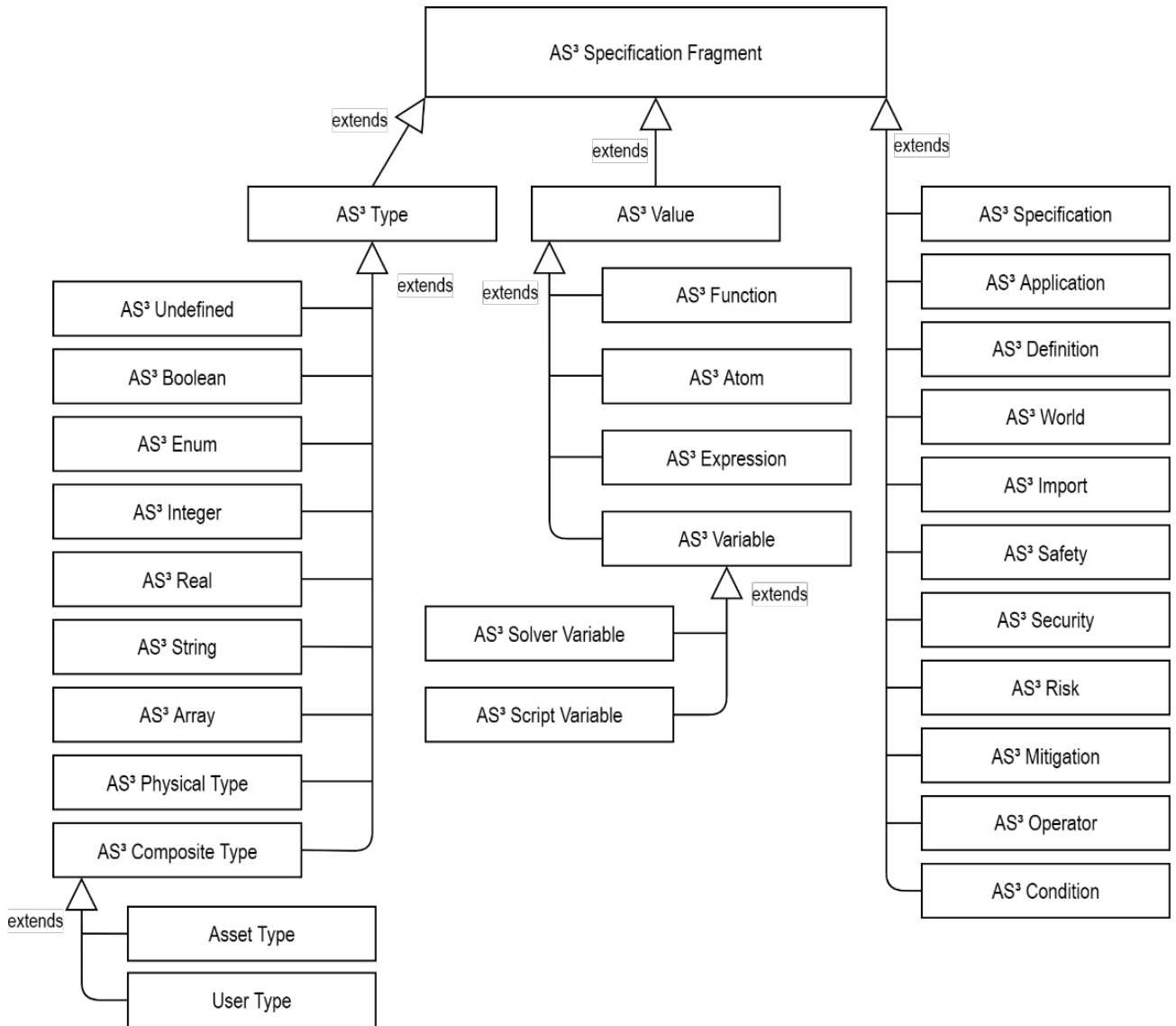


Fig. 5

9/14

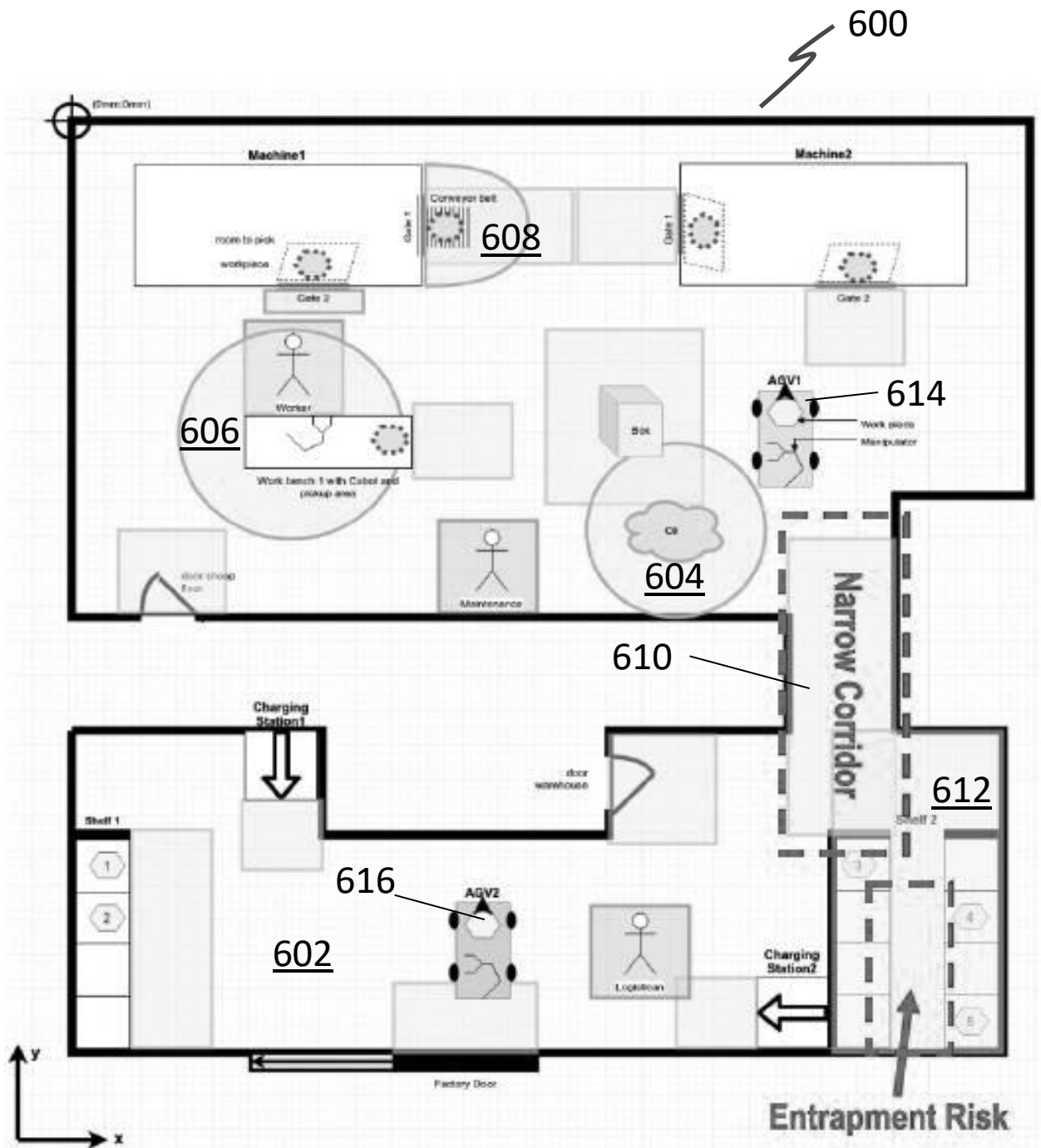


Fig. 6

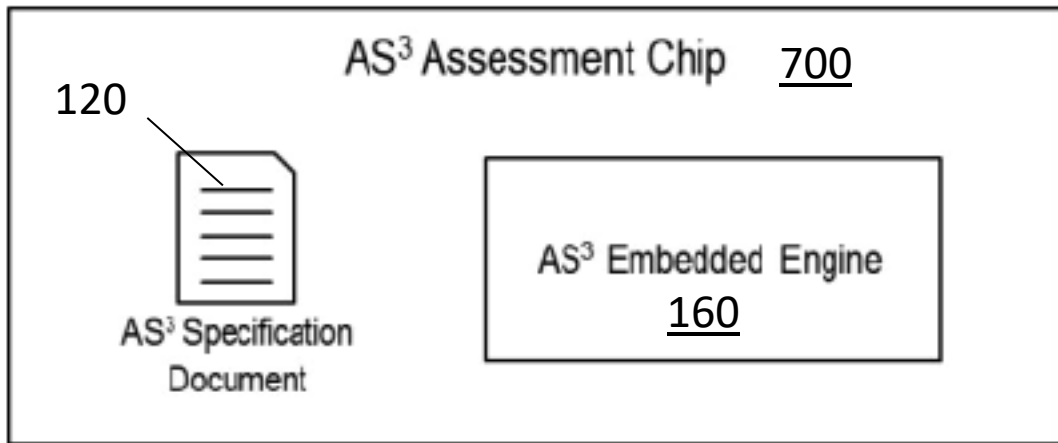


Fig. 7

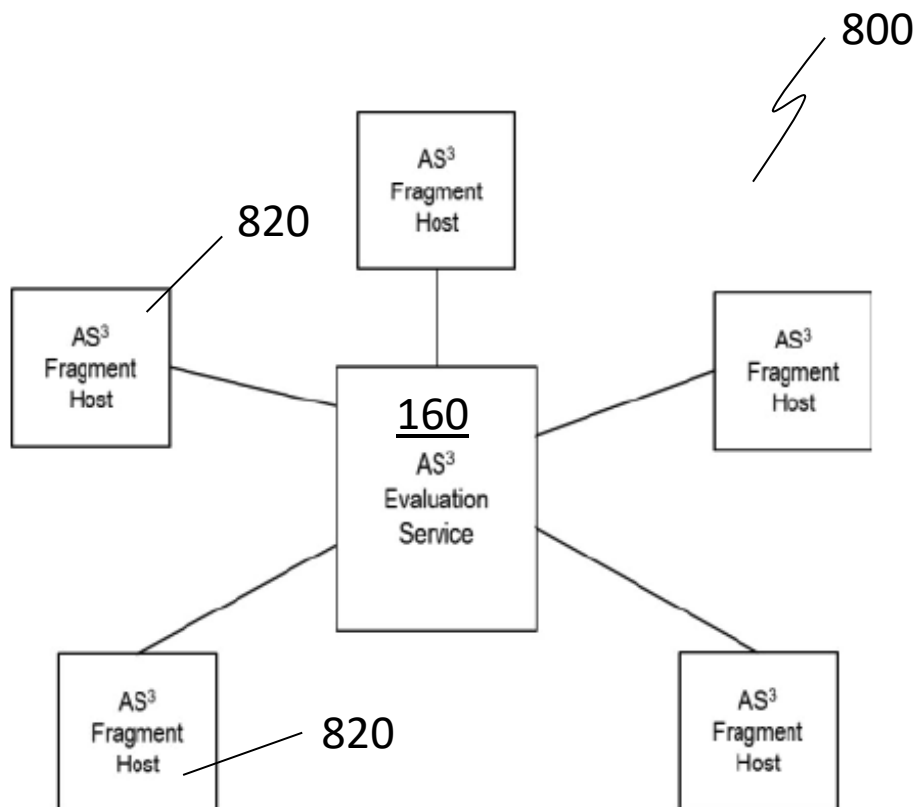


Fig. 8

12/14

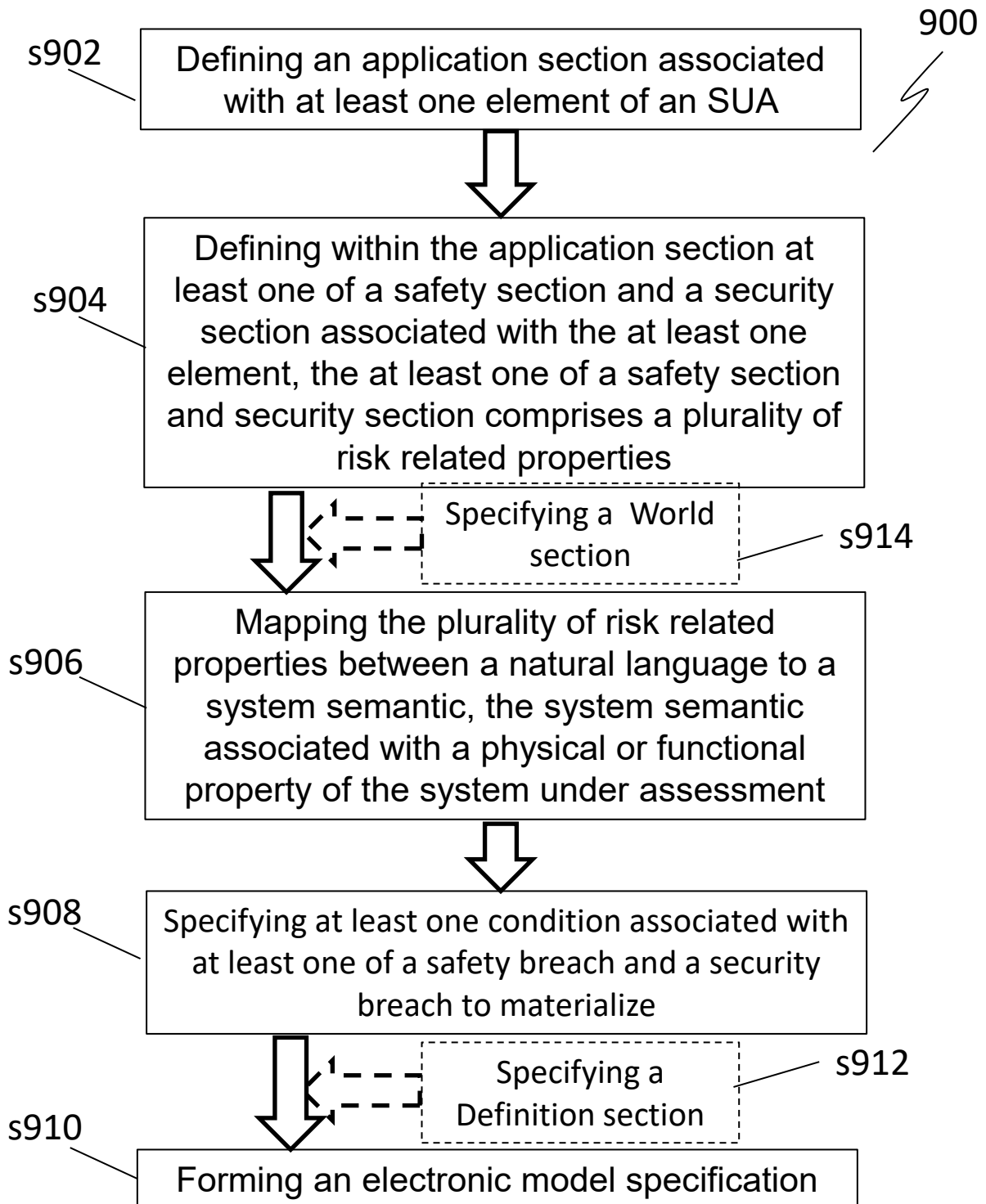


Fig. 9

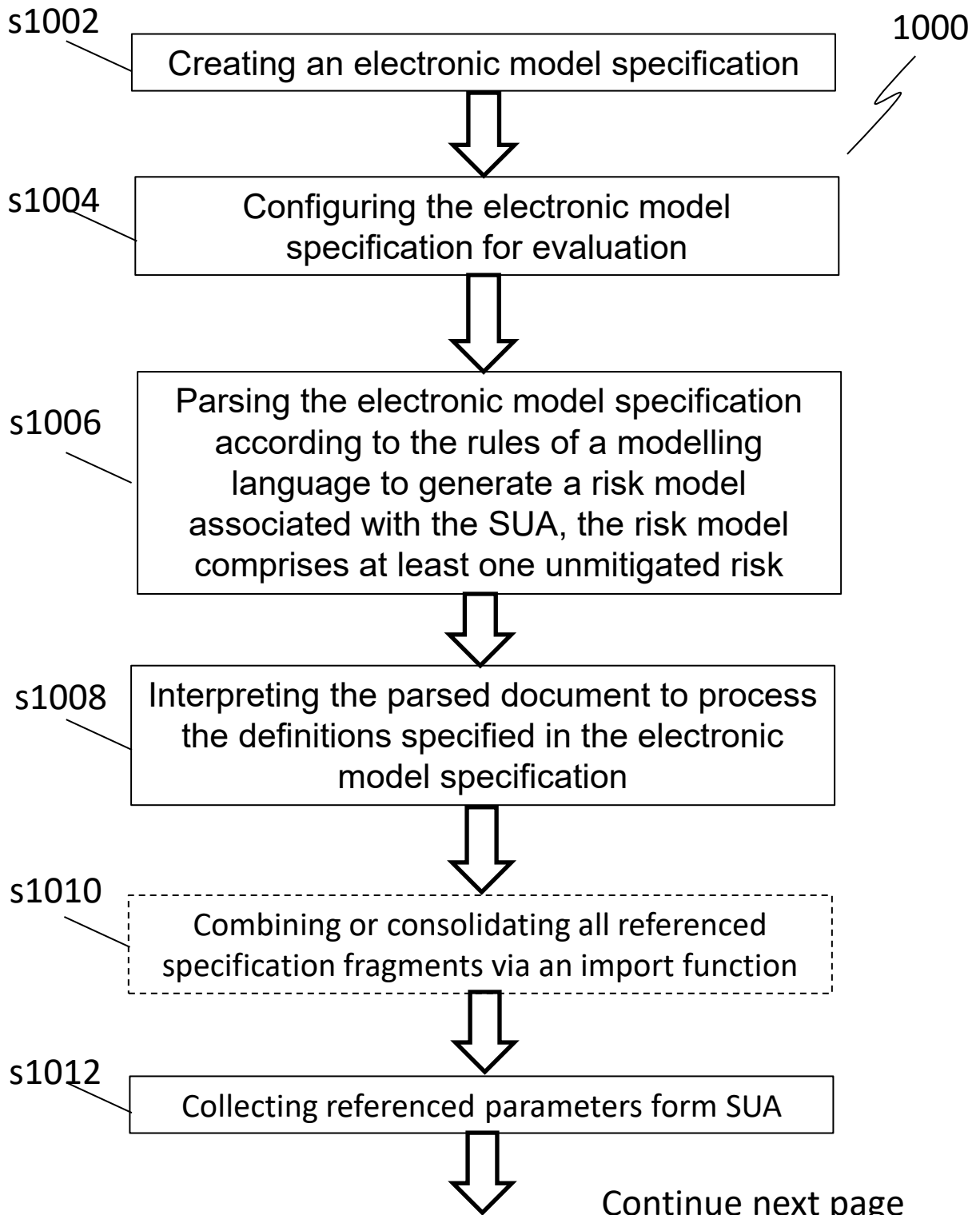


Fig. 10

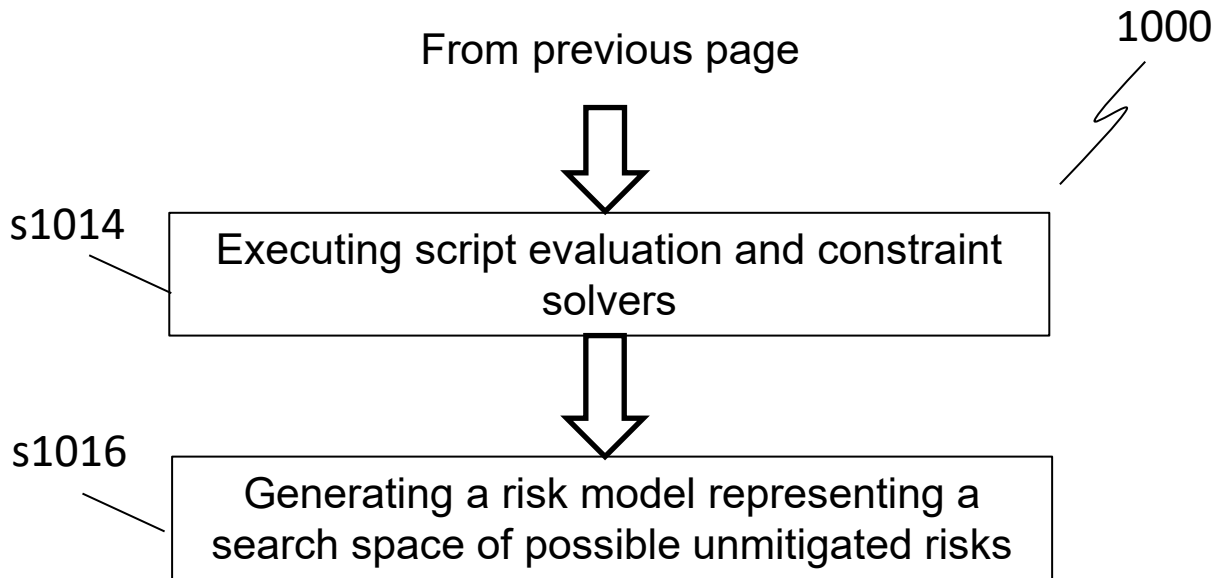


Fig. 10