OntoGSN Design Document

Author: Tomas Bueno Momčilović (fortiss)

Contributors: Ingmar Kessler (fortiss), Dian Balta (fortiss), Barbara Gallina (Mälardalen University)

Created: 04-12-2024

Last updated: 08-05-2025

Version: 1.0.2

Contents

Reviewers: Read Here

Thank you for agreeing to review this ontology!

Acknowledgements: Many thanks to Yannick Landeck and Radouane Bouchekir for reviewing this work. Thanks to Damir Safin for tool suggestions. Thanks to Will Franks and Adelard (NCC Group) for providing the ASCE academic license.

About

Purpose of the document: This document records all the design decisions made in the process of creating the ontology.

Goal of the ontology: I want to enable users to create, maintain, align, evaluate, explain and display assurance cases using the advantages of the semantic technology stack. While users can already create assurance cases with custom tools and store them in other formats, using an ontology has the following benefits:

- storing and querying graph data in a structure made for that purpose;
- representing the domain or world in human-readable *visual-izable* format;
- integrating references to data, documents or code easily and in the same store;
- automating rules and verification of quality with logic-based reasoners;
- providing the basis for more advanced methods and extensions (e.g., GraphRAG);
- making use of a vibrant community and (mostly) free and open-source tools.

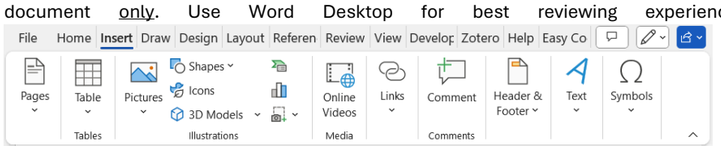
Methodology: Every element of the ontology is sourced directly from the Goal Structuring Notation Community Standard v3. There were two main activities: creating the taxonomy of classes and properties (i.e., the TBox); and creating the roles/rules governing the properties between individuals of classes (i.e., the RBox and the rules)[1]. Regarding the taxonomy, each sentence of the standard is parsed with the goal of translating the concepts and their relations into semantic triples (i.e., subject-predicate-object statements). Regarding the rules, the sentences which place conditions or restrictions on the identified elements of triples are translated into logical statements. Given that things can be represented in multiple ways, this process involves a good degree of interpretation.

Technical implementation: The ontology was created using *Stanford Protégé*[2] (v5.6.3) ontology editor, and following the *Web Ontology Language (OWL 2)*[3] standard. Existing objects and properties are imported from the following foundational ontologies: *Resource Description Framework Schema* (RDFS)[4], *XML Schema Definition Language* (XSD)[5], *Dublin-Core* (DC)[6], *Schema.org*[7], and *Simple Knowledge Organization System* (SKOS)[8]. Reasoning is based on the *Semantic Web Rule Language* (SWRL)[9] rules and OWL axioms, which can be executed with supported rule engines (e.g., Pellet[10] or Drools[11]). Rules defined as *SPARQL Protocol and RDF Query Language* (SPARQL)[12] queries, or constraints under *Shapes Constraints Language* (SHACL)[13], are currently not active, but may be supported in a future version.

How to review

General instructions

- These files represent a *work-in-progress* artefact that is prepared for a submission to a conference. Please do not share the contents of this repository. Once work is complete, everything will be open-sourced.
- Please provide any comments and questions using *"Insert > Comment"* in this Word document only. Use Word Desktop for best reviewing experience.



- Relevant columns for review are: *Item in GSN Community Standard*, *Simplified Item in Ontology*, and *Reason(s) for in-/exclusion*. Other columns can be safely ignored.
- All reviewers will be mentioned in the acknowledgements of the published document. If your name is not visible above the comment, please add it at the end of the comment text. If you wish to remain anonymous, please let me know in a direct message.
- Please do not modify any of the .owl ontology files directly in sync with the Sharepoint. Feel free to use your favorite editors or tools, but please do so with local copies.

Ontological POV

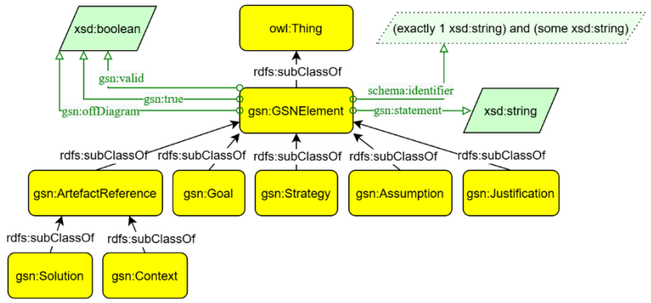
- For best experience, view gsn.owl in Protégé. To view the ontology in code/text editor, it is best to save it as a .ttl file beforehand.
- Each GSN class or property should have a "core or extension" annotation corresponding to its source section. In special cases, sources of particular assertions about a class or property (e.g., restrictions) should also be indicated. If that is not the case, please let me know.
- To view the rules in Protégé, install the ROWLTab[14] Plugin (v2.1.3). To test the execution of rules, install the Pellet Reasoner Plug-In (v2.2.0) to run the Pellet reasoner, or the ROWLTab Plugin (v2.1.3) to run the Drools reasoner.
- This ontology has already been evaluated using the Ontology Pitfall Scanner (see results here).

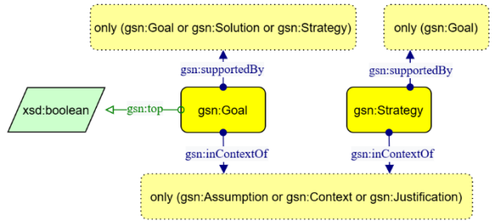
Assurance case POV

- If you have comments regarding something in the GSN standard (e.g., a rule missing in the ontology), please provide a page number so that I can find it more easily.
- Part 1 is normative, while Part 2 is non-normative. However, where sensible, suggestions from Part 2 should have been implemented in the ontology.
- If you disagree with my interpretation of the standard, please do not hesitate to leave a comment. Even choices that I made for feasibility reasons (i.e., due to the limitations of the editor or the underlying ontology frameworks) should be scrutinized. Alternative interpretations are welcome.

[1] The assertion box (ABox) is to be created by the user; currently, only example individuals are provided in a separate file for tutorial purposes.
[2] https://protege.stanford.edu/
[3] https://www.w3.org/TR/owl2-overview
[4] https://www.w3.org/TR/rdf-schema/
[5] https://www.w3.org/TR/xmlschema11-1/
[6] http://purl.org/dc/elements/1.1/
[7] https://www.schema.org
[8] https://www.w3.org/2004/02/skos/
[9] https://www.w3.org/submissions/SWRL/
[10] https://www.w3.org/2001/sw/wiki/Pellet
[11] https://www.drools.org/
[12] https://www.w3.org/TR/sparql11-query/
[13] https://www.w3.org/TR/shacl/
[14] In v2.1.2 and below, it is known as SWRLTab.

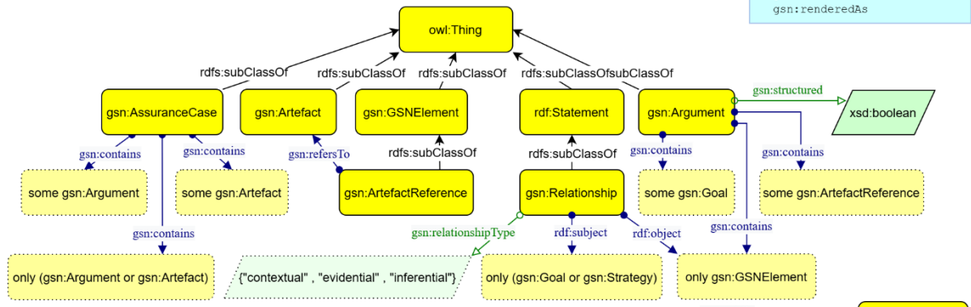
Diagram

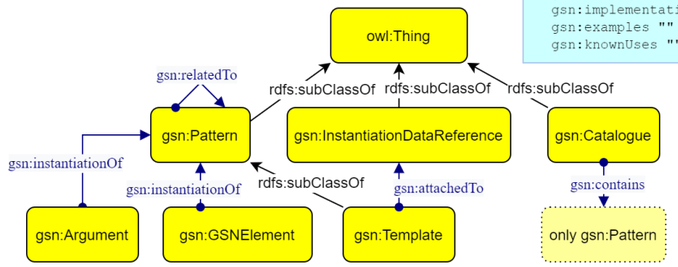




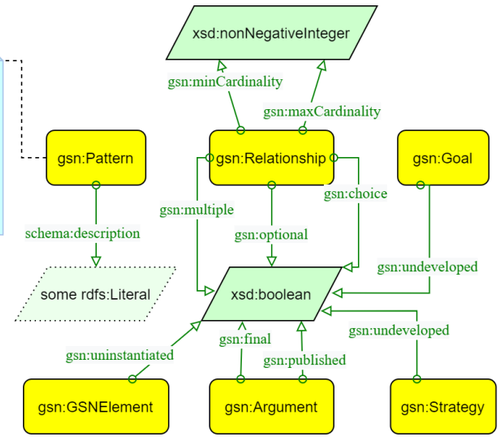

```
Custom Annotations:
  gsn:coreOrExtension ""
  gsn:renderedAs
```

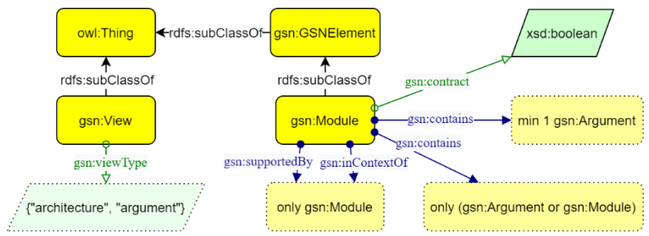
```
Imported Annotations:
  gsn:coreOrExtension ""
  dc:created ""
  dc:creator ""
  dc:identifier
  dc:modified
  dc:publisher
  dc:source
  dc:title
  dcterms:abstract
  dcterms:created
  swrla:isRuleEnabled
  rdfs:label
  schema:citation
  schema:description
  schema:disclaimer
  schema:license
  schema:url
  schema:version
  skos:altLabel
  skos:definition
  skos:note
```

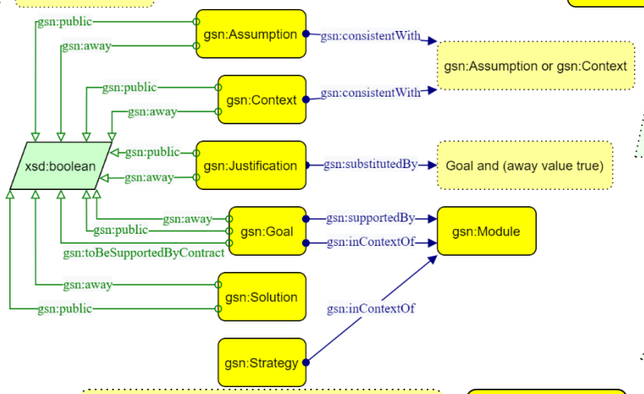


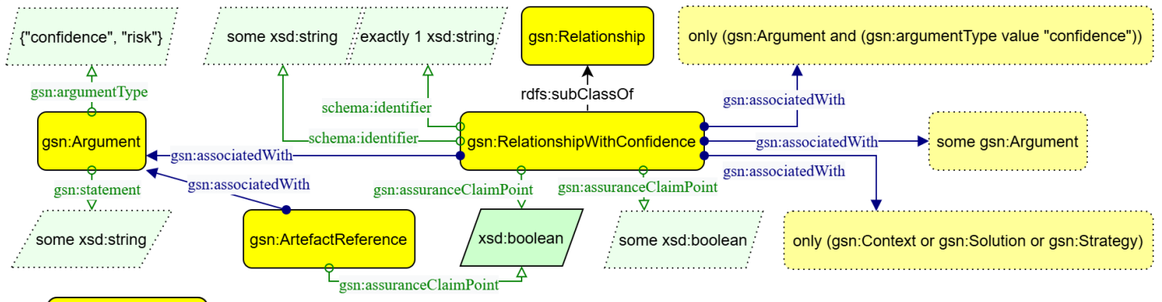


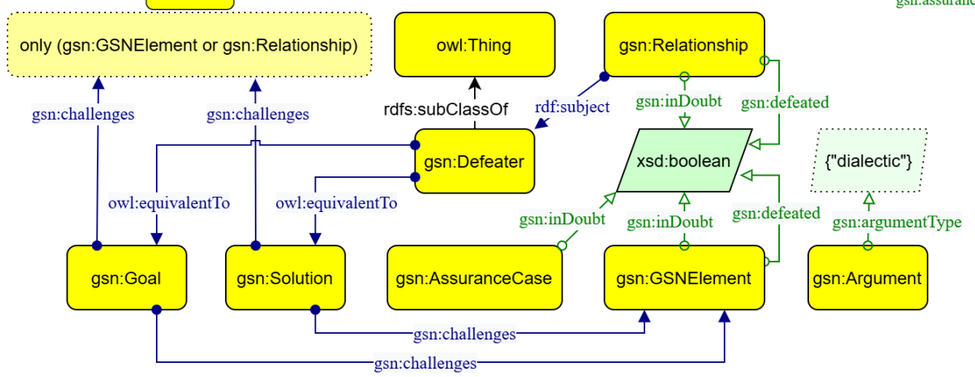

```
Annotation:
  gsn:intent ""
  gsn:motivation ""
  gsn:structure ""
  gsn:participants ""
  gsn:collaboration ""
  gsn:applicability ""
  gsn:consequences ""
  gsn:implementation ""
  gsn:examples ""
  gsn:knownUses ""
```











Part 0 etc.

Ontology-Specific Statements

id	Item in GSN Community Standard	Page(s)	Item in GSN Ontology File(s)	Simplified Item in Ontology	Reason(s) for in-/exclusion
I.1	-	-	xmlns="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#"	Base prefix “http://...gsn”	Protege’s automatic declaration of default prefixes for the GSN ontology namespace. The URL is tentative.
I.2	-	-	xml:base="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn"		
I.3	-	-	xmlns:gsn="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#"	Prefix gsn “http://...gsn#”	
I.4	-	-	xmlns:dc="http://purl.org/dc/elements/1.1/"	Prefix dc “http://...1.1/”	Protege’s automatic declaration of prefixes of helper (foundational) ontologies. The use of HTTP instead of HTTPS is due to Protégé.
I.5	-	-	xmlns:owl="http://www.w3.org/2002/07/owl#"	Prefix owl “http://...owl#”	
I.6	-	-	xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"	Prefix rdf “http://...ns#”	
I.7	-	-	xmlns:xml="http://www.w3.org/XML/1998/namespace"	Prefix xml “http://...namespace#”	
I.8	-	-	xmlns:xsd="http://www.w3.org/2001/XMLSchema#"	Prefix xsd “http://...XMLSchema#”	
I.9	-	-	xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"	Prefix rdfs “http://...schema#”	
I.10	-	-	xmlns:skos="http://www.w3.org/2004/02/skos/core#"	Prefix skos “http://...core#”	
I.11	-	-	xmlns:schema="http://schema.org/"	Prefix schema “http://...org/”	
I.12	-	-	xmlns:vann="http://purl.org/vocab/vann/"	Prefix vann “...vann/”	
I.13	-	-	xmlns:swrla="http://swrl.stanford.edu/ontologies/3.3/swrla.owl#"	Prefix swrla “...swrla.owl#“	
I.14	-	-	xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"	Prefix swrlb “...swrlb#”	
I.15	-	-	xmlns:swrl="http://www.w3.org/2003/11/swrl#"	Prefix swrl “...swrl#”	
I.15b	-	-	xmlns:swrlx=“http://swrl.stanford.edu/ontologies/built-ins/3.3/swrlx.owl#”	Prefix swrl “...swrlx#”	
I.16	-	-	xmlns:terms="http://purl.org/dc/terms/"	Prefix terms “terms/”	
I.17	-	-	<owl:AnnotationProperty rdf:about="http://purl.org/dc/elements/1.1/created"/>	created a AnnotationProperty	Imported reusable/helper annotation and datatype properties from foundational ontologies: Schema, SKOS and FOAF.
I.18	-	-	<owl:AnnotationProperty rdf:about="http://purl.org/dc/elements/1.1/creator"/>	creator a AnnotationProperty	
I.19	-	-	<owl:AnnotationProperty rdf:about="http://purl.org/dc/elements/1.1/identifier"/>	dc:identifier a AnnotationProperty	
I.20	-	-	<owl:AnnotationProperty rdf:about="http://purl.org/dc/elements/1.1/modified"/>	modified a AnnotationProperty	
I.21	-	-	<owl:AnnotationProperty rdf:about="http://purl.org/dc/elements/1.1/publisher"/>	publisher a AnnotationProperty	
I.22	-	-	<owl:AnnotationProperty rdf:about="http://purl.org/dc/elements/1.1/source"/>	source a AnnotationProperty	
I.23	-	-	<owl:AnnotationProperty rdf:about="http://schema.org/disclaimer"/>	disclaimer a AnnotationProperty	
I.24	-	-	<owl:AnnotationProperty rdf:about="http://schema.org/license"/>	license a AnnotationProperty	
I.25	-	-	<owl:AnnotationProperty rdf:about="http://schema.org/url"/>	url a AnnotationProperty	
I.26	-	-	<owl:AnnotationProperty rdf:about="http://schema.org/version"/>	version a AnnotationProperty	
I.27	-	-	<owl:AnnotationProperty rdf:about="http://www.w3.org/2004/02/skos/core#definition"/>	definition a AnnotationProperty	
I.28	-	-	<owl:AnnotationProperty rdf:about="http://www.w3.org/2004/02/skos/core#altLabel"/>	altLabel a AnnotationProperty	
I.29	-	-	<owl:DatatypeProperty rdf:about="http://schema.org/identifier"/>	schema:identifier a DatatypeProperty	
I.30	-	-	<owl:Class rdf:about="http://www.w3.org/1999/02/22-rdf-syntax-ns#Statement"/>	Statement a Class	Base class and object properties for the reification of triples, i.e. statements which allow attaching properties to triples. For example, for some triple ‘*A* **follows** *B*’, we can assert that: *A_follows_B* **is a** *Statement*; *A_follows_B* **has subject** *A*; *A_follows_B* **has predicate** *follows*; *A_follows_B* **has object** *B*; and *A_follows_B* **is** *valid*.
I.31	-	-	<owl:ObjectProperty rdf:about="http://www.w3.org/1999/02/22-rdf-syntax-ns#subject"/>	subject a ObjectProperty	
I.32	-	-	<owl:ObjectProperty rdf:about="http://www.w3.org/1999/02/22-rdf-syntax-ns#predicate"/>	predicate a ObjectProperty	
I.33	-	-	<owl:ObjectProperty rdf:about="http://www.w3.org/1999/02/22-rdf-syntax-ns#object"/>	object a ObjectProperty	
I.34	-	-	<owl:Ontology rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn">	gsn a Ontology	Protege’s automatic declaration of the universal resource identifier (URI) and membership for the GSN ontology.
I.35	-	-	<terms:created xml:lang="en">20th February 2025</terms:created>	gsn created “20th February 2025”	Metadata about the ontology creation process.
I.36	-	-	<dc:creator xml:lang="en">Tomas Bueno Momčilović</dc:creator>	gsn creator “Ontology: ...”	
I.37	-	-	<dc:modified rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2024-12-04T00:00:00Z</dc:modified>	gsn modified “2024-12-04T00:00:00Z”	
I.38	-	-	<dc:title xml:lang="en">OntoGSN</dc:title>	gsn title “OntoGSN”	
I.39	-	-	<terms:abstract xml:lang="en">OntoGSN is an ontology for creating assurance cases in the Goal Structuring Notation (GSN). The goal of the ontology is to help users in linking the elements of their cases - claims and evidence - with the internationalized resource identifiers (IRIs) of represented concepts, events and data, and in evaluating the validity of their argument.</terms:abstract>	gsn abstract “OntoGSN ...”	
I.40	-	-	<vann:preferredNamespacePrefix>gsn</vann:preferredNamespacePrefix>	gsn preferredNamespacePrefix “gsn”	
I.41	-	-	<schema:citation>Bueno Momcilovic, T. (2025 February). OntoGSN: Ontology for Goal Structuring Notation.</schema:citation>	gsn citation “Bueno ...”	
I.42	-	-			
I.43	-	-	<owl:versionInfo rdf:datatype="http://www.w3.org/2001/XMLSchema#decimal">1.0</owl:versionInfo>	gsn versionInfo “1.0”	

Preamble

id	Item in GSN Community Standard	Page(s)	Item in GSN Ontology	Simplified Item in Ontology	Reason(s) for in-/exclusion
II.1	Footer	2	<dc:creator xml:lang="en">Standard: The Assurance Case Working Group (ACWG)</dc:creator>	gsn creator "Standard: The ..."	Important metadata regarding the GSN metamodel.
II.2	SCSC-141C	1	<dc:identifier>SCSC-141C</dc:identifier>	gsn identifier "SCSC-141C"	
II.3	(...) GSN is a graphical argument notation which can be used to document explicitly the elements and structure of an argument and the argument’s relationship to evidence. (...)	9	<skos:definition xml:lang="en">GSN is a graphical argument notation which can be used to document explicitly the elements and structure of an argument and the argument’s relationship to evidence.</skos:definition>	gsn definition "GSN is ..."	
II.4	Footnote 1: SCSC : Safety-Critical Systems Club C.I.C. A Community Interest Company registered in England (Company number 13084663)	2	<dc:publisher xml:lang="en">https://www.fortiss.org/</dc:publisher>	gsn publisher "https://..."	
II.5	Goal Structuring Notation Community Standard Version 3. The Assurance Case Working Group (ACWG)	1	<dc:source xml:lang="en">The Assurance Case Working Group (ACWG). (2021 May). Goal Structuring Notation Community Standard Version 3. URL: http://scsc.uk/SCSC-141C</dc:source>	gsn source "The Assurance ..."	

II.6	Disclaimer		<schema:disclaimer xml:lang="en">[Disclaimer from the GSN Community Standard v3.0]: While (...) </schema:disclaimer>	gsn disclaimer "[Disclaimer from ..."	
II.7	License: This work is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/ or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.	3	<schema:license rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">https://creativecommons.org/licenses/by/4.0/deed.en</schema:license>	gsn license "https://..."	
II.8	A meta-model of GSN, showing the relationship to SACM, can be found at scsc.uk/gsn	3	<schema:url rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">http://scsc.uk/gsn</schema:url>	gsn url "http://scsc.uk/gsn"	
II.9	Document History	4	<schema:version rdf:datatype="http://www.w3.org/2001/XMLSchema#decimal">3.0</schema:version>	gsn version "3.0"	
II.10	Declaration	2	-	-	These metadata are relevant only for the document (provenance, etc.), and not the current model itself.
II.11	Foreword	3	-	-	
II.12	Change History	4	-	-	
II.13	Future Development	4	-	-	
II.14	Contributors	5	-	-	

Part 0, Glossary & Annex

id	Item in GSN Community Standard	Page(s)	Item in GSN Ontology	Simplified Item in Ontology	Reason(s) for in-/exclusion
III.1	1:2.1.4 Table 1:2-1 provides the definition and rendering of these elements.	17	<owl:AnnotationProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#renderedAs"> <rdfs:label xml:lang="en">rendered as</rdfs:label> </owl:AnnotationProperty>	renderedAs a AnnotationProperty	Because GSN is a visual language, “renderedAs” provides the geometric description of each element.
III.2				renderedAs label "rendered as"	
III.3	1.2 Core GSN 1.3 Argument Pattern Extension 1.4 Modular Extension 1.5 Confidence Argument Extension 1.6 Dialectic Extension	6	<owl:AnnotationProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#coreOrExtension"> <rdfs:label xml:lang="en">core or extension</rdfs:label> </owl:AnnotationProperty>	coreOrExtension a AnnotationProperty	Allows each element to be associated with the relevant subset of the normative standard, so that the user can filter out triples of unused extension.
III.4				coreOrExtension label "core or extension"	
III.5	0:2.2 An assurance case can be defined as: A reasoned and compelling argument, supported by a body of evidence, that a system, service or organisation will operate as intended for a defined application in a defined environment.	10	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#AssuranceCase"> <coreOrExtension>Core GSN</coreOrExtension> <rdfs:label xml:lang="en">Assurance Case</rdfs:label> <skos:definition xml:lang="en">A reasoned and compelling argument, supported by a body of evidence, that a system, service or organisation will operate as intended for a defined application in a defined environment.</skos:definition> <skos:definition xml:lang="en">Arguments and evidence intended to demonstrate that a system meets its assurance requirements.</skos:definition> </owl:Class>	AssuranceCase a Class	
III.6				AssuranceCase coreOrExtension "Core GSN"	
III.7				AssuranceCase label "Assurance Case"	
III.8				AssuranceCase definition "A reasoned ...”	The definitions are duplicates, but both are preserved since they are in the standard.
III.9	**Glossary: Assurance Case** Arguments and evidence intended to demonstrate that a system meets its assurance requirements.	128		AssuranceCase definition "Arguments ...”	
III.10	0:3.1 In the sense used in assurance cases, an argument is defined as a connected series of claims intended to establish an overall claim.	10	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"> <coreOrExtension xml:lang="en">Core GSN</coreOrExtension> <rdfs:label xml:lang="en">Argument</rdfs:label> <skos:definition xml:lang="en">A body of information presented with the intention to establish one or more claims through the presentation of related supporting claims, evidence and contextual information.</skos:definition> <skos:definition xml:lang="en">A connected series of claims intended to establish an overall claim.</skos:definition> </owl:Class>	Argument a Class	
III.11				Argument coreOrExtension "Core GSN"	
III.12				Argument label "Argument"	
III.13				Argument definition "A connected ... “	
III.14	**Glossary: Argument** A body of information presented with the intention to establish one or more claims through the presentation of related supporting claims, evidence and contextual information.	128		Argument definition "A body ...”	
III.15	**Glossary: Claim** A proposition being asserted by the author that is a true or false statement.	128	~~<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"> <coreOrExtension xml:lang="en">Core GSN</coreOrExtension> <rdfs:label xml:lang="en">Claim</rdfs:label> <skos:definition xml:lang="en">A proposition being asserted by the author that is a true or false statement.</skos:definition> </owl:Class>~~	~~Claim a Class~~	The meaning of claim in GSN was initially mixed up with claim in the language. After reviewing the standard text, it is clearer that claims only refer to goals.
III.16				~~Claim coreOrExtension "Core GSN"~~	
III.17				~~Claim label "Claim"~~	
III.18				~~Claim definition "A proposition ...”~~	
III.15b			[INSERT OWL STATEMENTS HERE]	Goal altLabel “Claim”	
III.19	**Glossary: Structured argument** A particular kind of argument where the relationships between the asserted claims, and from the evidence to the claims, are explicitly represented.	128	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#structured"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension>Core GSN</coreOrExtension> <rdfs:label xml:lang="en">structured</rdfs:label> <skos:definition xml:lang="en">A particular kind of argument where the relationships between the asserted claims, and from the evidence to the claims, are explicitly represented.</skos:definition> </owl:DatatypeProperty>	structured a DatatypeProperty	It is unclear whether “structured” is a property relevant for evaluation. It is added as a datatype property just in case.
III.20				structured domain Argument	
III.21				structured range boolean	
III.22				structured label "structured"	
III.23				structured definition "A particular ...”	
III.24	Footnote 3: ‘dialectic’ is defined by the oxford English dictionary as “Logic, reasoning; critical investigation of truth through reasoned argument, often spec. by means of dialogue or discussion.”	11	-	-	“Dialectic” is implicitly represented in the “Defeater” concept. For “dialectic argument”, see Part 2.
III.25	**Glossary: Dialectic** The process of investigating truth. This can occur in a minimal form by simply challenging statements made in an assurance case, but can also take a graphical form within a GSN argument	128			
III.26	0.4.1 (...) The relationships represented in GSN are:	11	-	-	

	• The premise-conclusion relationship between supporting goals and their parent goal; • The support that solutions provide for goals;				Because their role in GSN is unclear, and there is an equivalent data property for defining a top goal, “Premise” and “Conclusion” are not defined as subclasses or types of goals in this version.
III.27	0:4.2 The purpose of GSN is to document how claims (conclusions, represented in GSN as goals) are said to be supported by sub-claims (premises, also represented in GSN as goals).	11			
III.28	0.4.1 (…) • The relationship between the argument and the context in which it is stated.	11	-	-	Argument-Context relationship is undefined in the standard (as opposed to, e.g., Goal-Context), so this part is ignored.
III.29	0.4.11 (…) The goal structures also clearly document the context in which the claims of the argument are being put forward.	15			
III.30	0:4.3 Where evidence is asserted to support the truth of the claim, this can be documented by providing a solution in GSN.	12	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"> <skos:altLabel xml:lang="en">Evidence</skos:altLabel> </owl:Class>	ArtefactReference altLabel "Evidence"	“Evidence” is represented as an alternative label and not explicitly, due to competing “Artefact Reference” and “Solution” concepts. Unclear how these three concepts interface.
III.31	**Glossary: Evidence** Information or objective artefacts being offered in support of one or more claims.	128			
III.32	0:4.6 Some claims and argument strategies are expressed in the context of assumptions. These assumptions must be valid for the claim or the strategy to be valid.	12	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#valid">	valid a DatatypeProperty	
III.33			<rdfs:domain>	valid domain GSN Element	
III.34			<owl:Class>	valid range boolean	
III.35			<owl:unionOf rdf:parseType="Collection">	valid coreOrExtension “Core GSN”	
III.36			<rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> </owl:unionOf> </owl:Class> </rdfs:domain> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension>Core GSN</coreOrExtension> <rdfs:label xml:lang="en">valid</rdfs:label> </owl:DatatypeProperty>	valid label “valid”	
III.37			gsn:Assumption(?A) ^ gsn:valid(?A, false) ^ gsn:inContextOf(?B, ?A) -> gsn:valid(?B, false)	**IF** ?A is an Assumption **AND** ?A is not valid **AND** ?B is in context of ?A **THEN** ?B is not valid	
III.38	0:4.9 GSN provides two types of linkage between elements: SupportedBy and InContextOf. SupportedBy relationships – represented by lines with solid arrowheads – indicate inferential or evidential relationships between elements. InContextOf relationships – represented as lines with hollow arrowheads – declare contextual relationships.	13	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#relationshipType"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> <rdfs:range> <rdfs:Datatype> <owl:oneOf> <rdf:Description> <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#List"/> <rdf:first>contextual</rdf:first> <rdf:rest> <rdf:Description> <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#List"/> <rdf:first>evidential</rdf:first> <rdf:rest> <rdf:Description> <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#List"/> <rdf:first>inferential</rdf:first> <rdf:rest rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#nil"/> </rdf:Description> </rdf:rest> </rdf:Description> </rdf:rest> </rdf:Description> </owl:oneOf> </rdfs:Datatype> </rdfs:range> <coreOrExtension>Core GSN</coreOrExtension> <rdfs:label xml:lang="en">relationship type</rdfs:label> <skos:definition xml:lang="en">SupportedBy relationships – represented by lines with solid arrowheads – indicate inferential or evidential relationships between elements. InContextOf relationships – represented as lines with hollow arrowheads – declare contextual relationships.</skos:definition> </owl:DatatypeProperty>	relationshipType a DatatypeProperty relationshipType domain Relationship relationshipType range one of (contextual, evidential, inferential) relationshipType coreOrExtension "Core GSN" relationshipType label "relationship type" relationshipType definition "SupportedBy …”	
III.39					
III.40					
III.41					
III.42					
III.43					
III.44			~~rdf:predicate(?R, ?O) ^ gsn:inContextOf(?O) -> gsn:relationshipType(?R, ”contextual”)~~	~~**IF** ?R has predicate ?O **AND** ?O is “in context of”~~ ~~**THEN** ?R has relationship type “contextual”~~	*EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rule formulated instead.*
III.45			rdf:subject(?R, ?A) ^ rdf:object(?R, ?B) ^ gsn:inContextOf(?A, ?B) -> gsn:relationshipType(?R, "contextual")	**IF** ?R has subject ?A **AND** ?R has object ?B **AND** ?A is in context of ?B **THEN** ?R has relationship type “contextual”	

III.46	0:4.10 When the elements of GSN are connected together, they are said to form a ‘goal structure’.	13	-	-	“Goal structure” is not represented explicitly, because: 1. The difference between goal structure, assurance case and argument is unclear; 2. “Goal structure” does not have any particular properties that cannot be represented in assurance cases and arguments.
III.47	0:4.11 Goal structures document the asserted chain of reasoning in the argument (through the visible decomposition of claimed goals and the description of argument strategies) and indicate how this argument is supported by evidence (through solutions).	15			
III.48	1:2.1.5 The core GSN elements defined here are intended to be combined to represent logical structures, known as ‘goal structures’.	18			
III.49	**Glossary: Evidential Relationship** A declared relationship between a claim and an evidence item by which the claim is substantiated.	128	~~rdf:predicate(?R, ?O) ^ gsn:supportedBy(?O) ^ rdf:object(?R, ?S) ^ gsn:Solution(?S) -> gsn:relationshipType(?R, ”evidential”)~~	~~**IF** ?R has predicate ?O **AND** ?O is “supported by” **AND** ?R has object ?S **AND** ?S is a Solution **THEN** ?R has relationship type “evidential”~~	*EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rule formulated instead.*
III.50			rdf:subject(?R, ?A) ^ rdf:object(?R, ?B) ^ gsn:supportedBy(?A, ?B) ^ gsn:Solution(?B) -> gsn:relationshipType(?R, "evidential")	**IF** ?R has subject ?A **AND** ?R has object ?B **AND** ?A is supported by ?B **AND** ?B is a Solution **THEN** ?R has relationship type “evidential”	
III.51			<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#relationshipType">	relationshipType definition “An evidential …”	Adding annotations to more advanced domain and range expressions leads to an error, so these statements are added as description.
III.52	**Glossary: Inferential Relationship** A declared inference between claims in the argument.	128	<skos:definition xml:lang="en">An evidential relationship is a declared relationship between a claim and an evidence item by which the claim is substantiated. An inferential relationship is a declared inference between claims in the argument. A contextual relationship draws attention to explanatory contextual information.</skos:definition> </owl:DatatypeProperty>		
III.53			~~rdf:predicate(?R, ?O) ^ gsn:supportedBy(?O) ^ rdf:object(?R, ?S) ^ gsn:Strategy(?S) -> gsn:relationshipType(?R, ”inferential”)~~	~~**IF** ?R has predicate “supported by” **AND** ?R has object ?S **AND** ?S is a Strategy **THEN** ?R has relationship type “inferential”~~	*EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rule formulated instead.*
III.54			~~rdf:predicate(?R, ?O) ^ gsn:supportedBy(?O) ^ rdf:object(?R, ?G) ^ gsn:Goal(?G) -> gsn:relationshipType(?R, ”inferential”)~~	~~**IF** ?R has predicate “supported by” **AND** ?R has object ?G **AND** ?G is a Goal **THEN** ?R has relationship type “inferential”~~	
III.55			rdf:subject(?R, ?A) ^ rdf:object(?R, ?B) ^ gsn:supportedBy(?A, ?B) ^ gsn:Strategy(?B) -> gsn:relationshipType(?R, "inferential")	**IF** ?R has subject ?A **AND** ?R has object ?B **AND** ?A is supported by ?B **AND** ?B is a Strategy **THEN** ?R has relationship type “inferential”	
III.56			rdf:subject(?R, ?A) ^ rdf:object(?R, ?B) ^ gsn:supportedBy(?A, ?B) ^ gsn:Goal(?B) -> gsn:relationshipType(?R, "inferential")	**IF** ?R has subject ?A **AND** ?R has object ?B **AND** ?A is supported by ?B **AND** ?B is a Goal **THEN** ?R has relationship type “inferential”	
III.57	0.3.2 At the heart of GSN is the explicit documentation of the hierarchy of claims and evidence. The top goal presents the overall claim asserted by the author and it is up to the reader to determine their belief that it is adequately supported.	11	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#top">	top a DatatypeProperty	
III.58			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/>	top domain Goal	
III.59			<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>	top range boolean	
III.60			<rdfs:label xml:lang="en">top (goal)</rdfs:label> </owl:DatatypeProperty>	top label "top (goal)"	
III.61			gsn:Goal(?A) ^ gsn:supportedBy(?A, ?B) ^ gsn:supportedBy(?X, ?A) -> gsn:top(?A, false)	**IF** ?A is a Goal **AND** ?A is supported by ?B **AND** ?X is supported by ?A **THEN** ?A is <u>not</u> a top goal	SWRL does not support negation or checking for blanks, and can only define class membership
III.62			~~INSERT { ?A :topGoal true . } WHERE { ?A a gsn:Goal . ?B :supportedBy ?A . FILTER NOT EXISTS { ?X :supportedBy ?A . } }~~	~~**IF** ?A is a Goal **AND** ?B is supported by ?A **AND** ?X is <u>not</u> supported by ?A **THEN** ?A is a top goal~~	SPARQL supports negation, but this option is left for future versions of the ontology, to reduce the number of dependencies.

III.63			<owl:Class> <owl:intersectionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <owl:Class> <owl:complementOf> <owl:Restriction> <owl:onProperty> <rdf:Description> <owl:inverseOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"/> </rdf:Description> </owl:onProperty> <owl:someValuesFrom rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> </owl:Restriction> </owl:complementOf> </owl:Class> </owl:intersectionOf> <owl:equivalentClass> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#top"/> <owl:hasValue rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean">true</owl:hasValue> </owl:Restriction> </owl:equivalentClass> </owl:Class> </rdf:RDF>	**IF** ?A is a Goal **AND** ?A is not inverse supported by some Goal **THEN** ?A is equivalent to a top goal	This is an OWL general class axiom, meaning that negation and inversion are supported, but also the inverse of the rule itself is enforced by a reasoner.
III.64				**IF** ?A is equivalent to a top goal **THEN** ?A must be a Goal AND ?A must not be inverse supported by some Goal	
III.65			<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#top"> <skos:definition xml:lang="en">The top goal presents the overall claim asserted by the author and it is up to the reader to determine their belief that it is adequately supported.</skos:definition> <skos:definition xml:lang="en">A GSN Goal that presents the pinnacle claim in an argument. It is ‘top’ in terms of the argument hierarchy, rather than necessarily its physical layout. There may be more than one top goal in a GSN structure.</skos:definition> </owl:DatatypeProperty>	top definition "The top …”	Both definitions are preserved because they introduce new information rather than duplicate it.
III.66	**Glossary: Top Goal** A GSN Goal that presents the pinnacle claim in an argument. It is ‘top’ in terms of the argument hierarchy, rather than necessarily its physical layout. There may be more than one top goal in a GSN structure.	128		top definition "A GSN …”	

Part 1

Core GSN

id	Item in GSN Community Standard	Page(s)	Item in GSN Ontology	Simplified Item in Ontology	Reason(s) for in-/exclusion
IV.1	1.1.1 (…) GSN defines elements, the allowable relationships between these elements and the acceptable language of the text within these elements.	16	-	-GSNElement a Class	In the previous version, “GSN Element” was not an explicit concept or alias, because: it was assumed thatGSN elements are essentially a union / superclass of Claims and Artifact References. However, 1.
IV.2	1:2.1.2 Each element contains an element identifier. The identifier shall identify the element uniquely within an argument module.	16	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://schema.org/identifier"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> </rdfs:subClassOf> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://schema.org/identifier"/> <owl:qualifiedCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger">1</owl:qualifiedCardinality> <owl:onDataRange rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	GSNElement identifier exactly 1 xsd:string	Earlier version contained Claim, which is now changed to be a GSNElement.
IV.3				GSNElement identifier some xsd:string	
IV.4			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://schema.org/identifier"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> </rdfs:subClassOf> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://schema.org/identifier"/> <owl:qualifiedCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger">1</owl:qualifiedCardinality>	ArtefactReference identifier exactly 1 xsd:string	
IV.5				ArtefactReference identifier some xsd:string	

			<owl:onDataRange rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> </rdfs:subClassOf> </owl:Class>		
IV.6			~~<owl:Axiom> <owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2002/07/owl#disjointWith"/> <owl:annotatedTarget rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> <coreOrExtension>Core GSN</coreOrExtension> </owl:Axiom>~~	~~<ArtefactReference disjointWith Claim> coreOrExtension "Core GSN"~~	Claim and Artefact Reference can be marked as disjoint – i.e., either an element is a reference or a claim – but this is not explicity stated in the standard. Furthermore, it is not clear how to distinguish between Claim:Context and ArtefactReference:Context. Therefore, this rule is currently inactive.
IV.7	**Table 1:2-1 Core GSN Elements – Definition** A goal, rendered as a rectangle, presents a claim forming part of the argument.	17	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal">	Goal a Class	
IV.8			<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/>	Goal subClassOf Claim	
IV.9			<coreOrExtension xml:lang="en">Core GSN</coreOrExtension>	Goal coreOrExtension "Core GSN"	
IV.10			<renderedAs xml:lang="en">rectangle</renderedAs>	Goal renderedAs "rectangle"	
IV.11			<rdfs:label xml:lang="en">Goal</rdfs:label>	Goal label "Goal"	
IV.12			<skos:definition xml:lang="en">A goal, rendered as a rectangle, presents a claim forming part of the argument.</skos:definition> </owl:Class>	Goal definition "A goal …"	
IV.13	**Table 1:2-1 Core GSN Elements – Definition** A strategy, rendered as a parallelogram, describes the inference that exists between a goal and its supporting goal(s).	17	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy">	Strategy a Class	
IV.14			<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/>	Strategy subClassOf Claim	
IV.15			<rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"/>	Strategy inContextOf only (Assumption or Context or Justification)	
IV.16			<owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection">	Strategy supportedBy only Goal	
IV.17			<rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"/>	Strategy coreOrExtension "Core GSN"	
IV.18			</owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf>	Strategy renderedAs "parallelogram"	
IV.19			<rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"/> <owl:allValuesFrom rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> </owl:Restriction> </rdfs:subClassOf>	Strategy label "Strategy"	
IV.20			<coreOrExtension xml:lang="en">Core GSN</coreOrExtension> <renderedAs xml:lang="en">parallelogram</renderedAs> <rdfs:label xml:lang="en">Strategy</rdfs:label> <skos:definition xml:lang="en">A strategy, rendered as a parallelogram, describes the inference that exists between a goal and its supporting goal(s).</skos:definition> </owl:Class>	Strategy definition "A strategy …"	
IV.21	**Table 1:2-1 Core GSN Elements – Definition** A solution, rendered as a circle, presents a reference to an evidence item.	17	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Solution">	Solution a Class	
IV.22			<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/>	Solution subClassOf ArtefactReference	
IV.23			<coreOrExtension xml:lang="en">Core GSN</coreOrExtension> <renderedAs xml:lang="en">circle</renderedAs> <rdfs:label xml:lang="en">Solution</rdfs:label>	Solution coreOrExtension "Core GSN"	
IV.24			<skos:definition xml:lang="en">A solution, rendered as a circle, presents a reference to an evidence item.</skos:definition> </owl:Class>	Solution renderedAs "circle"	
IV.25				Solution label "Solution"	
IV.26				Solution definition "A solution …"	
IV.27	**Table 1:2-1 Core GSN Elements – Definition** A context, rendered as shown left, presents a contextual artefact. This can be a reference to contextual information, or a statement.	17	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context">	Context a Class	Context can be a Claim or an Artefact Reference (i.e., a class representing a reference which both Solution and Context fit). However, it is not clear how distinct should a Claim:Context be from ArtefactReference:Context. Clarification needed.
IV.28			<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/>	Context subClassOf ArtefactReference	
IV.29			<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/>	Context subClassOf Claim	
IV.30			<coreOrExtension xml:lang="en">Core GSN</coreOrExtension> <renderedAs xml:lang="en">rounded rectangle</renderedAs> <rdfs:label xml:lang="en">Context</rdfs:label>	Context coreOrExtension "Core GSN"	
IV.31			<skos:definition xml:lang="en">A context, rendered as shown left, presents a contextual artefact. This can be a reference to contextual information, or a statement.</skos:definition> </owl:Class>	Context renderedAs "rounded rectangle"	
IV.32				Context label "Context"	
IV.33				Context definition "A context …"	
IV.34	**Table 1:2-1 Core GSN Elements – Definition**	17	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption">	Assumption a Class	

IV.35	An assumption, rendered as an oval with the letter ‘A’ at the top- or bottom-right, presents an intentionally unsubstantiated statement.		<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> <coreOrExtension xml:lang="en">Core GSN</coreOrExtension> <renderedAs xml:lang="en">oval</renderedAs> <rdfs:label xml:lang="en">Assumption</rdfs:label> <skos:definition xml:lang="en">An assumption, rendered as an oval with the letter 'A' at the top- or bottom-right, presents an intentionally unsubstantiated statement.</skos:definition> </owl:Class>	Assumption subClassOf Claim	
IV.36				Assumption coreOrExtension "Core GSN"	
IV.37				Assumption renderedAs "oval"	
IV.38				Assumption label "Assumption"	
IV.39				Assumption definition "An assumption …”	
IV.40	**Table 1:2-1 Core GSN Elements – Definition** A justification, rendered as an oval with the letter ‘J’ at the top- or bottom-right, presents a statement of rationale.	18	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"> <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> <coreOrExtension xml:lang="en">Core GSN</coreOrExtension> <renderedAs xml:lang="en">oval</renderedAs> <rdfs:label xml:lang="en">Justification</rdfs:label> <skos:definition xml:lang="en">A justification, rendered as an oval with the letter 'J' at the top- or bottom-right, presents a statement of rationale.</skos:definition> </owl:Class>	Justification a Class	
IV.41				Justification subClassOf Claim	
IV.42				Justification coreOrExtension "Core GSN"	
IV.43				Justification renderedAs "oval"	
IV.44				Justification label "Justification"	
IV.45				Justification definition "A justification …”	
IV.46	**Table 1:2-1 Core GSN Elements – Definition** Undeveloped element decorator, rendered as a hollow diamond applied to the bottom centre of an element, indicates that a line of argument has not been developed.	18	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#undeveloped"> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension> <renderedAs>hollow diamond</renderedAs> <rdfs:label xml:lang="en">undeveloped</rdfs:label> <skos:definition xml:lang="en">Undeveloped element decorator, rendered as a hollow diamond applied to the bottom centre of an element, indicates that a line of argument has not been developed.</skos:definition> </owl:DatatypeProperty>	undeveloped a DatatypeProperty	
IV.47				undeveloped range boolean	
IV.48				undeveloped coreOrExtension "Argument Pattern Extension"	
IV.49				undeveloped renderedAs "hollow diamond"	
IV.50				undeveloped label "undeveloped"	
IV.51				undeveloped definition "Undeveloped …”	
IV.52	It can apply to goals (as below) and strategies.	18	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#undeveloped"> <rdfs:domain> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy"/> </owl:unionOf> </owl:Class> </rdfs:domain> </owl:DatatypeProperty>	undeveloped domain (Goal or Strategy)	
IV.53	**Table 1:2-2 Core GSN Relationships - Definition** SupportedBy, rendered as a line with a solid arrowhead, allows support relationships between elements to be documented.	18	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"> <coreOrExtension xml:lang="en">Core GSN</coreOrExtension> <rdfs:label xml:lang="en">supported by</rdfs:label> <skos:definition xml:lang="en">SupportedBy, rendered as a line with a solid arrowhead, allows support relationships between elements to be documented.</skos:definition> </owl:ObjectProperty>	supportedBy a ObjectProperty	
IV.54				supportedBy coreOrExtension "Core GSN"	
IV.55				supportedBy label "supported by"	
IV.56				supportedBy definition "SupportedBy, …”	
IV.57	Permitted ‘supported by’ connections are: goal-to-goal, goal-to-strategy, goal-to-solution, strategy to goal.	18	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy"/> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Solution"/> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy"/> </owl:ObjectProperty>	supportedBy domain (Goal or Strategy)	
IV.58				supportedBy range Goal	
IV.59				supportedBy range Module	
IV.60				supportedBy range Solution	
IV.61				supportedBy range Strategy	
IV.62			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Solution"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	Goal supportedBy only (Goal or Solution or Strategy)	In addition to defining the “domain” and “range” for the property, OWL restriction axioms were added on “Goal” and “Strategy” for validation purposes.
IV.63			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"/> <owl:allValuesFrom rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> </owl:Restriction> </owl:Class>	Strategy supportedBy only Goal	
IV.64	**Table 1:2-2 Core GSN Relationships - Definition**	18	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf">	inContextOf a ObjectProperty	

IV.65	InContextOf, rendered as a line with a hollow arrowhead, declares a contextual relationship.		<coreOrExtension xml:lang="en">Core GSN</coreOrExtension> <rdfs:label xml:lang="en">in context of</rdfs:label> <skos:definition xml:lang="en">InContextOf, rendered as a line with a hollow arrowhead, declares a contextual relationship.</skos:definition> </owl:ObjectProperty>	inContextOf coreOrExtension "Core GSN" inContextOf label "in context of" inContextOf definition "InContextOf, …"	
IV.66					
IV.67					
IV.68	Permitted 'in context of' connections are: goal-to-context, goal-to-assumption, goal-to-justification, strategy-to-context, strategy-to-assumption and strategy-to-justification.	18	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy"/> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"/> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"/> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> </owl:ObjectProperty>	inContextOf domain (Goal or Strategy) inContextOf range (Assumption or Context or Justification or Module)	
IV.69					
IV.70			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	Goal inContextOf only (Assumption or Context or Justification)	In addition to defining of "domain" and "range" for the property, OWL restriction axioms were added on "Goal" and "Strategy"
IV.71			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy"> <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </owl:Class>	Strategy inContextOf only (Assumption or Context or Justification)	
IV.72	1:2.2.2 A GSN goal structure is a directed acyclic graph. This means that the graph does not allow loops, although one element can have multiple parents and children.	19	rdf:subject(?R1, ?A) ^ rdf:predicate(?R1, ?P) ^ rdf:object(?R1, ?B) ^ rdf:subject(?R2, ?B) ^ rdf:predicate(?R2, ?P) ^ rdf:object(?R2, ?C) ^ rdf:subject(?R3, ?C) ^ rdf:predicate(?R3, ?P) ^ rdf:object(?R3, ?A) -> gsn:valid(?R1, false) ^ gsn:valid(?R2, false) ^ gsn:valid(?R3, false)	**IF** (?R1 has subject ?A **AND** ?R1 has predicate ?P **AND** ?R1 has object ?B) **AND** (?R2 has subject ?B **AND** ?R2 has predicate ?P **AND** ?R2 has object ?C) **AND** (?R3 has subject ?C **AND** ?R3 has predicate ?P **AND** ?R3 has object ?A) **THEN** ?R1 is not valid **AND** ?R2 is not valid **AND** ?R3 is not valid	Direct prohibited relations are handled by making the property asymmetric. Cycles are handled by checking for the same directed relationship between three different elements.
IV.73			~~DELETE { ?start gsn:valid ?oldStartValue. ?middle gsn:valid ?oldMiddleValue.}~~ ~~INSERT { ?start gsn:valid false . ?middle gsn:valid false .}~~ ~~WHERE {~~ ~~?start gsn:supportedBy+ ?middle .~~ ~~?middle gsn:supportedBy+ ?start .~~ ~~}~~	~~**IF** ?start is supported by (some sequence of paths until) ?middle **AND** ?middle is supported by (some sequence of paths until) ?start **THEN** ?start is not valid **AND** ?middle is not valid~~	SPARQL allows for recursive checking, however, this rule is not active in this version to reduce dependencies in the ontology.
IV.74	SupportedBy relationships shall not be constructed so as to directly or indirectly allow a goal to support itself.	19	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"> <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AsymmetricProperty"/> <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#IrreflexiveProperty"/> </owl:ObjectProperty>	supportedBy a AsymmetricProperty supportedBy a IrreflexiveProperty	By defining the properties as irreflexive, it means an element cannot support or contextualize itself. By defining the properties as asymmetric, it means that element A cannot support and also be supported by some element B at the same time.
IV.75					
IV.76	Similarly, InContextOf relationships shall not be constructed so as to directly or indirectly allow a goal to provide its own context.	19	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"> <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AsymmetricProperty"/> <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#IrreflexiveProperty"/> </owl:ObjectProperty>	inContextOf a AsymmetricProperty inContextOf a IrreflexiveProperty	
IV.77					
IV.78	1:2.2.4 (…) G1 may be referred to as the parent goal, whilst G2 and G3 would commonly be referred to as 'supporting goals', 'sub-goals' or 'child goals' of G1.	19	-	-	This is an emergent property of the case. These positional aliases (i.e., "parent", "child", "supporting" and "sub-") are not made explicit in this version of the ontology.
IV.79		19	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#true">	true a DatatypeProperty	

IV.80	1:2.2.5 The structure shown in Figure 1:2-2 also asserts that if the claims presented in Goals G2 and G3 are true, this is sufficient to establish that the claim in Goal G1 is true.		<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension xml:lang="en">Core GSN</coreOrExtension> <rdfs:label xml:lang="en">true</rdfs:label> </owl:DatatypeProperty>	true domain Goal true range boolean true label “true” true coreOrGSN “Core GSN”	Since claims can be valid but untrue, “true” is represented as a separate data property.
IV.81					
IV.82					
IV.83					
IV.84			gsn:true(?A, false) ^ gsn:Goal(?A) ^ gsn:supportedBy(?B, ?A) ^ gsn:true(?B, true) -> gsn:valid(?B, false)	**IF** ?A is <u>not</u> true **AND** ?A is a Goal **AND** ?B is supported by ?A **AND** ?B is true **THEN** ?B is <u>not</u> valid	
IV.85			gsn:true(?A, true) ^ gsn:Goal(?A) ^ gsn:supportedBy(?C, ?A) ^ gsn:true(?B, true) ^ gsn:Goal(?B) ^ gsn:supportedBy(?C, ?B) ^ gsn:true(?C, true) -> gsn:valid(?C, false)	**IF** ?A is true **AND** ?A is a Goal **AND** ?C is supported by ?A **AND** ?B is <u>not</u> true **AND** ?B is a Goal **AND** ?C is supported by ?B **AND** ?C is true **THEN** ?C is <u>not</u> valid	
IV.86			~~DELETE { ?G1 gsn:true ?oldValue . }~~ ~~INSERT { ?G1 gsn:true true . }~~ ~~WHERE {~~ ~~?G1 gsn:supportedBy ?G2 .~~ ~~?G1 gsn:supportedBy ?G3 .~~ ~~?G2 gsn:true true .~~ ~~?G3 gsn:true true .~~ ~~FILTER NOT EXISTS {~~ ~~?G1 gsn:supportedBy ?Gx .~~ ~~?Gx gsn:true false .~~ ~~}~~ ~~OPTIONAL { ?G1 gsn:true ?oldValue. }~~ ~~}~~	~~**IF** ?G1 is supported by ?G2 **AND** ?G1 is supported by ?G3 **AND** ?G2 is true **AND** ?G3 is true **AND** ?G1 is not supported by ?Gx **AND** ?Gx is not true~~ ~~**THEN** ?G1 is true~~	SPARQL supports negation, but to reduce the number of dependencies, the rule is made inactive for this version.
IV.87			~~gsn:true(?A, true) ^ gsn:true(?B, true) ^ gsn:supportedBy(?C, ?A) ^ gsn:supportedBy(?C, ?B) -> gsn:true(?C, true)~~	~~**IF** ?A is true **AND** ?B is true **AND** ?C is supported by ?A **AND** ?C is supported by ?B~~ ~~**THEN** ?C is true~~	This rule is deprecated, because the reasoner would evaluate C as true if A and B are true, regardless if more elements (e.g., D, E, …) which are false support C also.
IV.88	1:2.2.7 Figure 1:2-4 represents the use of a reference to an evidence item to support a claim. 1:2.2.8 This structure represents an evidential relationship that asserts that the evidence referred to in the solution (Sn1) is sufficient to establish the truth of the claim made in the goal (G1).	21	gsn:true(?A, true) ^ gsn:Solution(?A) ^ gsn:supportedBy(?B, ?A) -> gsn:true(?B, true)	**IF** ?A is true **AND** ?A is a Solution **AND** ?B is supported by ?A **THEN** ?B is true	Beyond a simple propagation, the assertion of truthfulness is implicit and left to the user to validate, since that (i.e., the claim-evidence relation) is the underlying purpose of assurance cases.
IV.89			gsn:true(?A, false) ^ gsn:Solution(?A) ^ gsn:supportedBy(?B, ?A) -> gsn:true(?B, false)	**IF** ?A is <u>not</u> true **AND** ?A is a Solution **AND** ?B is supported by ?A **THEN** ?B is <u>not</u> true	
IV.90	1:2.2.9 (…) It is noted that the evidential relationship between the goal and its supporting evidence is provided by the indivisible combination of the two ‘SupportedBy’ relationships.	22	gsn:valid(?A, true) ^ gsn:supportedBy(?C, ?A) ^ gsn:valid(?B, false) ^ gsn:supportedBy(?C, ?B) -> gsn:valid(?C, false)	**IF** ?A is valid **AND** ?C is supported by ?A **AND** ?B is <u>not</u> valid **AND** ?C is supported by ?B **THEN** ?C is <u>not</u> valid	Unclear how to implement the concept of indivisibility in the ontology, beyond a simple AND operator.
IV.91			gsn:true(?A, true) ^ gsn:Solution(?A) ^ gsn:supportedBy(?C, ?A) ^ gsn:true(?B, false) ^ gsn:Solution(?B) ^ gsn:supportedBy(?C, ?B) -> gsn:true(?C, false)	**IF** ?A is true **AND** ?A is a Solution **AND** ?C is supported by ?A **AND** ?B is <u>not</u> true **AND** ?B is a Solution **AND** ?C is supported by ?B **THEN** ?C is <u>not</u> true	
IV.92	1:2.2.10 Claims can only be asserted to be true in a specified context. Context elements can be used in GSN to make this relationship clear.	22	~~gsn:Relationship(?R) ^ rdf:subject(?R, ?A) ^ rdf:predicate(?R, ?O) ^ gsn:inContextOf(?O) ^ rdf:object(?R, ?B) ^ gsn:true(?R, false) -> gsn:true(?A, false)~~	~~**IF** ?R is a Relationship **AND** ?R has subject ?A **AND** ?R has predicate “in context of” **AND** ?R has object ?B **AND** ?R is <u>not</u> true~~ ~~**THEN** ?A is <u>not</u> true~~	Given that ‘Context’ cannot be evaluated as true or false, the interpretation of this rule is that an “inContextOf” relation can be evaluated as true or false (e.g., if it is falsely asserted that a claim is true in a given context). *EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rule formulated instead.*
IV.93			gsn:Relationship(?R) ^ rdf:subject(?R, ?A) ^ gsn:inContextOf(?A, ?B) ^ rdf:object(?R, ?B) ^ gsn:true(?R, false) -> gsn:true(?A, false)	**IF** ?R is a Relationship **AND** ?R has subject ?A **AND** ?A is in context of ?B **AND** ?R has object ?B **AND** ?R is <u>not</u> true **THEN** ?A is <u>not</u> true	
IV.94			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"> <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"/> <owl:someValuesFrom rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	Goal inContextOf some Context	According to the interpretation of this point, a ‘Goal’ must have at least one associated ‘Context’. However, given the contradictory examples provided in other normative and non-normative sections of the standard, it is not clear whether this constraint should apply to all claims, including ‘Context’ class itself.
IV.94	1:2.2.11 Where used, contexts define or constrain the scope over which the claim is made. Since a contextual statement makes an assertion in the argument structure, nothing in the supporting argument for the goal to which the context is applied should contradict or undermine the relationship between the goal and the context.	22	-	-	Unclear how to apply the contradiction or undermining evaluation automatically, and without the additional elements from the Dialectic Extension.
IV.95	1:2.2.16 As before, since a contextual statement makes an assertion in the argument structure, nothing in the	23			

	supporting argument deriving from the strategy to which the context is applied should contradict or undermine the relationship between the strategy and the context.				
IV.96	1:2.2.11 (…) Context is taken to be connected to the entirety of the argument supporting the referenced element. Therefore, it is not necessary to restate the context in the supporting argument.	22	gsn:Context(?A) ^ gsn:inContextOf(?B, ?A) ^ gsn:supportedBy(?B, ?C) ^ gsn:Claim(?C) -> gsn:inContextOf(?C, ?A)	**IF** ?A is a Context **AND** ?B is in context of ?A **AND** ?B is supported by ?C **AND** ?C is a Claim **THEN** ?C is in context of ?A	
IV.97	1:2.2.16 (…) Context is taken to be connected to the entirety of the argument supporting the referenced element. Therefore, it is not necessary to restate the context in the supporting argument.	23			
IV.98	1:2.2.12 An assumption applied to a goal declares an assumption made in stating the claim. The meaning of the structure in Figure 1:2-7 is that the claim in goal G1 is asserted in a context where the assumption A1 is true.	22	gsn:Assumption(?A) ^ gsn:true(?A, false) ^ gsn:inContextOf(?B, ?A) -> gsn:true(?B, false)	**IF** ?A is an Assumption **AND** ?A is <u>not</u> true **AND** ?B is in context of ?A **THEN** ?B is <u>not</u> true	Given that a true 'Assumption' is a precondition for a true 'Goal', when they are linked, the inverse then becomes a rule.
IV.99	1:2.2.13 An assumption is an unsubstantiated statement. Having connected an assumption to a goal G1, the assumption is taken to be connected to the entirety of the relevant argument. Therefore, it is not necessary to restate the assumption in the relevant argument.	23	gsn:Assumption(?A) ^ gsn:inContextOf(?B, ?A) ^ gsn:supportedBy(?B, ?C) ^ gsn:Claim(?C) -> gsn:inContextOf(?C, ?A)	**IF** ?A is an Assumption **AND** ?B is in context of ?A **AND** ?B is supported by ?C **AND** ?C is a Claim **THEN** ?C is in context of ?A	
IV.100	1:2.2.17 (…) Having connected an assumption to a strategy S1, the assumption is taken to be connected to the entirety of the argument resulting from S1. Therefore, it is not necessary to restate the assumption in the supporting argument.	24			
IV.101	1:2.2.14 Figure 1:2-8 shows the connection of a justification to a goal. A justification does not alter the meaning of the claim made in the goal, but provides rationale for its inclusion or its phrasing. Unlike assumptions, justifications are not taken to be connected to the entirety of the argument supporting the referenced goal. They are local to the element to which they are linked. Should an equivalent justification be required elsewhere in the argument, it will need to be re stated or re-linked.	23	-	-	This lack of propagation means that it follows the "open world" approach – i.e., justification linked to one element is not necessarily linked to another supporting element.
IV.102	1:2.2.19 A justification applies to the element to which it is connected. Should an equivalent justification be required elsewhere in the argument, it will need to be re-stated or re-linked.	24			
IV.103	1:2.2.20 GSN structures can become large and it is often convenient to illustrate fragments of the argument structure in separate diagrams. To be able to convey that the argument continues in, or is a continuation from, a separate diagram an optional 'off diagram' decorator may be applied to elements at the top or bottom of each diagram.	24	-	-	This is a property of any user interface used for visualization, and not necessarily of the ontology. Subsequent versions may include a rule for the maximum amount of nodes and edges to be visualized on any given page.
IV.104	1:2.3.1 This section presents a series of simple rules which govern the grammatical structure of statements used in GSN elements.	25	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#statement">	statement a DatatypeProperty	The 'statement' datatype property is defined for the text contained in the element. It is datatype instead of annotation, because it is needed for reasoning. The 'label' property is reserved for visualization purposes.
IV.105			<rdfs:domain>	statement domain (Assumption or Context or Goal or Justification or Solution or Strategy)	
IV.106			<owl:Class>	statement coreOrExtension "Core GSN"	
IV.107			<owl:unionOf rdf:parseType="Collection">	statement label "statement"	
IV.108			<rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Solution"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy"/> </owl:unionOf> </owl:Class> </rdfs:domain> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> <coreOrExtension>Core GSN</coreOrExtension> <rdfs:label xml:lang="en">statement</rdfs:label> </owl:DatatypeProperty>	statement range string	
IV.109	1:2.3.2 GSN goals document the claims made in the argument (i.e. premises and conclusions). Each goal shall contain a single goal statement, expressed as a proposition in the form of a noun phrase + verb-phrase sentence.	25	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"> <rdfs:subClassOf rdf:nodeID="genid101"/> </owl:Class> <owl:Restriction rdf:nodeID="genid101"> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#statement"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> <owl:Axiom> <owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2000/01/rdf-schema#subClassOf"/>	Goal statement some xsd:string	Current version of the ontology only provides the grammar rule as a guideline for the user, and does not support automatic checks of correctness. Checks could later be added by explicating each noun- and verb-phrase, and verifying that the element has the right combination.
IV.110				<Goal statement some xsd:string> definition "GSN goals …"	

			<owl:annotatedTarget rdf:nodeID="genid101"/> <skos:definition xml:lang="en">GSN goals document the claims made in the argument (i.e. premises and conclusions). Each goal shall contain a single goal statement, expressed as a proposition in the form of a noun phrase + verb-phrase sentence.</skos:definition> </owl:Axiom>		
IV.111 IV.112	1:2.3.3 GSN strategy statements describe the reasoning that connects parent goals and their supporting goals, but the core claims and the structure connecting those claims remain unchanged. Strategy statements contain a brief description of the argument approach.	25	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy"> <rdfs:subClassOf rdf:nodeID="genid156"/> </owl:Class> <owl:Restriction rdf:nodeID="genid156"> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#statement"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> <owl:Axiom> <owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2000/01/rdf-schema#subClassOf"/> <owl:annotatedTarget rdf:nodeID="genid156"/> <skos:definition xml:lang="en">GSN strategy statements describe the reasoning that connects parent goals and their supporting goals, but the core claims and the structure connecting those claims remain unchanged. Strategy statements contain a brief description of the argument approach.</skos:definition> </owl:Axiom>	Strategy statement some xsd:string <Strategy statement some xsd:string> definition "GSN strategy …"	
IV.113 IV.114	1:2.3.4 GSN solutions make no claim, but are simply references to evidence items that provide support for a particular claim. They shall therefore be stated as noun-phrases.	25	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Solution"> <rdfs:subClassOf rdf:nodeID="genid152"/> </owl:Class> <owl:Restriction rdf:nodeID="genid152"> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#statement"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> <owl:Axiom> <owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Solution"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2000/01/rdf-schema#subClassOf"/> <owl:annotatedTarget rdf:nodeID="genid152"/> <skos:definition xml:lang="en">GSN solutions make no claim, but are simply references to evidence items that provide support for a particular claim. They shall therefore be stated as noun-phrases.</skos:definition> </owl:Axiom>	Solution statement some xsd:string <Solution statement some xsd:string> definition "GSN solutions …"	
IV.115 IV.116	1:2.3.5 Two kinds of GSN context statement exist. Where a context statement is a reference to an artefact of some kind, which informs the reasoning step, the context statement shall be expressed as a noun phrase. 1:2.3.5 (…) Where a context statement draws attention to explanatory contextual information (such as the definition of some term), this information shall be stated briefly using complete sentences of a noun-phrase + verb-phrase structure.	25 25	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"> <rdfs:subClassOf rdf:nodeID="genid82"/> </owl:Class> <owl:Restriction rdf:nodeID="genid82"> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#statement"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> <owl:Axiom> <owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2000/01/rdf-schema#subClassOf"/> <owl:annotatedTarget rdf:nodeID="genid82"/> <skos:definition xml:lang="en"> Two kinds of GSN context statement exist. Where a context statement draws attention to explanatory contextual information (such as the definition of some term), this information shall be stated briefly using complete sentences of a noun-phrase + verb-phrase structure. Where a context statement is a reference to an artefact of some kind, which informs the reasoning step, the context statement shall be expressed as a noun phrase.</skos:definition> </owl:Axiom>	Context statement some xsd:string <Context statement some xsd:string> definition "Two kinds …"	
IV.117 IV.118 IV.119 IV.120	1:2.3.6 GSN assumptions and justifications provide additional information necessary for the correct understanding of the argument. This information is stated as fully as necessary, using complete sentences in the form noun phrase + verb phrase.	25	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"> <rdfs:subClassOf rdf:nodeID="genid103"/> </owl:Class> <owl:Restriction rdf:nodeID="genid103"> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#statement"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> <owl:Axiom> <owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2000/01/rdf-schema#subClassOf"/> <owl:annotatedTarget rdf:nodeID="genid103"/> <skos:definition xml:lang="en">GSN justifications provide additional information necessary for the correct understanding of the argument. This information is stated as fully as necessary, using complete sentences in the form noun phrase + verb phrase.</skos:definition> </owl:Axiom> <owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"> <rdfs:subClassOf rdf:nodeID="genid78"/> </owl:Class> <owl:Restriction rdf:nodeID="genid78"> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#statement"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> <owl:Axiom>	Justification statement some xsd:string <Justification statement some xsd:string> definition "GSN justifications …" Assumption statement some xsd:string <Assumption statement some xsd:string> definition "GSN assumptions …"	

			<owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2000/01/rdf-schema#subClassOf"/> <owl:annotatedTarget rdf:nodeID="genid78"/> <skos:definition xml:lang="en">GSN assumptions provide additional information necessary for the correct understanding of the argument. This information is stated as fully as necessary, using complete sentences in the form noun phrase + verb phrase.</skos:definition> </owl:Axiom>		
IV.121 IV.122 IV.123 IV.123 IV.124 IV.125 IV.126 IV.127	-	-	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"> <rdfs:subClassOf rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Statement"/> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#object"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#predicate"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#challenges"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#subject"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Defeater"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> <coreOrExtension>Core GSN</coreOrExtension> <rdfs:label xml:lang="en">Relationship</rdfs:label> </owl:Class>	Relationship a Class Relationship subClassOf Statement Relationship coreOrExtension "Core GSN" Relationship subject only (Defeater or Goal or Strategy) ~~Relationship predicate only (challenges or 'in context of' or 'supported by')~~ Relationship object only ('Artefact Reference' or Claim) Relationship coreOrExtension “Core GSN” Relationship label "Relationship"	In order to be able to execute rules (e.g., if X supportedBy Y is true, do Z) or make assertions (e.g., X supportedBy Y is invalid) about specific relationships, we need to reify that triple into a Relationship class. This is an ontology-specific assertion. *EDIT 21-02-25: Punning disabled because of conflict with SWRL rules.*
IV.128			~~<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"/>~~	~~inContextOf a Class~~	Object properties are intentionally asserted as classes (“punning”), to allow connection to Relationships via rdf:predicate. *EDIT 21-02-25: Punning disabled because of conflict with SWRL rules.*
IV.129			~~<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"/>~~	~~supportedBy a Class~~	
IV.130			~~gsn:supportedBy(?A, ?B) ^ swrlx:makeOWLThing(?B, ?R) -> gsn:Relationship(?R) ^ rdf:subject(?R, ?A) ^ rdf:predicate(?R, ?O) ^ gsn:supportedBy(?O) ^ rdf:object(?R, ?B)~~	~~**IF** ?A is supported by ?B **AND DO** (for every ?B create ?R) **THEN** ?R is a Relationship **AND** ?R has subject ?A **AND** ?R has predicate “supported by” **AND** ?R has object ?B~~	Triples containing “supportedBy” and “inContextOf” are automatically reified. *EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rules are defined instead.*
IV.131			gsn:supportedBy(?A, ?B) ^ swrlx:makeOWLThing(?B, ?R) -> gsn:Relationship(?R) ^ rdf:subject(?R, ?A) ^ rdf:object(?R, ?B)	**IF** ?A is supported by ?B **AND DO** (for every ?B create ?R) **THEN** ?R is a Relationship **AND** ?R has subject ?A **AND** ?R has object ?B	
IV.132			~~gsn:inContextOf(?A, ?B) ^ swrlx:makeOWLThing(?B, ?R) -> gsn:Relationship(?R) ^ rdf:subject(?R, ?A) ^ rdf:predicate(?R, gsn:inContextOf) ^ rdf:object(?R, ?B)~~	~~**IF** ?A is in context of ?B **AND DO** (for every ?B create ?R)~~	

				~~**THEN** ?R is a Relationship **AND** ?R has subject ?A **AND** ?R has predicate "in context of" **AND** ?R has object ?B~~	
IV.133			gsn:inContextOf(?A, ?B) ^ swrlx:makeOWLThing(?B, ?R) -> gsn:Relationship(?R) ^ rdf:subject(?R, ?A) ^ rdf:object(?R, ?B)	**IF** ?A is in context of ?B **AND DO** (for every ?B create ?R) **THEN** ?R is a Relationship **AND** ?R has subject ?A **AND** ?R has object ?B	

Argument Pattern Extension

id	Item in GSN Community Standard	Page(s)	Item in GSN Ontology	Simplified Item in Ontology	Reason(s) for in-/exclusion
V.1	1:3.1.3 In cases where the elements defined in the following sections are used in the development of instantiations of the patterns to produce individual assurance arguments, it is important to ensure that they are all removed, or instantiated, in the final, delivered, version of the argument.	26	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#instantiationOf"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> <coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension> <rdfs:label xml:lang="en">instantiation of</rdfs:label> </owl:ObjectProperty>	instantiationOf a ObjectProperty	
V.2				instantiationOf domain (Argument or ArtefactReference or Claim)	
V.3				instantiationOf range Pattern	
V.4				instantiationOf coreOrExtension "Argument Pattern Extension"	
V.5				instantiationOf label "instantiation of"	
V.6			gsn:final(?A, true) ^ gsn:contains(?A, ?B) ^ gsn:contains(?A, ?C) ^ gsn:Pattern(?C) ^ gsn:instantiationOf(?B, ?C) -> gsn:valid(?A, false)	**IF** ?A is final **AND** ?A contains ?B **AND** ?A contains ?C **AND** ?C is Pattern **AND** ?B instantiationOf ?C **THEN** ?A is <u>not</u> valid	
V.7	1:3.1.3 (…) By exception, a final, delivered, version of the argument may be provided in a form that includes instantiable elements together with instantiation data as defined in section 1:3.5.	26	-	-	This is addressed under section 1:3.5.
V.8	1:3.2.1 This section describes the extensions to GSN defined in order to support two aspects of structural abstraction: • Multiplicity – generalised n-ary relationships between GSN elements;	26	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#multiple"> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension> <rdfs:label xml:lang="en">multiple</rdfs:label> </owl:DatatypeProperty>	multiple a DatatypeProperty	
V.9				multiple range boolean	
V.10				multiple coreOrExtension "Argument Pattern Extension"	
V.11				multiple label "multiple"	
V.12	1:3.2.1 (…) • Optionality – optional and alternative relationships between GSN elements.	26	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#optional"> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension> <rdfs:label xml:lang="en">optional</rdfs:label> </owl:DatatypeProperty>	optional a DatatypeProperty	
V.13				optional range boolean	
V.14				optional coreOrExtension "Argument Pattern Extension"	
V.15				optional label "optional"	
V.16	1:3.2.2 Table 1:3-1 illustrates the extensions made to GSN to facilitate the representation of multiplicity. These symbols are defined for use as decorators on all existing GSN relation types.	26	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#multiple"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> </owl:DatatypeProperty>	multiple domain Relationship	
V.17			<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#optional"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> </owl:DatatypeProperty>	optional domain Relationship	
V.18			<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#choice"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> </owl:DatatypeProperty>	choice domain Relationship	
V.19	1:3.2.2 (…) Multiplicity symbols can be used to describe how many instances of one element-type relate to another element.	26	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#maxCardinality"> <renderedAs>text</renderedAs> </owl:DatatypeProperty>	maxCardinality renderedAs "text"	Rendering added to min and max cardinality for the purpose of visualization.
V.20			<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#minCardinality"> <renderedAs>text</renderedAs> </owl:DatatypeProperty>	minCardinality renderedAs "text"	
V.21	**Table 1:3-1 – Definition** A solid ball is the symbol for multiple instantiations.	27	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#multiple"> <renderedAs>solid ball</renderedAs> </owl:DatatypeProperty>	multiple renderedAs "solid ball"	
V.22	**Table 1:3-1 – Definition** The optional label next to the ball indicates the cardinality of the relationship. It can be expressed as an instantiable parameter relevant to the argument.	27	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#maxCardinality"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#nonNegativeInteger"/> <coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension> <rdfs:label xml:lang="en">maximum cardinality</rdfs:label> </owl:DatatypeProperty>	maxCardinality a DatatypeProperty	
V.23				maxCardinality range nonNegativeInteger	
V.24				maxCardinality coreOrExtension "Argument Pattern Extension"	
V.25				maxCardinality label "maximum cardinality"	
V.26			<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#minCardinality"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#nonNegativeInteger"/> <coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension> <rdfs:label xml:lang="en">minimum cardinality</rdfs:label> </owl:DatatypeProperty>	maxCardinality a DatatypeProperty	
V.27				maxCardinality domain Relationship	
V.28				maxCardinality range nonNegativeInteger	
V.29				maxCardinality coreOrExtension "Argument Pattern Extension"	

V.30				maxCardinality label “maximum cardinality”	
V.31	**Table 1:3-1 – Definition** If no label is included then the cardinality can be any value from one upwards.	27	~~@prefix sh: <http://www.w3.org/ns/shacl#> .~~ ~~@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .~~ ~~@prefix gsn: <http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#> .~~ ~~gsn:RelationshipShape a sh:NodeShape ;~~ ~~sh:targetClass gsn:Relationship ;~~ ~~sh:property [~~ ~~sh:path gsn:minCardinality ;~~ ~~sh:datatype xsd:integer ;~~ ~~sh:minCount 1 ;~~ ~~sh:defaultValue 1 ;~~ ~~] .~~	~~Class **Relationship** must have **minCount** 1 xsd:integer or **defaultValue** “1” for property **minCardinality**.~~	SHACL constraints can add default values and advanced restrictions. However, this constraint is not active, in order to reduce the number of dependencies in this ontology version.
V.32	**Table 1:3-1 – Definition** If cardinality from zero onwards is required this should be explicitly declared e.g. 0..x declares that there may be zero to x branches (inclusive). It could also be written as 0≤n≤x.	27	-	-	This remains implicit, since there is already a rule indicating that default minCardinality is 1, and minCardinality can be a non-negative integer.
V.33	**Table 1:3-1 – Definition** A hollow ball indicates ‘optional’ instantiation,	27	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#optional"> <renderedAs xml:lang="en">hollow ball</renderedAs> </owl:DatatypeProperty>	optional renderedAs “hollow ball”	
V.34	**Table 1:3-1 – Definition** Optional instantiation means that the relationship and the argument below may or may not be instantiated.	27	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#optional"> <skos:definition xml:lang="en">Optional instantiation means that the relationship and the argument below may or may not be instantiated.</skos:definition> </owl:DatatypeProperty>	optional definition “Optional …”	
V.35	**Table 1:3-1 – Definition** A solid diamond is the symbol for Choice.	27	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#choice"> <renderedAs>solid diamond</renderedAs> </owl:DatatypeProperty>	choice renderedAs “solid diamond”	
V.36	**Table 1:3-1 – Definition** The optional label next to the diamond indicates the cardinality of the relationship.	27	-	-	See minCardinality and maxCardinality datatype properties.
V.37	**Table 1:3-1 – Definition** It can be expressed as an instantiable parameter relevant to the argument.	27	-	-	This is already allowed in the ontology.
V.38	**Table 1:3-1 – Definition** If no label is included then the cardinality can be any value from one to the number of supporting elements.	27	-	-	See SHACL rule under Table 1:3-1
V.39 V.40 V.41 V.42 V.43	1:3.2.3 The extension to GSN shown in Figure 1:3-1 shows the representation of structural choice A GSN choice can be used to denote possible alternatives in satisfying a relationship.	27	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#choice"> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension> <rdfs:label xml:lang="en">choice</rdfs:label> <skos:definition xml:lang="en">A GSN choice can be used to denote possible alternatives in satisfying a relationship.</skos:definition> </owl:DatatypeProperty>	choice a DatatypeProperty choice range boolean choice coreOrExtension “Argument Pattern Extension” choice label “choice” choice definition “A GSN choice …”	
V.44 V.45 V.46 V.47 V.48	**Table 1:3-2 – Definition** [TRIANGLE] This decorator denotes that the attached element remains to be instantiated, i.e. at some later stage the ‘abstract’ element needs to be replaced (instantiated) with a more concrete instance.	28	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#uninstantiated"> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension> <renderedAs>hollow triangle</renderedAs> <rdfs:label xml:lang="en">uninstantiated</rdfs:label> </owl:DatatypeProperty>	uninstantiated a DatatypeProperty uninstantiated range boolean uninstantiated coreOrExtension “Argument Pattern Extension” uninstantiated renderedAs “hollow triangle” uninstantiated label “uninstantiated”	
V.49	**Table 1:3-2 – Definition** This decorator can be applied to any GSN element type, and should be applied to the bottom centre of the element.	28	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#uninstantiated"> <rdfs:domain> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> </owl:unionOf> </owl:Class> </rdfs:domain> </owl:DatatypeProperty>	uninstantiated domain (Claim or Artefact Reference)	Location of the decorator is not indicated in this version. Future versions of the ontology can make visual rules explicit.
V.50	**Table 1:3-2 – Definition** The token to be instantiated is contained within curly brackets.	28	-	-	This is a visualization rule, and thus not included in the ontology.
V.51	**Table 1:3-2 – Definition** Decorators can be overlaid to denote that the attached element requires both further development and instantiation. The ‘undeveloped’ decorator was introduced in Table 1:2-1.	28	-	-	This is a visualization rule, and thus not included in the ontology. Nothing in the ontology restricts overlaying. Future versions of the ontology can make visual rules explicit.
V.52	**Table 1:3-2 – Definition** This combined decorator can be applied to any GSN element that the undeveloped decorator can be applied	28			

	to, and should be applied to the bottom centre of the element.				
V.53	1:3.4.1 A Pattern is not just the collection of GSN symbols.	28	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"> <coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension> <rdfs:label xml:lang="en">Pattern</rdfs:label> </owl:Class>		
V.54	1:3.4.1 (…) Additionally there should always be a supporting pattern description that defines the underlying intent and constraints on its use.	28, 29	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://schema.org/description"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2000/01/rdf-schema#Literal"/> </owl:Restriction> </owl:Class>	Pattern description some Literal	In order to enforce the weaker constraint (i.e., some textual description), the description annotation property is also defined as a datatype property. Future version can be revised to enforce the stronger constraint (i.e., description must contain components under subsections in 1:3.4.1).
V.55			<owl:DatatypeProperty rdf:about="http://schema.org/description"/>	description a DatatypeProperty	
V.56	1:3.4.1 (…) The format and presentation of the definition is not prescribed by this standard.	29	-	-	
V.57	1:3.4.1 (…) A pattern catalogue may be created to collate a series of patterns; where such a catalogue is created the structure and format of the definition should be consistent and each pattern's definition should have a unique {pattern identifier}.	29	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Catalogue">	Catalogue a Class	
V.58			<rdfs:subClassOf>	Catalogue contains only Pattern	
V.59			<owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#contains"/>	Catalogue coreOrExtension "Argument Pattern Extension"	
V.60			<owl:allValuesFrom rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> </owl:Restriction>	Catalogue label "Catalogue"	
V.61			</rdfs:subClassOf> <coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension> <rdfs:label xml:lang="en">Catalogue</rdfs:label> <skos:definition xml:lang="en">A pattern catalogue may be created to collate a series of patterns; where such a catalogue is created the structure and format of the definition should be consistent and each pattern's definition should have a unique {pattern identifier}.</skos:definition> </owl:Class>	Catalogue definition "A pattern …"	
V.62	1:3.4.1 (…) The following topics should be addressed in the pattern definition:	29	-	-	This is the stronger constraint, which is not enforced in this version.
V.63	**1:3.4.1 Name** 1:3.4.2 The pattern's name is the label by which the pattern can be identified and should meaningfully communicate the principle argument being presented.	29	-	-	This is already covered with the "label" annotation property.
V.64	1:3.4.2 (…) It may be accompanied by one or more aliases, which are an alternative identifiers by which the pattern may also be referred to.	29	-	-	This is already covered with the "altLabel" annotation property.
V.65	**1:3.4.2 Intent**	29	<owl:AnnotationProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#intent">	intent a AnnotationProperty	
V.66	1:3.4.3 The intent statement should state clearly what the pattern aims to achieve.		<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	intent coreOrExtension "Argument Pattern Extension"	
V.67			<rdfs:label xml:lang="en">intent</rdfs:label>	intent label "intent"	
V.68			<skos:definition xml:lang="en">The intent statement should state clearly what the pattern aims to achieve.</skos:definition>	intent definition "The intent …"	
V.69			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> </owl:AnnotationProperty>	intent domain Pattern	
V.70	**1:3.4.3 Motivation**	29	<owl:AnnotationProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#motivation">	motivation a AnnotationProperty	
V.71	1:3.4.4 The motivation statement can be used to state why the pattern was created. It could be expressed in terms of previous experiences e.g. as the abstraction of a successfully presented argument, or challenges addressed e.g. argument topics that are often incompletely or poorly addressed.		<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	motivation coreOrExtension "Argument Pattern Extension"	
V.72			<rdfs:label xml:lang="en">motivation</rdfs:label>	motivation label "motivation"	
V.73			<skos:definition xml:lang="en">The motivation statement can be used to state why the pattern was created. It could be expressed in terms of previous experiences e.g. as the abstraction of a successfully presented argument, or challenges addressed e.g. argument topics that are often incompletely or poorly addressed.</skos:definition>	motivation definition "The motivation …"	
V.74			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> </owl:AnnotationProperty>	motivation domain Pattern	
V.75	**1:3.4.4 Structure**	29	<owl:AnnotationProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#structure">	structure a AnnotationProperty	
V.76	1:3.4.5 The structure uses the structural and element abstraction notations to present the pattern, clearly indicating where the argument needs to be further developed or populated with details to instantiate the pattern for a specific case.		<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	structure coreOrExtension "Argument Pattern Extension"	
V.77			<rdfs:label xml:lang="en">structure</rdfs:label>	structure label "structure"	
V.78			<skos:definition xml:lang="en">The structure uses the structural and element abstraction notations to present the pattern, clearly indicating where the argument needs to be further developed or populated with details to instantiate the pattern for a specific case.</skos:definition>	structure definition "The structure …"	
V.79			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> </owl:AnnotationProperty>	structure domain Pattern	
V.80	**1:3.4.5 Participants**	29	<owl:AnnotationProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#participants">	participants a AnnotationProperty	This section can actually be a collection of descriptions of all "participating" elements. Next version of ontology can include "participatesIn" as an object property, a "description" annotation property for pattern elements, and a rule collating descriptions.
V.81	1:3.4.6 The participants section augments the structure by providing a description of each element. This can provide more complete descriptions, clarify the role of the element in the overall argument and emphasise the aspects that require development or instantiation.		<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	participants coreOrExtension "Argument Pattern Extension"	
V.82			<rdfs:label xml:lang="en">participants</rdfs:label>	participants label "participants"	
V.83			<skos:definition xml:lang="en">The participants section augments the structure by providing a description of each element. This can provide more complete descriptions, clarify the role of the element in the overall argument and emphasise the aspects that require development or instantiation.</skos:definition>	participants definition "The participants …"	
V.84			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> </owl:AnnotationProperty>	participants domain Pattern	
V.85	**1:3.4.6 Collaboration**	29	<owl:AnnotationProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#collaboration">	collaboration a AnnotationProperty	
V.86	1:3.4.7 The collaboration section should describe how elements of the pattern work together to achieve the		<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	collaboration coreOrExtension "Argument Pattern Extension"	
V.87			<rdfs:label xml:lang="en">collaboration</rdfs:label>	collaboration label "collaboration"	

V.88	desired effect, particularly where there are links that are not readily apparent from the argument structure.		<skos:definition xml:lang="en">The collaboration section should describe how elements of the pattern work together to achieve the desired effect, particularly where there are links that are not readily apparent from the argument structure.</skos:definition>	collaboration definition "The collaboration …"	
V.89			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> </owl:AnnotationProperty>	collaboration domain Pattern	
V.90	**1:3.4.7 Applicability**	30	<owl:AnnotationProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#applicability">	applicability a AnnotationProperty	Next version of the standard could address applicability by allowing an "inContextOf" relation between a pattern and context.
V.91	1:3.4.8 The applicability section should state under what circumstances the pattern can be applied, making clear the assumptions and principles underlying the pattern to avoid inappropriate application in a mismatched context. This section should record what contextual information is required in order to apply the pattern.		<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	applicability coreOrExtension "Argument Pattern Extension"	
V.92			<rdfs:label xml:lang="en">applicability</rdfs:label>	applicability label	
V.93			<skos:definition xml:lang="en">The applicability section should state under what circumstances the pattern can be applied, making clear the assumptions and principles underlying the pattern to avoid inappropriate application in a mismatched context. This section should record what contextual information is required in order to apply the pattern.</skos:definition>	applicability definition "The applicability …"	
V.94			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> </owl:AnnotationProperty>	applicability domain Pattern	
V.95	**1:3.4.8 Consequences**	30	<owl:AnnotationProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#consequences">	consequences a AnnotationProperty	Next version of the standard could address consequences by allowing an "inContextOf" relation between a pattern and assumption.
V.96	1:3.4.9 The consequences section should make clear what work remains after the pattern has been applied. This should highlight where further support to the argument is required, and assumptions that need to be discharged.		<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	consequences coreOrExtension "Argument Pattern Extension"	
V.97			<rdfs:label xml:lang="en">consequences</rdfs:label>	consequences label "consequences"	
V.98			<skos:definition xml:lang="en">The consequences section should make clear what work remains after the pattern has been applied. This should highlight where further support to the argument is required, and assumptions that need to be discharged.</skos:definition>	consequences definition "The consequences …"	
V.99			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> </owl:AnnotationProperty>	consequences domain Pattern	
V.100	**1:3.4.9 Implementation**	30	<owl:AnnotationProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#implementation">	implementation a AnnotationProperty	
V.101	1:3.4.10 The implementation section should communicate how the application of the pattern is carried out e.g. the order in which elements should be developed; communicate hints or techniques that may ease successful application; highlight common or recognised pitfalls with the application of the pattern; and record potential misinterpretation of the terms or concepts in the pattern.		<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	implementation coreOrExtension "Argument Pattern Extension"	
V.102			<rdfs:label xml:lang="en">implementation</rdfs:label>	implementation label "implementation"	
V.103			<skos:definition xml:lang="en">The implementation section should communicate how the application of the pattern is carried out e.g. the order in which elements should be developed; communicate hints or techniques that may ease successful application; highlight common or recognised pitfalls with the application of the pattern; and record potential misinterpretation of the terms or concepts in the pattern.</skos:definition>	implementation definition "The implementation …"	
V.104			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> </owl:AnnotationProperty>	implementation domain Pattern	
V.105	**1:3.4.10 Examples**	30	<owl:AnnotationProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#examples">	examples a AnnotationProperty	
V.106	1:3.4.11 It may be useful to provide example illustrations of the application of the pattern, particularly for more abstract patterns. Illustrations should include a typical case and can be supplemented with atypical cases where more than one example is provided.		<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	examples coreOrExtension "Argument Pattern Extension"	
V.107			<rdfs:label xml:lang="en">examples</rdfs:label>	examples label "examples"	
V.108			<skos:definition xml:lang="en">It may be useful to provide example illustrations of the application of the pattern, particularly for more abstract patterns. Illustrations should include a typical case and can be supplemented with atypical cases where more than one example is provided.</skos:definition>	examples definition "It may …"	
V.109			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> </owl:AnnotationProperty>	examples domain Pattern	
V.110	**1:3.4.11 Known uses**	30	<owl:AnnotationProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#knownUses">	knownUses a AnnotationProperty	Currently, there is nothing in the standard that would allow
V.111	1:3.4.12 It may be useful to provide references to known applications of the pattern. These can serve as additional examples.		<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	knownUses coreOrExtension "Argument Pattern Extension"	
V.112			<rdfs:label xml:lang="en">known uses</rdfs:label>	knownUses label "known uses"	
V.113			<skos:definition xml:lang="en">It may be useful to provide references to known applications of the pattern. These can serve as additional examples.</skos:definition>	knownUses definition "It may …"	
V.114			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> </owl:AnnotationProperty>	knownUses domain Pattern	
V.115	**1:3.4.12 Related patterns**	30	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#relatedTo">	relatedTo a ObjectProperty	
V.116	1:3.4.13 This section can be used to reference patterns that are related e.g. addressing the same intent in a different context.		<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/>	relatedTo domain Pattern	
V.117			<rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/>	relatedTo range Pattern	
V.118			<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	relatedTo coreOrExtension "Argument Pattern Extension"	
V.119			<rdfs:label xml:lang="en">related to</rdfs:label>	relatedTo label "related to"	
V.120			<skos:definition xml:lang="en">This section can be used to reference patterns that are related e.g. addressing the same intent in a different context.</skos:definition> </owl:ObjectProperty>	relatedTo definition "This section …"	
V.121	1:3.5.1 By exception, as an alternative to the obligation to instantiate all patterns elements a completed argument, instantiation can be by means a 'template argument' together with instantiation data. This can avoid producing multiple pages of GSN structure where the argument structure is highly repetitive when repeated over multiple aspects e.g. where an argument over requirement satisfaction for individual requirements appeals to different test cases, but otherwise is an identical argument.	30, 31	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Template">	Template a Class	
V.122			<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	Template coreOrExtension "Argument Pattern Extension"	
V.123			<rdfs:label xml:lang="en">Template</rdfs:label> </owl:Class>	Template label "Template"	
V.124	1:3.5.1 (…) Table 1:3-3 identifies an additional symbol used to indicate that the GSN argument is a template argument to be instantiated from instantiation data.	31	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#uninstantiated"> <renderedAs>hollow triangle</renderedAs> </owl:DatatypeProperty>	uninstantiated renderedAs "hollow triangle"	
V.125	1:3.5.2 A template argument is a special case of a pattern argument. It uses the core GSN and argument pattern extension to construct an argument structure which requires no further development.	31	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Template">	Template subClassOf Pattern	
V.126			<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Pattern"/> <skos:definition xml:lang="en">A template argument is a special case of a pattern argument. It uses the core GSN and argument pattern extension to construct an argument structure which requires no further development.</skos:definition> </owl:Class>	Template definition "A template …"	
V.127		31	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#published">	published a DatatypeProperty	

V.128	1:3.5.2 (…) The use of the 'undeveloped' decorator is not permitted within a template argument in its published form.		<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/>	published domain Argument	
V.129			<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>	published range boolean	
V.130			<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	published coreOrExtension "Argument Pattern Extension"	
V.131			<rdfs:label xml:lang="en">published</rdfs:label> </owl:DatatypeProperty>	published label "published"	
V.132			gsn:Template(?A) ^ rdf:subject(?R, ?A) ^ gsn:published(?A, true) ^ gsn:undeveloped(?R, true) -> gsn:valid(?A, false) ^ gsn:valid(?R, false)	**IF** ?A is a Template **AND** ?R has subject ?A **AND** ?A is published **AND** ?R is undeveloped **THEN** ?A is <u>not</u> valid **AND** ?R is <u>not</u> valid	Both the Template Argument and an element within the Template Argument are invalid if they contain the undeveloped decorator.
V.133			gsn:Template(?A) ^ gsn:contains(?A, ?B) ^ rdf:subject(?R, ?B) ^ gsn:published(?A, true) ^ gsn:undeveloped(?R, true) -> gsn:valid(?A, false) ^ gsn:valid(?R, false)	**IF** ?A is a Template **AND** ?A contains ?B **AND** ?R has subject ?B **AND** ?A is published **AND** ?R is undeveloped **THEN** ?A is <u>not</u> valid **AND** ?R is <u>not</u> valid	
V.134	1:3.5.2 (…) The instantiation data must cover all instantiable aspects including optionality, multiplicity and choice.	31	-	-	It is not clear how to implement this rule.
V.135	1:3.5.3 Where a template argument ends at an element other than a solution, that final element must exist elsewhere within the argument,	31	-	-	It is not clear how to implement this rule.
V.136	1:3.5.3 (…) or in the case of an away element, must be declared in instantiated form in the module interface.	31	-	-	It is not clear how to implement this rule.
V.137	1:3.5.4 Where a template argument and instantiation data is used, it must be possible to apply the instantiation, creating all instantiated versions and meet all the core GSN rules, including uniqueness of element identifiers.	31	-	-	It is not clear how to implement this rule.
V.138	1:3.5.4 (…) Uninstantiated identifiers in a template only need to be unique within the template.	31	-	-	It is not clear how to implement this rule.
V.139	**Table 1:3-3 – Definition** Instantiation Data Reference. This symbol indicates that the GSN argument below the attached element is to be instantiated as a template argument. It provides a reference to the information used to instantiate the template argument.	31	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#InstantiationDataReference">	InstantiationDataReference a Class	
V.140			<coreOrExtension>Argument Pattern Extension</coreOrExtension>	InstantiationDataReference coreOrExtension "Argument Pattern Extension"	
V.141			<renderedAs xml:lang="en">hollow triangle</renderedAs>	InstantiationDataReference renderedAs "hollow triangle"	
V.142			<rdfs:label xml:lang="en">Instantiation Data Reference</rdfs:label>	InstantiationDataReference label "Instantiation Data Reference"	
V.143			<skos:definition xml:lang="en">This symbol indicates that the GSN argument below the attached element is to be instantiated as a template argument. It provides a reference to the information used to instantiate the template argument.</skos:definition> </owl:DatatypeProperty>	InstantiationDataReference definition "This symbol …"	
V.144	**Table 1:3-3 – Definition** The symbol is not considered a GSN element as it does not form part of the argument.	31	-	-	Although IDR would conceptually fit as a subclass of "Artefact Reference", this sentence indicates that it should not be the case.
V.145	**Table 1:3-3 – Definition** It is attached to the top element of the template argument by a dotted line between the top edges of that element and the symbol (as shown below).	31	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#attachedTo">	attachedTo a ObjectProperty	
V.146			<rdf:type rdf:resource="http://www.w3.org/2002/07/owl#SymmetricProperty"/>	attachedTo a SymmetricProperty	
V.147			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#InstantiationDataReference"/>	attachedTo domain "InstantiationDataReference"	
V.148			<rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Template"/>	attachedTo range "Template"	
V.149			<coreOrExtension xml:lang="en">Argument Pattern Extension</coreOrExtension>	attachedTo coreOrExtension "Argument Pattern Extension"	
V.150			<renderedAs>dotted line</renderedAs>	attachedTo renderedAs "dotted line"	
V.151			<rdfs:label xml:lang="en">attached to</rdfs:label> </owl:ObjectProperty>	attachedTo label "attached to"	

Modular Extension

id	Item in GSN Community Standard	Page(s)	Item in GSN Ontology	Simplified Item in Ontology	Reason(s) for in-/exclusion
VI.1	1:4.1.1 Goal structures can be partitioned into separate, but interrelated, modules. This can allow the division of an overall goal structure into separate goal structures focusing on particular aspects of the overall argument. This section describes how GSN has been extended to represent modular arguments.	32	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module">	Module a Class	
VI.2			<coreOrExtension xml:lang="en">Modular Extension</coreOrExtension>	Module coreOrExtension "Modular Extension"	
VI.3			<rdfs:label xml:lang="en">Module</rdfs:label>	Module label "Module"	
VI.4			<skos:definition xml:lang="en">Goal structures can be partitioned into separate, but interrelated, modules. This can allow the division of an overall goal structure into separate goal structures focusing on particular aspects of the overall argument. A module may contain one or more arguments and may contain other modules.</skos:definition> </owl:Class>	Module definition "Goal structures …"	
VI.5	1:4.1.3 A module may contain one or more arguments and may contain other modules.	32	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#contains">	contains a ObjectProperty	
VI.6			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/>	contains domain Module	
VI.7			<rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/>	contains range (Argument or Module)	
VI.8					

VI.9			<rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> <coreOrExtension>Modular Extension</coreOrExtension> <rdfs:label xml:lang="en">contains</rdfs:label> </owl:ObjectProperty>	contains coreOrExtension “Modular Extension”	
VI.10				contains label “contains”	
VI.11			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#contains"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	Module contains only (Argument or Module)	
VI.12			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#contains"/> <owl:minQualifiedCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger">1</owl:minQualifiedCardinality> <owl:onClass rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	Module contains min 1 Argument	
VI.13	1:4.1.2 The concepts of ‘argument view’, ‘architecture view’, module interfaces and inter-module contracts are introduced.	32	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#View"> <coreOrExtension xml:lang="en">Modular Extension</coreOrExtension> <rdfs:label xml:lang="en">View</rdfs:label> </owl:Class>	View a Class	
VI.14				View coreOrExtension “Modular Extension”	
VI.15				View label “View”	
VI.16			<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#viewType"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#View"/> <rdfs:range> <rdfs:Datatype> <owl:oneOf> <rdf:Description> <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#List"/> <rdf:first>architecture</rdf:first> <rdf:rest> <rdf:Description> <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#List"/> <rdf:first>argument</rdf:first> <rdf:rest rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#nil"/> </rdf:Description> </rdf:rest> </rdf:Description> </owl:oneOf> </rdfs:Datatype> </rdfs:range> <coreOrExtension>Modular Extension</coreOrExtension> <rdfs:label xml:lang="en">view type</rdfs:label> </owl:DatatypeProperty>	viewType a DatatypeProperty	
VI.17				viewType domain View	
VI.18				viewType range oneOf {“argument”, “architecture”}	
VI.19				viewType coreOrExtension “Modular Extension”	
VI.20				viewType label “view type”	
VI.21	1:4.2.1 The argument view depicts the argument inside an individual module. The following elements are used in addition to the core GSN notation: • Away Goal; • Away Solution; • Away Context; • Away Assumption; • Away Justification. • Module Reference; • Contract Reference	32	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#away"> <rdfs:domain> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Solution"/> </owl:unionOf> </owl:Class> </rdfs:domain> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension>Modular Extension</coreOrExtension> <renderedAs>bisecting line in the lower half</renderedAs> <rdfs:label xml:lang="en">away</rdfs:label> </owl:DatatypeProperty>	away a DatatypeProperty	Away is defined as a datatype property.
VI.22				away domain (Assumption or Context or Goal or Justification or Solution)	
VI.23				away range boolean	
VI.24				away coreOrExtension “Modular Extension”	
VI.25				away renderedAs “bisecting line in the lower half”	
VI.26				away label “away”	

VI.27			gsn:Module(?M) ^ gsn:Module(?N) ^ gsn:contains(?M, ?A) ^ gsn:contains(?N, ?B) ^ swrlb:notEqual(?M, ?N) ^ gsn:supportedBy(?A, ?B) -> gsn:away(?B, true)	**IF** ?M is a Module **AND** ?N is a Module **AND** ?M contains ?A **AND** ?N contains ?B **AND** ?M is not equal to ?N **AND** ?A is supported by ?B **THEN** B is away	Because away is a property of an element supporting or contextualizing another element, only the former should have the.
VI.28			gsn:Module(?M) ^ gsn:Module(?N) ^ gsn:contains(?M, ?A) ^ gsn:contains(?N, ?B) ^ swrlb:notEqual(?M, ?N) ^ gsn:inContextOf(?A, ?B) -> gsn:away(?B, true)	**IF** ?M is a Module **AND** ?N is a Module **AND** ?M contains ?A **AND** ?N contains ?B **AND** ?M is not equal to ?N **AND** ?A is in context of ?B **THEN** B is away	
VI.29	1:4.2.2 (…) Note that each argument module has its own namespace for identifiers, thus two elements with the same element identifier can exist in different argument modules.	32	gsn:Module(?M) ^ gsn:contains(?M, ?A) ^ gsn:contains(?M, ?B) ^ schema:identifier(?A, ?ID) ^ schema:identifier(?B, ?ID) ^ swrlb:notEqual(?A, ?B) -> gsn:valid(?A, false) ^ gsn:valid(?B, false)	**IF** ?M is a Module **AND** ?M contains ?A **AND** ?M contains ?B **AND** ?A has identifier ?ID **AND** ?B has identifier ?ID **AND** ?A is not equal to ?B **THEN** ?A is not valid **AND** ?B is not valid	The open world assumption allows same identifiers, so we define a rule when it is not allowed (i.e., in the same module).
VI.30	1:4.2.2 (…) Element identifiers must be unique within a single argument module.	32			
VI.31	**Table 1:4-1 – Definition** An away goal reference is rendered as a rectangle with a bisecting line in the lower half of the rectangle. The area in the lower portion contains a miniature shaded module element symbol.	33	~~gsn:Goal(?A) ^ gsn:away(?A, true) -> gsn:renderedAs(?A, "rectangle with a bisecting line in the lower half of the rectangle")~~	~~**IF** ?A is a Goal **AND** ?A is away **THEN** ?A is rendered as "rectangle with a bisecting line in the lower half of the rectangle"~~	This "renderedAs" property is inserted through a rule that overwrites the class-level "renderedAs" assertion. Next ontology version can include data needed for visualizing directly. *EDIT 21-02-25: SWRL cannot assert annotations; rule disabled unless renderedAs should be redefined as a datatype property.*
VI.32	**Table 1:4-1 – Definition** An away solution, rendered as a semi-circle sitting on top of a rectangle (the semi-circle may be raised above the rectangle by extending its vertical extremes in a straight line).	33	~~gsn:Solution(?A) ^ gsn:away(?A, true) -> gsn:renderedAs(?A, "semi-circle sitting on top of a rectangle")~~	~~**IF** ?A is a Solution **AND** ?A is away **THEN** ?A is rendered as "semi-circle sitting on top of a rectangle"~~	This "renderedAs" property is inserted through a rule that overwrites the class-level "renderedAs" assertion. Next ontology version can include data needed for visualizing directly. *EDIT 21-02-25: SWRL cannot assert annotations; rule disabled unless renderedAs should be redefined as a datatype property.*
VI.33	**Table 1:4-1 – Definition** An away context, rendered as shown left, repeats a contextual artefact.	33	~~gsn:Context(?A) ^ gsn:away(?A, true) -> gsn:renderedAs(?A, "ellipse sitting on top of a rectangle")~~	~~**IF** ?A is a Context **AND** ?A is away **THEN** ?A is rendered as "ellipse sitting on top of a rectangle"~~	The shape description is interpreted, since the standard only refers to "as shown left". This "renderedAs" property is inserted through a rule that overwrites the class-level "renderedAs" assertion. Next ontology version can include data needed for visualizing directly. *EDIT 21-02-25: SWRL cannot assert annotations; rule disabled unless renderedAs should be redefined as a datatype property.*
VI.34	**Table 1:4-1 – Definition** An away assumption, rendered as a semi-ellipse sitting on top of a rectangle with the letter 'A' at the top-right (the semi-ellipse may be raised above the rectangle by extending its vertical extremes in a straight line).	33	~~gsn:Assumption(?A) ^ gsn:away(?A, true) -> gsn:renderedAs(?A, "semi-ellipse sitting on top of a rectangle with the letter 'A' at the top-right")~~	~~**IF** ?A is an Assumption **AND** ?A is away **THEN** ?A is rendered as "semi-ellipse sitting on top of a rectangle with the letter 'A' at the top-right"~~	This "renderedAs" property is inserted through a rule that overwrites the class-level "renderedAs" assertion. Next ontology version can include data needed for visualizing directly. *EDIT 21-02-25: SWRL cannot assert annotations; rule disabled unless renderedAs should be redefined as a datatype property.*
VI.35	**Table 1:4-1 – Definition** An away justification, rendered as a semi-ellipse sitting on top of a rectangle with the letter 'J' at the top-right (the semi-ellipse may be raised above the rectangle by extending its vertical extremes in a straight line).	33	~~gsn:Justification(?A) ^ gsn:away(?A, true) -> gsn:renderedAs(?A, "semi-ellipse sitting on top of a rectangle with the letter 'J' at the top-right")~~	~~**IF** ?A is a Justification **AND** ?A is away **THEN** ?A is rendered as "semi-ellipse sitting on top of a rectangle with the letter 'J' at the top-right"~~	This "renderedAs" property is inserted through a rule that overwrites the class-level "renderedAs" assertion. Next ontology version can include data needed for visualizing directly. *EDIT 21-02-25: SWRL cannot assert annotations; rule disabled unless renderedAs should be redefined as a datatype property.*
VI.36	An away goal reference repeats a claim presented in another argument module.	33	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#away"> <schema:description xml:lang="en">An away assumption repeats an assumption presented in another argument module and is typically used only in Contract Modules. An away context repeats a reference to context presented in another argument module. An away goal reference repeats a claim presented in another argument module. An away justification repeats a justification presented in another argument module and is typically used only in Contract Modules. An away solution repeats a reference to evidence items presented in another argument module.</schema:description> <skos:definition xml:lang="en">An away element reference repeats a claim or evidence presented in another argument module.</skos:definition> </owl:DatatypeProperty>	away description "An away …"	Adding annotations to more advanced domain and range expressions leads to an error, so these statements are added as description. For the definition, the statement is formulated with a reference to an element.
VI.37	An away solution repeats a reference to evidence items presented in another argument module.	33		away definition "An away …"	
VI.38	An away context repeats a reference to context presented in another argument module.	33			
VI.39	An away assumption repeats an assumption presented in another argument module and is typically used only in Contract Modules.	33			

VI.40	An away justification repeats a justification presented in another argument module and is typically used only in Contract Modules.	33			
VI.41	For all away elements defined above, the element has an identifier which is the {element identifier} of the referenced element in the module in which it was originally declared.	33	-	-	No assertion is needed, since nothing changes except the rendering for the away element.
VI.42	The <element statement> contains an exact repetition of the text of the referenced element.	33	-	-	
VI.43 VI.44	The {module identifier} is the identifier of the module in which the referenced element occurs.	33	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://schema.org/identifier"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> </rdfs:subClassOf> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://schema.org/identifier"/> <owl:qualifiedCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger">1</owl:qualifiedCardinality> <owl:onDataRange rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	Module identifier some string Module identifier exactly 1 string	
VI.45	**Table 1:4-1 – Definition** A module reference, rendered as a rectangle with a second smaller rectangle adjoining at the top left, presents a reference to a module containing an argument.	34	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"> <renderedAs>a rectangle with a second smaller rectangle adjoining at the top left</renderedAs> </owl:Class>	Module renderedAs „a rectangle ...“	
VI.46	Note that a module reference points to the totality of the argument contained in the referenced argument module, rather than just to an individual claim.	34	-	-	This is already implicit in the relation. No further restriction is made (conditional transitivity, e.g., if X supportedBy Y and Y contains Z, then X supportedBy Z).
VI.47	A module reference may be used in support and/or as context for an argument.	34	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> </owl:ObjectProperty>	supportedBy range Module	
VI.48			<owl:Axiom> <owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2000/01/rdf-schema#range"/> <owl:annotatedTarget rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> <coreOrExtension xml:lang="en">Modular Extension</coreOrExtension> </owl:Axiom>	<supportedBy range Module> coreOrExtension “Modular Extension”	
VI.49			<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> </owl:ObjectProperty>	inContextOf range Module	
VI.50			<owl:Axiom> <owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2000/01/rdf-schema#range"/> <owl:annotatedTarget rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> <coreOrExtension>Modular Extension</coreOrExtension> </owl:Axiom>	<inContextOf range Module> coreOrExtension “Modular Extension”	
VI.51	A module reference cannot be used within a contract module.	34	~~gsn:supportedBy(?A, ?M1) ^ gsn:Module(?M1) ^ gsn:contains(?M2, ?A) ^ gsn:Module(?M2) ^ gsn:contract(?M2, true) ^ rdf:subject(?R, ?A) ^ rdf:predicate(?R, ?O) ^ gsn:supportedBy(?O) ^ rdf:object(?R, ?M1) -> gsn:valid(?R, false)~~	~~**IF** ?A is supported by ?M1 **AND** ?M1 is a Module **AND** ?M2 contains ?A **AND** ?M2 is a Module **AND** ?M2 is a contract **AND** ?R has subject ?A **AND** ?R has predicate “supported by” **AND** ?R has object ?M1 **THEN** ?R is not valid~~	*EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rules are defined instead.*
VI.52			~~gsn:inContextOf(?A, ?M1) ^ gsn:Module(?M1) ^ gsn:contains(?M2, ?A) ^ gsn:Module(?M2) ^ gsn:contract(?M2, true) ^ rdf:subject(?R, ?A) ^ rdf:predicate(?R, ?O) ^ gsn:inContextOf(?O) ^ rdf:object(?R, ?M1) -> gsn:valid(?R, false)~~	~~**IF** ?A is in context of ?M1 **AND** ?M1 is a Module **AND** ?M2 contains ?A **AND** ?M2 is a Module **AND** ?M2 is a contract **AND** ?R has subject ?A **AND** ?R has predicate “in context of” **AND** ?R has object ?M1 **THEN** ?R is not valid~~	
VI.53			gsn:supportedBy(?A, ?M1) ^ gsn:Module(?M1) ^ gsn:contains(?M2, ?A) ^ gsn:Module(?M2) ^ gsn:contract(?M2, true) ^ rdf:subject(?R, ?A) ^ rdf:object(?R, ?M1) -> gsn:valid(?R, false)	**IF** ?A is supported by ?M1 **AND** ?M1 is a Module **AND** ?M2 contains ?A **AND** ?M2 is a Module **AND** ?M2 is a contract **AND** ?R has subject ?A **AND** ?R has object ?M1 **THEN** ?R is not valid	
VI.54			gsn:inContextOf(?A, ?M1) ^ gsn:Module(?M1) ^ gsn:contains(?M2, ?A) ^ gsn:Module(?M2) ^ gsn:contract(?M2, true) ^ rdf:subject(?R, ?A) ^ rdf:object(?R, ?M1) -> gsn:valid(?R, false)	**IF** ?A is in context of ?M1 **AND** ?M1 is a Module **AND** ?M2 contains ?A **AND** ?M2 is a Module **AND** ?M2 is a contract **AND** ?R has subject ?A **AND** ?R has object ?M1	

				THEN ?R is not valid	
VI.55 VI.56 VI.57 VI.58 VI.59 VI.60 VI.61	**Table 1:4-1 – Definition** A contract reference, rendered as a rectangle with a two smaller rectangles (of equal size to each other) adjoining at the top left and bottom right, presents a reference to a contract module.	34	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#contract"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <owl:propertyDisjointWith rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#undeveloped"/> <coreOrExtension>Modular Extension</coreOrExtension> <renderedAs>a rectangle with two smaller rectangles (of equal size to each other) adjoining at the top left and bottom right</renderedAs> <rdfs:label xml:lang="en">contract</rdfs:label> </owl:DatatypeProperty>	contract a DatatypeProperty contract domain Module contract range boolean contract propertyDisjointWith undeveloped contract coreOrExtension “Modular Extension” contract renderedAs “a rectangle …” contract label “contract”	Although it can be considered a subclass of a module, contract is defined as a boolean datatype property.
VI.62	Note that a contract reference points to the totality of the relationship contained in the referenced contract module, rather than just to an individual claim.	34	-	-	See comment on the statement for module references.
VI.63	A contract reference cannot be used within a contract module.	34	-	-	This rule is already covered by preventing any kind of module references (incl. where contract is true) being used in contract modules.
VI.64 VI.65 VI.66 VI.67 VI.68	**Table 1:4-1 – Definition** Public Decorator, rendered as a miniature module symbol and superimposed on a goal, solution, context, assumption or justification symbol at the top right.	34	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#public"> <rdfs:domain> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Solution"/> </owl:unionOf> </owl:Class> </rdfs:domain> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension xml:lang="en">Modular Extension</coreOrExtension> <renderedAs>miniature module symbol superimposed on an element at the top right</renderedAs> <rdfs:label xml:lang="en">public</rdfs:label> </owl:DatatypeProperty>	public a DatatypeProperty public domain (Assumption or Context or Goal or Justification or Solution) public range boolean public coreOrExtension “Modular Extension” public label “public”	
VI.69	This indicates that the element is publicly visible in one or more interfaces of the module and can be referenced as an away element.	34	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#public"> <skos:definition xml:lang="en">This indicates that the element is publicly visible in one or more interfaces of the module and can be referenced as an away element.</skos:definition> </owl:DatatypeProperty>	public definition “This indicates …”	
VI.70	The preferred location of the public decorator is within the element shape. Where this is not practical (e.g. as shown below) the exact positioning of the public decorator is not important as long as the association with the element is clear.	34	-	-	This is a visualization rule, and thus not included in this version.
VI.71 VI.72 VI.73 VI.74 VI.75	**Table 1:4-1 – Definition** To be supported by contract: This decorator, attached centrally immediately below the goal to which it relates, denotes that support for the claim presented by the attached goal is intended to be provided from an argument in another module linked by an as-yet-undisclosed contract.	35	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#toBeSupportedByContract"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension>Modular Extension</coreOrExtension> <renderedAs>attached centrally immediately below the goal to which it relates</renderedAs> <rdfs:label xml:lang="en">to be supported by contract</rdfs:label> </owl:DatatypeProperty>	toBeSupportedByContract a DatatypeProperty toBeSupportedByContract range boolean toBeSupportedByContract coreOrExtension “Modular Extension” toBeSupportedByContract renderedAs “attached centrally …” toBeSupportedByContract label “to be supported by contract”	Although this can be represented as a “supportedBy” link between a goal and a contract module that is undisclosed, the decorator is explicitly applied to a goal.
VI.76	At some later stage, the element may be updated to replace this decorator with support from a named contract module, or may be left as it is, with the necessary support defined in a higher-level argument’s architecture view.	35	-	-	Selective display in argument and architecture views seems to be for visualization purposes, and not directly ontology-relevant.
VI.77 VI.78	This decorator can only be applied to goal elements, and can be used in conjunction with the ‘to be instantiated’ annotation, but is mutually exclusive with the ‘to be developed’ annotation.	35	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#toBeSupportedByContract"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> </owl:DatatypeProperty>	toBeSupportedByContract domain Goal	
			<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#undeveloped"> <owl:propertyDisjointWith rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#toBeSupportedByContract"/> </owl:DatatypeProperty>	undeveloped propertyDisjointWith toBeSupportedByContract	“Mutually exclusive” is represented with propertyDisjointWith in OWL.
VI.79	This decorator cannot be used within a contract module.	35	gsn:toBeSupportedByContract(?A, true) ^ gsn:Module(?M) ^ gsn:contains(?M, ?A) ^ gsn:contract(?M, true) -> gsn:valid(?A, false)	**IF** ?A is to be supported by contract **AND** ?M is a Module **AND** ?M contains ?A **AND** ?M is a contract **THEN** ?A is not valid	
VI.80	**Table 1:4-2 – Definition – SupportedBy** In addition to the permitted connections defined in the core GSN definition (See Table 1:2-2), in modular GSN the	36	-	-	No additional constraints seem to be specified here. Development of away

	following ‘supported by’ connections are permitted: goal-to-away_goal, goal-to-away_solution, goal-to-module_reference, goal-to-contract_reference, strategy to away_goal.				elements in the module under view is already constrained.
VI.81	In a Contract module, the following ‘supported by’ connections are also permitted: away_goal-to-goal, away_goal-to-strategy, away_goal-to-away_goal.	36	-	-	The “away_x-to-away_x“ clause contradicts the rule that no away element should be developed in the module under view. This constraint would need clarification or a condition to ignore this when visualizing contract modules.
VI.82	**Table 1:4-2 – Definition – InContextOf** In addition to the permitted connections defined in the core GSN definition (See Table 1:2-2), in modular GSN the following ‘in context of’ connections are permitted: goal-to-away_goal, goal-to-away_context, goal-to-away_assumption, goal-to-away_justification, goal-to-module_reference, strategy-to-away_goal, strategy-to-away_context, strategy-to-away_assumption, strategy-to-away_justification and strategy-to-module_reference.	36	-	-	No additional constraints seem to be specified here. Development of away elements in the module under view is already constrained.
VI.83			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Strategy"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	Strategy inContextOf only (Assumption or Context or Justification or Module)	Added the possibility for a strategy or a goal to be in the context of a module.
VI.84			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	Goal inContextOf only (Assumption or Context or Justification or Module)	
VI.85	In a Contract module, the following ‘in context of’ connections are also permitted: away_goal-to-away_context, away_goal-to-away_assumption, away_goal-to-away_justification.	36	-	-	The “away_x-to-away_x“ clause contradicts the rule that no away element should be developed in the module under view. This constraint would need clarification or a condition to ignore this when visualizing contract modules.
VI.86	1:4.3.1 The GSN elements defined in Sections 1:2.1 and 1:4.2 above are intended to be combined to represent logical structures. The notation interpretation for core elements within modular extensions is unchanged. Away goals, away solutions and away context elements are used in place of their core counterparts with the addition that they are references to the goal, solution or context in the referenced argument module.	36	-	-	No additional constraints are specified here, so nothing needs to be represented.
VI.87	1:4.3.2 Away goals cannot be (hierarchically) decomposed and further supported by sub-elements within the current argument module; rather, decomposition needs to occur within the referenced argument module.	36	~~gsn:Module(?M1) ^ gsn:Goal(?G1) ^ gsn:contains(?M1, ?G1) ^ gsn:Module(?M2) ^ gsn:away(?G1, true) ^ gsn:supportedBy(?G1, ?G2) ^ gsn:contains(?M2, ?G2) ^ rdf:subject(?R, ?G1) ^ rdf:predicate(?R, ?O) ^ gsn:supportedBy(?O) ^ rdf:object(?R, ?G2) -> gsn:valid(?R, false)~~	~~**IF** ?M1 is a Module **AND** ?G1 is a Goal **AND** ?M1 contains ?G1 **AND** ?M2 is a Module **AND** ?G1 is away **AND** ?G1 is supported by ?G2 **AND** ?M2 contains ?G2 **AND** ?R has subject ?G1 **AND** ?R has predicate “supported by” **AND** ?R has object ?G2 **THEN** ?R is not valid~~	EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rules are defined instead.

VI.88			gsn:Module(?M1) ^ gsn:Goal(?G1) ^ gsn:contains(?M1, ?G1) ^ gsn:Module(?M2) ^ gsn:away(?G1, true) ^ gsn:supportedBy(?G1, ?G2) ^ gsn:contains(?M2, ?G2) ^ rdf:subject(?R, ?G1) ^ rdf:object(?R, ?G2) -> gsn:valid(?R, false)	**IF** ?M1 is a Module **AND** ?G1 is a Goal **AND** ?M1 contains ?G1 **AND** ?M2 is a Module **AND** ?G1 is away **AND** ?G1 is supported by ?G2 **AND** ?M2 contains ?G2 **AND** ?R has subject ?G1 **AND** ?R has object ?G2 **THEN** ?R is <u>not</u> valid	
VI.89	1:4.3.2 (…) By exception, it is valid to decompose away goals within safety case contract modules where they refer to a goal requiring support from a contract module.	36, 37	gsn:toBeSupportedByContract(?A, true) ^ gsn:Module(?M1) ^ gsn:contains(?M1, ?A) ^ gsn:supportedBy(?A, ?B) ^ gsn:Module(?M2) ^ gsn:contains(?M2, ?B) ^ gsn:away(?A, true) ^ swrlb:notEqual(?M1, ?M2) -> gsn:contract(?M2, true)	**IF** ?A is to be supported by contract **AND** ?M1 is a Module **AND** ?M1 contains ?A **AND** ?A is supported by ?B **AND** ?M2 is a Module **AND** ?M2 contains ?B **AND** ?A is away **AND** ?M1 is different from ?M2 **THEN** ?M2 is a contract	Decomposing an away goal in a contract module does not make it automatically valid, because there may be other reasons the goal is invalid. Instead, the module in which it is decomposed is marked as a contract.
VI.90	1:4.3.2 (…) Conversely, the goal requiring support, which is addressed via a contract, must not be decomposed in its host module.	37	~~gsn:toBeSupportedByContract(?A, true) ^ gsn:Module(?M) ^ gsn:contains(?M, ?A) ^ gsn:supportedBy(?A, ?B) ^ gsn:contains(?M, ?B) ^ rdf:subject(?R, ?A) ^ rdf:predicate(?R, ?O) ^ gsn:supportedBy(?O) ^ rdf:object(?R, ?B) -> gsn:valid(?R, false)~~	~~**IF** ?A is to be supported by contract **AND** ?M is a Module **AND** ?M contains ?A **AND** ?A is supported by ?B **AND** ?M contains ?B **AND** ?R has subject ?A **AND** ?R has predicate "supported by" **AND** ?R has object ?B **THEN** ?R is <u>not</u> valid~~	*EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rules are defined instead.*
VI.91			gsn:toBeSupportedByContract(?A, true) ^ gsn:Module(?M) ^ gsn:contains(?M, ?A) ^ gsn:supportedBy(?A, ?B) ^ gsn:contains(?M, ?B) ^ rdf:subject(?R, ?A) ^ rdf:object(?R, ?B) -> gsn:valid(?R, false)	**IF** ?A is to be supported by contract **AND** ?M is a Module **AND** ?M contains ?A **AND** ?A is supported by ?B **AND** ?M contains ?B **AND** ?R has subject ?A **AND** ?R has object ?B **THEN** ?R is <u>not</u> valid	
VI.92	1:4.3.3 Arguments supported by another argument module can be indicated in a number of ways. Figure 1:4-1 illustrates a firm relationship by which the parent goal is supported by a specific goal in the referenced argument module.	37	-	-	This is already covered by a regular supportedBy property.
VI.93	1:4.3.3 (…) As with core GSN, an intermediate strategy could be shown and the parent goal/strategy could be supported by one or more argument elements in addition to the away goal.	37	-	-	There is no constraint preventing this, so no change to the ontology is needed.
VI.94	1:4.3.4 By making the relationship to the away goal the author is asserting not only the inference of support for the parent goal, but also that the context in which the away goal is declared is consistent with the context and assumptions in scope for the parent goal.	37	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#consistentWith">	consistentWith a ObjectProperty	
VI.95			<rdf:type rdf:resource="http://www.w3.org/2002/07/owl#SymmetricProperty"/>	consistentWith type SymmetricProperty	
VI.96			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"/>	consistentWith domain (Assumption or Context)	
VI.97			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/>	consistentWith range (Assumption or Context)	
VI.98			<rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Assumption"/>	consistentWith coreOrExtension "Modular Extension"	
VI.99			<rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Context"/>	consistentWith label "consistent with"	
VI.100			<coreOrExtension>Modular Extension</coreOrExtension> <rdfs:label xml:lang="en">consistent with</rdfs:label> <skos:definition>By making the relationship to the away goal the author is asserting not only the inference of support for the parent goal, but also that the context in which the away goal is declared is consistent with the context and assumptions in scope for the parent goal.</skos:definition> </owl:ObjectProperty>	consistentWith definition "By making …"	
VI.101			gsn:Goal(?G1) ^ gsn:Goal(?G2) ^ gsn:supportedBy(?G1, ?G2) ^ gsn:away(?G2, true) ^ gsn:inContextOf(?G1, ?C1) ^ gsn:Context(?C2) ^ gsn:inContextOf(?G2, ?C2) -> gsn:consistentWith(?C1, ?C2)	**IF** ?G1 is a Goal **AND** ?G2 is a Goal **AND** ?G1 is supported by ?G2 **AND** ?G2 is away **AND** ?G1 is in context of ?C1 **AND** ?C2 is a Context **AND** ?G2 is in context of ?C2 **THEN** ?C1 is consistent with ?C2	
VI.102	1:4.3.6 An alternative approach is illustrated in Figure 1:4-3. The contract module instantiating the support relationship is not specified. Here, the relevant higher-level argument abstraction (e.g. architecture view) should be referred to, which will indicate where the required contract details are specified.	38			
VI.103	1:4.3.8 There may be occasions when a goal or strategy requires fuller justification than can be provided within the confines of a GSN justification element (described in Section 1:2.1 above). In such cases, an away goal can be substituted for the justification. This enables the author to invoke the argument supporting the away goal in the remote argument module as context for the goal or strategy they are currently working with.	38	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#substitutedBy">	substitutedBy a ObjectProperty	Although substitution is not mentioned elsewhere, it is defined as a separate property to allow an away goal to stand for a justification, while still being consistent with the Core GSN (i.e., inContextOf range only Context, Assumption, Justification).
VI.104			<rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Justification"/>	substitutedBy domain Justification	
VI.105			<rdfs:range>	substitutedBy range (Goal and (away value true))	
VI.106			<owl:Class>	substitutedBy coreOrExtension Modular Extension	
VI.107			<owl:intersectionOf rdf:parseType="Collection">	substitutedBy label "substituted by"	
VI.108			<rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#away"/> <owl:hasValue rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean">true</owl:hasValue> </owl:Restriction> </owl:intersectionOf> </owl:Class> </rdfs:range> <coreOrExtension>Modular Extension</coreOrExtension>	substitutedBy definition "There may …"	

			<rdfs:label xml:lang="en">substituted by</rdfs:label> <skos:definition xml:lang="en">There may be occasions when a goal or strategy requires fuller justification than can be provided within the confines of a GSN justification element. In such cases, an away goal can be substituted for the justification.</skos:definition> </owl:ObjectProperty>		
VI.109			gsn:Module(?M1) ^ gsn:Module(?M2) ^ swrlb:notEqual(?M1, ?M2) ^ gsn:inContextOf(?E, ?J) ^ gsn:Justification(?J) ^ gsn:substitutedBy(?J, ?G) ^ gsn:contains(?M1, ?J) ^ gsn:contains(?M2, ?G) -> gsn:valid(?J, false)	**IF** ?M1 is a Module **AND** ?M2 is a Module **AND** ?M1 is different from ?M2 **AND** ?E is in context of ?J **AND** ?J is a Justification **AND** ?J is substituted by ?G **AND** ?M1 contains ?J **AND** ?M2 contains ?G **THEN** ?J is <u>not</u> valid	In order to support further reasoning (i.e., abstraction of contextual relations between modules), a justification that is substituted must be in the same module as the goal; otherwise it is invalid.
VI.110	1:4.4.1 The architecture view provides an abstract view of the relationship between argument modules.	39	-	-	Already added as a viewType earlier.
VI.111	1:4.4.1 (…) The use of links in the architecture view is extended and there is a clear distinction between the use of SupportedBy and InContextOf relationships between individual elements within modules and their use in the architecture view.	39	-	-	There is no need to specify further restrictions here, because the architecture view is just an abstraction of relations between elements contained by two or more modules. More details below.
VI.112	**Table 1:4-3 – Definition – Module** Module symbols are used in the architecture view to represent an argument module.	39	-	-	Modules already have renderedAs information, and exclusion of a line is purely a visualization choice.
VI.113	The module identifier may be located internal to the symbol (as shown) or immediately below the symbol.	39	-	-	Existence of module identifiers is enforced,
VI.114	Inclusion of the module description is optional	39	<owl:AnnotationProperty rdf:about="http://schema.org/description"/>	description a AnnotationProperty	Possibility of description is added, but not enforced.
VI.115	**Table 1:4-3 – Definition – Contract** Contract symbols are used in the architecture view to represent a special type of module that defines the relationship between argument module interfaces and shows how one module supports the argument in another.	39	-	-	Identifiers are enforced. The rest is purely a visualization choice.
VI.116	Alternative contract module symbols are available to suit different styles of presentation of the architecture.	39	-	-	
VI.117	The contract identifier may be located internal to the symbol (as shown) or immediately below the symbol. Where the simple form symbol is used the identifier may be located to the side of the symbol.	39			
VI.118	Inclusion of the contract description is optional.	39	-	-	Possibility of description is there.
VI.119	**Table 1:4-3 – Definition – Module** A Module Interface Connector, rendered as a small square on the boundary of a module symbol, can optionally be added to aid clarity of the specific interface (specified by the {interface identifier}) used by the inter-module relationship.	40	-	-	Since this is purely a choice of visualization, and the building blocks already exist, nothing needs to be modified in the ontology itself.
VI.120	Where no interface is declared the default interface is assumed.	40			
VI.121	**Table 1:4-4 – Definition** The ModuleSupportedBy and ModuleInContextOf relationships are used in the architecture view represent one or more support/context relationship(s) between the elements within the modules.	40	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"/> <owl:allValuesFrom rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> </owl:Restriction> </rdfs:subClassOf> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"/> <owl:allValuesFrom rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Module"/> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	Module inContextOf only Module Module supportedBy only Module	Since this is an abstraction of relations between the elements contained in two separate modules, and nothing from earlier statements restricts using existing relations for modules, there is no need to define separate object properties.
VI.122					
VI.123			gsn:Module(?M1) ^ gsn:Module(?M2) ^ swrlb:notEqual(?M1, ?M2) ^ gsn:contains(?M1, ?E1) ^ gsn:contains(?M2, ?E2) ^ gsn:supportedBy(?E1, ?E2) -> gsn:supportedBy(?M1, ?M2)	**IF** ?M1 is a Module **AND** ?M2 is a Module **AND** ?M1 is different from ?M2 **AND** ?M1 contains ?E1 **AND** ?M2 contains ?E2 **AND** ?E1 is supported by ?E2 **THEN** ?M1 is supported by ?M2	Relations between elements in separate modules are abstracted as relations between the modules themselves, for the purpose of the architecture view.
VI.124			gsn:Module(?M1) ^ gsn:Module(?M2) ^ swrlb:notEqual(?M1, ?M2) ^ gsn:contains(?M1, ?E1) ^ gsn:contains(?M2, ?E2) ^ gsn:inContextOf(?E1, ?E2) -> gsn:inContextOf(?M1, ?M2)	**IF** ?M1 is a Module **AND** ?M2 is a Module **AND** ?M1 is different from ?M2 **AND** ?M1 contains ?E1 **AND** ?M2 contains ?E2 **AND** ?E1 is in context of ?E2 **THEN** ?M1 is in context of ?M2	
VI.125			gsn:Module(?M1) ^ gsn:Module(?M2) ^ swrlb:notEqual(?M1, ?M2) ^ gsn:contains(?M1, ?E1) ^ gsn:supportedBy(?E1, ?M2) -> gsn:supportedBy(?M1, ?M2)	**IF** ?M1 is a Module **AND** ?M2 is a Module **AND** ?M1 is different from ?M2 **AND** ?M1	Since entire modules can be added as support to a goal, the supportedBy property between a goal and a module is

				contains ?E1 **AND** ?E1 is supported by ?M2 **THEN** ?M1 is supported by ?M2	considered as a relation between the latter and the module containing the goal.
VI.126	**Table 1:4-4 – Definition** Note that the use of these symbols in the architecture View differs from that within the argument view. In the architecture view the asserted relationship is between modules and may reflect multiple individual support/context relationships across the modules' interfaces.	40	-	-	This is already covered in the rules for the previous statement (directly above).
VI.127	**Table 1:4-4 – Definition** The Composite Relationship is used where both a supported and a context relationship exists between modules.	40	-	-	Ontologies support multiple and bidirectional relations, including for visualization purposes within Protege, so "composite" and "bidirectionality" are purely stylistic choices.
VI.128	**Table 1:4-4 – Definition** The support/context relationships between modules may be bidirectional, and therefore the relationship may be shown with any of the support, context or composite arrow at either end and in any combination. A small selection of the possible combinations are illustrated (4 out of the possible 9 in addition to the 3 single ended variants)	40	-	-	
VI.129	1:4.5.1 It is useful to represent the abstracted structure of an argument in an architecture view. The process of abstraction hides the detailed structure of the argument. Goals, strategies, solutions and context are not shown in the architecture view; instead, just the modules and their relationships are depicted. The relationships are summarised such that rather than using separate links for each pairing of elements between the modules, only one link is shown.	40, 41	-	-	This is already addressed above. In future versions, it would be possible to add a quantifier showing how many particular relations are between modules.
VI.130	1:4.5.2 Figure 1:4-6 shows a SupportedBy relationship between modules. The relationship indicates that there exists one or more goal and/or strategy within module 1 which is supported by one or more goal(s) and/or evidence elements within module 2, and similarly for modules 1 and 3. There is no inference that the supporting argument provided in modules 2 and 3 necessarily supports the same goal in module 1.	41	-	-	No further restrictions are specified.
VI.131	1:4.5.2 (…) It is entirely permissible for a module both to provide support, and to be supported by another module, provided that this does not create circularity within the argument established by the composed argument modules.	41	gsn:Module(?M1) ^ gsn:Module(?M2) ^ gsn:Module(?M3) ^ swrlb:notEqual(?M1, ?M2) ^ swrlb:notEqual(?M2, ?M3) ^ swrlb:notEqual(?M3, ?M1) ^ gsn:supportedBy(?M1, ?M2) ^ gsn:supportedBy(?M2, ?M3) ^ gsn:supportedBy(?M3, ?M1) -> gsn:valid(?M1, false) ^ gsn:valid(?M2, false) ^ gsn:valid(?M3, false)	**IF** ?M1 is a Module **AND** ?M2 is a Module **AND** ?M3 is a Module **AND** ?M1 is not equal to ?M2 **AND** ?M2 is not equal to ?M3 **AND** ?M3 is not equal to ?M1 **AND** ?M1 is supported by ?M2 **AND** ?M2 is supported by ?M3 **AND** ?M3 is supported by ?M1 **THEN** ?M1 is not valid **AND** ?M2 is not valid **AND** ?M3 is not valid	This rule is specified only for the simplest case of circularity. Cyclic graphs with more nodes will require more advanced solutions.
VI.132	1:4.5.3 Contract modules can be used in the support relationship between modules to aid decoupling as shown in Figure 1:4-7. Both the full and simple forms of the contract module symbol are shown for comparison. An architecture view may use either form but should be self-consistent.	41	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#contract"> <skos:note xml:lang="en">Contract modules can be used in the support relationship between modules to aid decoupling. The de-coupling by use of a contract permits argument module construction in cases where the eventual source of support for an argument is unknown at the time of authoring or can be changed for example through re-use or planned product improvement or reconfiguration.</skos:note> </owl:DatatypeProperty>	contract note "Contract modules …"	This is added as a note for the user.
VI.133	The de-coupling by use of a contract permits argument module construction in cases where the eventual source of support for an argument is unknown at the time of authoring or can be changed for example through re-use or planned product improvement or reconfiguration.	41			
VI.134	1:4.5.5 The addition of module interface connectors can aid visualisation of which particular interface is used for a particular inter-module relationship and can be used to give greater clarity where multiple relationships exist between modules.	42	-	-	The use of module interface connectors is purely a choice of visualization aid. Given that building blocks for such visualization exist, but it is not mandatory, there is no need to specify anything additional in the ontology.
VI.135	1:4.6.1 A Module Interface identifies the published elements of the argument that a module contains.	43	-	-	
VI.136	1:4.6.2 A Module may have one or more interfaces defined, each of which should have a unique {interface identifier}.	43	-	-	
VI.137	The default interface publishes all public goals and relevant context together with all goals and references requiring external support.	43	-	-	
VI.138	Other interfaces may be published to suit specific purposes and these may be more restrictive than the default interface e.g. to allow detail to be hidden for simplicity or to control	43	-	-	

	exposure of details necessary to relate to peer modules, but unnecessary for integration into a higher level argument.				
VI.139	1:4.6.3 The Module Interface by default contains the following elements; each GSN element should be stated in full including element identifier and the complete element statement:	44	-	-	
VI.140	1:4.6.3 (…) 1. The module and interface identifier, description and configuration information.	44	-	-	The configuration information is not specified elsewhere. Therefore, the default assumption is that it should be included as part of the description.
VI.141	1:4.6.3 (…) 2. The goal(s) addressed by the module. These are all the goals declared public using the public decorator within the module. These are not necessarily the 'top' goals of a module.	44	-	-	The indicated elements of a visualization already exist as building blocks. Visualization choices should be handled outside the ontology. However, future versions of the ontology can encode visualization rules if necessary.
VI.142	1:4.6.3 (…) 3. Goals requiring support. This should include all those indicated as 'to be supported by contract' and any goals requiring support where an explicit dependency has not been declared.	44	-	-	
VI.143	1:4.6.3 (…) 4. The contextual elements (context, assumptions and justifications) relevant to the goals defined above (2 and 3).	44	-	-	
VI.144	1:4.6.3 (…) The interface needs to include all relevant contextual element in scope for that goal, which may be more than the context directly linked to the goal in the argument.	44	-	-	
VI.145	1:4.6.3 (…) Any contextual element included as in scope of a goal in the interface needs to be made public, even if not intended for reference by another argument module. Contextual elements are specific to each goal.	44, 45	gsn:public(?G, true) ^ gsn:Goal(?G) ^ gsn:inContextOf(?G, ?C) -> gsn:public(?C, true)	**IF** ?G is public **AND** ?G is a Goal **AND** ?G is in context of ?C **THEN** ?C is public	This is treated as a general rule regardless of visualization choices.
VI.146	1:4.6.3 (…) 5. Solutions and context that are available to be cited in support of goals in other argument modules. This includes all solutions and context declared public within the module.	45	-	-	The indicated elements of a visualization already exist as building blocks. Visualization choices should be handled outside the ontology. However, future versions of the ontology can encode visualization rules if necessary.
VI.147	1:4.6.3 (…) 6. Dependencies explicitly referenced within the module. This includes all away-goal, away-solution, away-contextual element references used by the argument within the module.	45	-	-	
VI.148	1:4.6.3 (…) It also includes module(s) and contract module(s) referenced from within the module, together with the goals supported by them.	45	-	-	
VI.149	1:4.6.4 Where a module interface is declared that is a subset of the default interface the sub-set should include all related contextual elements for any goals that are included.	45	-	-	
VI.150	1:4.6.5 Where a module contains other modules, the interface for the containing module can contain any element of the interface of any of the contained modules, in effect promoting the element from the contained module interface to the containing module interface.	45	-	-	
VI.151	1:4.6.5 (…) Where such a promotion occurs, this should ensure that the associated contextual elements for promoted goals are also promoted.	45	-	-	
VI.152	1:4.6.6 The identifiers for all elements within an interface, including that for any promoted element must be unique.	45	-	-	These requirements regarding uniqueness of identifiers are handled in earlier rules.
VI.153	1:4.6.6 (…) Where potential duplication occurs, e.g. where goals of the same identifier are promoted from two contained modules, this can be achieved by including the relevant module identifier, or by introducing an alias for the promoted element.	45			
VI.154	1:4.6.7 The default interface should maintain full traceability between promoted elements and their originating module, but this does not have to be carried through to an interface that is published for a specific purpose. This abstraction allows an interface to be published without revealing the internal structure of the argument it contains.	45	-	-	
VI.155	1:4.7.1 A contract may be used to relate the interfaces of modules to show how the arguments in one module support another.	45	-	-	
VI.156	1:4.7.1 (…) A contract may be described in textual form (e.g. as a table) or for more complex relationships may be described within a contract module using GSN.	45	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#contract"> <skos:note xml:lang="en">A contract may be described in textual form (e.g. as a table) or for more complex relationships may be described within a contract module using GSN.</skos:note> </owl:DatatypeProperty>	contract note "A contract …"	This statement is provided as a note for the user, because it provides details that are relevant for creators of assurance

					cases, but otherwise does not have clear rules for checking validity.
VI.157	1:4.7.2 A contract module is a special type of module that controls the relationship between argument module interfaces using arguments to define how a goal in one module is supported by one or more goals in one or more other modules.	46	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#contract"> <skos:definition xml:lang="en">A contract module is a special type of module that controls the relationship between argument module interfaces using arguments to define how a goal in one module is supported by one or more goals in one or more other modules.</skos:definition> </owl:DatatypeProperty>	contract definition “A contract …”	
VI.158	1:4.7.2 (…) It also enables argument to justify the consistency of context between those goals.		-	-	This statement does not provide enough information regarding the relation between a contract module and a justification.
VI.159	1:4.7.3 As the contract module’s purpose is to define the relationship between module interfaces it does not have a module interface of its own and cannot publish public elements.	46	gsn:public(?E, true) ^ gsn:contract(?C, true) ^ gsn:contains(?C, ?E) -> gsn:valid(?E, false)	**IF** ?E is public **AND** ?C is a contract **AND** ?C contains ?E **THEN** ?E is <u>not</u> valid	
VI.160	1:4.7.3 (…) All references from the contract module to elements in argument modules must be made using away elements (e.g. away goal, away solution, away context) and can only be made to elements that exist in module interfaces that have been made visible to it.	46	-	-	Currently not clear how to handle “interfaces that have been made visible to it”. Clarification is needed.
VI.161	1:4.7.4 A contract module can contain other modules, however the interfaces for these contained modules are only be available to the contract module in which they are contained, and/or to other modules within the same scope, i.e. they are private to the containing contract module.	46	-	-	Currently not clear how to handle “[contained modules] are private ot the containing contract module”. Clarification is needed.

Confidence Argument Extension

id	Item in GSN Community Standard	Page(s)	Item in GSN Ontology	Simplified Item in Ontology	Reason(s) for in-/exclusion
VII.1	1:5.1.1 An Assurance Claim Point (ACP) can be used in GSN to indicate that a confidence argument is associated with an assertion in a risk argument.	46	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#assuranceClaimPoint"> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension>Confidence Argument Extension</coreOrExtension> <rdfs:label xml:lang="en">assurance claim point</rdfs:label> <skos:definition xml:lang="en">An Assurance Claim Point (ACP) can be used in GSN to indicate that a confidence argument is associated with an assertion in a risk argument.</skos:definition> </owl:DatatypeProperty>	assuranceClaimPoint a DatatypeProperty	
VII.2				assuranceClaimPoint range boolean	
VII.3				assuranceClaimPoint coreOrExtension “Confidence Argument Extension”	
VII.4				assuranceClaimPoint label “assurance claim point”	
VII.5				assuranceClaimPoint definition “An Assurance …”	
VII.6			<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#argumentType"> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> <rdfs:range> <rdfs:Datatype> <owl:oneOf> <rdf:Description> <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#List"/> <rdf:first>confidence</rdf:first> <rdf:rest> <rdf:Description> <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#List"/> <rdf:first>risk</rdf:first> <rdf:rest rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#nil"/> </rdf:Description> </rdf:rest> </rdf:Description> </owl:oneOf> </rdfs:Datatype> </rdfs:range> <coreOrExtension>Confidence Argument Extension</coreOrExtension> <rdfs:label xml:lang="en">argument type</rdfs:label> </owl:DatatypeProperty>	argumentType a DatatypeProperty	
VII.7				argumentType range string	
VII.8				argumentType range oneOf {“confidence”, “risk”}	
VII.9				argumentType coreOrExtension “Confidence Argument Extension”	
VII.10				argumentType label “argument type”	
VII.11	1:5.1.1 (…) For each ACP there should exist a corresponding confidence argument.	46	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#RelationshipWithConfidence"> <rdfs:subClassOf> <owl:Class> <owl:intersectionOf rdf:parseType="Collection"> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#associatedWith"/> <owl:someValuesFrom rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> </owl:Restriction> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#assuranceClaimPoint"/>	RelationshipWithConfidence (associatedWith some Argument) and (‘assurance claim point’ some boolean)	These OWL restrictions ensure that, for some Relationship with Confidence, there exists at least one associated argument and at least one assurance claim point value, and that only confidence arguments can be associated with it.
VII.12				RelationshipWithConfidence associatedWith only (Argument and (argumentType value “confidence”))	

			<owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> </owl:Restriction> </owl:intersectionOf> </owl:Class> </rdfs:subClassOf> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#associatedWith"/> <owl:allValuesFrom> <owl:Class> <owl:intersectionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#argumentType"/> <owl:hasValue>confidence</owl:hasValue> </owl:Restriction> </owl:intersectionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> <rdfs:subClassOf> <owl:Restriction> </owl:Class>		
VII.13 VII.14 VII.15 VII.16 VII.17	1:5.2.1 Table 1:5-1 illustrates the extensions made to GSN to facilitate the representation of ACPs. These symbols are defined for use as decorators on all core GSN relation types.	46	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#RelationshipWithConfidence"> <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#predicate"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#assuranceClaimPoint"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> </owl:Restriction> </rdfs:subClassOf> <rdfs:label xml:lang="en">Relationship with Confidence</rdfs:label> </owl:Class>	RelationshipWithConfidence a Class RelationshipWithConfidence subClassOf Relationship ~~RelationshipWithConfidence predicate only (inContextOf or supportedBy)~~ RelationshipWithConfidence assuranceClaimPoint some boolean RelationshipWithConfidence label “Relationship with Confidence”	Because ACPs are only usable on core GSN relation types, additional class and restriction are added to exclude extensions (i.e., Dialectic Extension). *EDIT 21-02-25: Punning disabled because of conflict with SWRL rules.*
VII.18			<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#assuranceClaimPoint"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#RelationshipWithConfidence"/> </owl:DatatypeProperty>	assuranceClaimPoint domain RelationshipWithConfidence	
VII.19	1:5.2.2 ACPs may also be added to any element of an argument that provides a reference to an artefact e.g. solution or context where there is a need to argue the confidence in the artefact that the element references rather than the confidence related to its relationship to the argument.	47	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#assuranceClaimPoint"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/> </owl:DatatypeProperty>	associatedWith domain ArtefactReference	
VII.20	**Table 1:5-1 – Definition** A solid square is the symbol for ACP used as a decorator for a relationship. (…) It can be applied to ‘SupportedBy’ and ‘InContextOf’ relationships.	46	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#assuranceClaimPoint"> <renderedAs>solid square</renderedAs> </owl:DatatypeProperty>	assuranceClaimPoint renderedAs “solid square”	
VII.21	**Table 1:5-2 – Definition** A solid square is the symbol for ACP used as a decorator for an element. (…) It can be applied as a decorator to elements that make reference to an artefact (e.g. solution, context).	47			
VII.22	**Table 1:5-1 – Definition** The label next to the square indicates the ACP identifier.	46	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#RelationshipWithConfidence"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://schema.org/identifier"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> </rdfs:subClassOf>	RelationshipWithConfidence identifier some string	
VII.23	**Table 1:5-2 – Definition** The label next to the square indicates the ACP identifier.	47		RelationshipWithConfidence identifier exactly 1 string	

			<rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://schema.org/identifier"/> <owl:qualifiedCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger">1</owl:qualifiedCardinality> <owl:onDataRange rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> </rdfs:subClassOf> </owl:Class>		
VII.24			gsn:assuranceClaimPoint(?A, true) ^ gsn:ArtefactReference(?A) ^ swrlx:makeOWLThing(?A, ?R) -> gsn:RelationshipWithConfidence(?R) ^ rdf:subject(?R, ?A)	**IF** ?A has an assurance claim point **AND** ?A is an Artefact Reference **AND DO** (for every ?A create ?R) **THEN** ?R is a Relationship with Confidence AND ?R has subject ?A	To ensure that Artefact References with ACPs have identifiers, we first have to enforce that they're associated with a RelationshipWithConfidence.
VII.25	**Table 1:5-2 – Definition** The ACP decorator can be combined with the 'uninstantiated' decorator.	47	-	-	There is no restriction that would not allow this.
VII.26	1:5.2.3 Each ACP should have a unique identifier, e.g. "ACP1". The ACP unique identifier should be used to indicate the corresponding argument.	47	schema:identifier(?A, ?N) ^ schema:identifier(?B, ?M) ^ swrlb:notEqual(?A, ?B) ^ swrlb:equal(?N, ?M) -> gsn:valid(?A, false) ^ gsn:valid(?B, false)	**IF** ?A has an identifier ?N **AND** ?B has an identifier ?M **AND** ?A is <u>not</u> equal to ?B **AND** ?N is equal to ?M **THEN** ?A is <u>not</u> valid **AND** ?B is <u>not</u> valid	
VII.27	1:5.2.3 (…) The corresponding argument could be located in a paragraph of accompanying text, a goal in the local argument, or a goal in a separate module.	47	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#RelationshipWithConfidence"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#associatedWith"/> <owl:allValuesFrom rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> </owl:Restriction> </rdfs:subClassOf>	RelationshipWithConfidence associatedWith only Argument	Because a separate confidence argument is expected, the *associatedWith* link between *RelationshipWithConfidence* and *Argument* is expected.
VII.28			<skos:note xml:lang="en">The corresponding argument could be located in a paragraph of accompanying text, a goal in the local argument, or a goal in a separate module.</skos:note> </owl:Class>	RelationshipWithConfidence note "The corresponding …"	
VII.29			<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#statement"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> </owl:DatatypeProperty>	statement domain Argument	
VII.30			<owl:Axiom> <owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#statement"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2000/01/rdf-schema#domain"/> <owl:annotatedTarget rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> <coreOrExtension>Confidence Argument Extension</coreOrExtension>	<statement domain Argument> coreOrExtension "Confidence Argument Extension"	
VII.31			<skos:definition xml:lang="en">The corresponding argument could be located in a paragraph of accompanying text, a goal in the local argument, or a goal in a separate module.</skos:definition> </owl:Axiom>	<statement domain Argument> definition "The corresponding …"	
VII.32			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"> <rdfs:subClassOf rdf:nodeID="genid35"/> </owl:Class>	Argument (contains some Goal) or (statement some string)	
VII.33			<owl:Class rdf:nodeID="genid35"> <owl:unionOf rdf:parseType="Collection"> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#contains"/> <owl:someValuesFrom rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> </owl:Restriction> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#statement"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#string"/> </owl:Restriction> </owl:unionOf> </owl:Class>		
VII.34			<owl:Axiom> <owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2000/01/rdf-schema#subClassOf"/> <owl:annotatedTarget rdf:nodeID="genid35"/> <coreOrExtension>Confidence Argument Extension</coreOrExtension>	<Argument (contains some Goal) or (statement some string)> coreOrExtension "Confidence Argument Extension"	
VII.35			<skos:definition xml:lang="en">The corresponding argument could be located in a paragraph of accompanying text, a goal in the local argument, or a goal in a separate module.</skos:definition> </owl:Axiom>	<Argument (contains some Goal) or (statement some string)> definition "The corresponding …"	
VII.36	1:5.2.3 (…) Where the corresponding argument is located in a separate module, the module identifier should be shown alongside the ACP identifier delimited with square brackets e.g. ACP1[Confidence].	47	~~gsn:assuranceClaimPoint(?R, true) ^ gsn:contains(?M1, ?R) ^ gsn:associatedWith(?R, ?A) ^ gsn:contains(?M2, ?A) ^ swrlb:notEqual(?M1, ?M2) ^ schema:identifier(?R, ?ID1) ^ schema:identifier(?M2, ?ID2) ^ schema:identifier(?A, ?ID3) -> swrlb:stringConcat(?ID3, ?ID1, "[", ?ID2, "]")~~	~~**IF** ?R has an assurance claim point **AND** ?M1 contains ?R **AND** ?R is associated with ?A **AND** ?M2 contains ?A **AND** ?M1 is not equal to ?M2 **AND** ?R has identifier ?ID1 **AND** ?M2 has identifier ?ID2 **AND** ?A has identifier ?ID3 **THEN** ?ID3 is a concatenation of (?ID1[?ID2])~~	The indicated SWRL constraint is inactive, because it is a visualization-relevant rule. If future ontology versions address visualizations, they can include this constraint.

VII.37			-	-	
VII.38 VII.39	1:5.3.1 The presence of an ACP indicates that a separate confidence argument documenting the reasons for having confidence in the relationship or referenced artefact is provided. The nature of confidence arguments is discussed in detail in [8] (Risk, Confidence and Compliance Arguments).	47, 48	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#RelationshipWithConfidence"> <rdfs:subClassOf> <owl:Class> <owl:intersectionOf rdf:parseType="Collection"> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#associatedWith"/> <owl:someValuesFrom rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> </owl:Restriction> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#assuranceClaimPoint"/> <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> </owl:Restriction> </owl:intersectionOf> </owl:Class> </rdfs:subClassOf> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#associatedWith"/> <owl:allValuesFrom> <owl:Class> <owl:intersectionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Argument"/> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#argumentType"/> <owl:hasValue>confidence</owl:hasValue> </owl:Restriction> </owl:intersectionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	RelationshipWithConfidence ('assurance claim point' some xsd:boolean) and ('associated with' some Argument) RelationshipWithConfidence 'associated with' only (Argument and ('argument type' value "confidence"))	Added constraint to confirm that a RelationshipWithConfidence must have ACP and an associated argument, and that the associated argument should be a confidence argument. The referenced source is not a direct part of the standard, and is therefore treated as non-normative for this ontology version. Future versions can include assertions from the source.
VII.40	1:5.3.1 (…) The separate confidence argument may be documented in the current argument module, or may be contained in a separate confidence argument module, in which case the ACP identifier is extended to include the {module identifier}.	48	-	-	See point under 1:5.2.3 regarding visualization-relevant rules.
VII.41	1:5.3.2 The {ACP identifier} may be a reference to a goal, a section in a document, or other form of unique reference that can be followed by the reader of the argument.	48	-	-	This was addressed earlier using a *RelationshipWithConfidence* class, and an *associatedWith* relation with an *Argument* class.
VII.42 VII.43	1:5.3.3 In Figure 1:5-1, ACP1 is associated with the inferential relationship between G1 and its supporting goals, G2 and G3, via strategy S1. This relationship is indivisible, such that the confidence argument relates to the entirety of support for G1.	48	~~gsn:Goal(?G1) ^ gsn:Strategy(?S) ^ gsn:supportedBy(?G1, ?S) ^ gsn:Goal(?G2) ^ gsn:supportedBy(?S, ?G2) ^ gsn:RelationshipWithConfidence(?R1) ^ rdf:subject(?R1, ?G1) ^ rdf:predicate(?R1, ?O) ^ gsn:supportedBy(?O) ^ rdf:object(?R1, ?S) ^ gsn:associatedWith(?R1, ?A) ^ swrlx:makeOWLThing(?G2, ?R2) -> gsn:RelationshipWithConfidence(?R2) ^ rdf:subject(?R2, ?S) ^ rdf:predicate(?R2, ?O) ^ rdf:object(?R2, ?G2) ^ gsn:associatedWith(?R2, ?A) ^ gsn:assuranceClaimPoint(?R2, true)~~ gsn:Goal(?G1) ^ gsn:Strategy(?S) ^ gsn:supportedBy(?G1, ?S) ^ gsn:Goal(?G2) ^ gsn:supportedBy(?S, ?G2) ^ gsn:RelationshipWithConfidence(?R1) ^ rdf:subject(?R1, ?G1) ^ rdf:object(?R1, ?S) ^ gsn:associatedWith(?R1, ?A) ^ swrlx:makeOWLThing(?G2, ?R2) -> gsn:RelationshipWithConfidence(?R2) ^ rdf:subject(?R2, ?S) ^ rdf:object(?R2, ?G2) ^ gsn:associatedWith(?R2, ?A) ^ gsn:assuranceClaimPoint(?R2, true)	~~**IF** ?G1 is a Goal **AND** ?S is a Strategy **AND** ?G1 is supported by ?S **AND** ?G2 is a Goal **AND** ?S is supported by ?G2 **AND** ?R1 is a Relationship With Confidence **AND** ?R1 has subject ?G1 **AND** ?R1 has predicate "supported by" **AND** ?R1 has object ?S **AND** ?R1 is associated with ?A **AND DO** (for all ?G2 make ?R2) **THEN** ?R2 is a Relationship With Confidence **AND** ?R2 has subject ?S **AND** ?R2 has predicate "supported by" **AND** ?R2 has object ?G2 **AND** ?R2 is associated with ?A **AND** ?R2 has an assurance claim point~~ **IF** ?G1 is a Goal **AND** ?S is a Strategy **AND** ?G1 is supported by ?S **AND** ?G2 is a Goal **AND** ?S is supported by ?G2 **AND** ?R1 is a Relationship With Confidence **AND** ?R1 has subject ?G1 **AND** ?R1 has object ?S **AND** ?R1 is associated with ?A **AND DO** (for all ?G2 make ?R2) **THEN** ?R2 is a Relationship With Confidence **AND** ?R2 has subject ?S **AND** ?R2 has object ?G2 **AND** ?R2 is associated with ?A **AND** ?R2 has an assurance claim point	If for a given Goal, an ACP (and relevant things) exists for a relation with one Strategy, then all Goals supporting that Strategy should have the same. *EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rules are defined instead.*
VII.44		48	~~gsn:Strategy(?S) ^ gsn:Goal(?G1) ^ gsn:supportedBy(?S, ?G1) ^ gsn:Goal(?G2) ^ gsn:supportedBy(?G2, ?S) ^ gsn:RelationshipWithConfidence(?R1) ^ rdf:subject(?R1, ?S) ^ rdf:predicate(?R1, ?O) ^ gsn:supportedBy(?O) ^ rdf:object(?R1, ?G1) ^~~	~~**IF** ?S is a Strategy **AND** ?G1 is a Goal **AND** ?S is supported by ?G1 **AND**~~	If for a given Strategy, an ACP (and relevant things) exists for a relation

	1:5.3.3 (...) The placement of an ACP on an individual 'SupportedBy' relationship below the strategy is ambiguous and should be avoided.		~~gsn:associatedWith(?R1, ?A) ^ swrlx:makeOWLThing(?G2, ?R2) -> gsn:RelationshipWithConfidence(?R2) ^ rdf:subject(?R2, ?G2) ^ rdf:predicate(?R2, ?O) ^ rdf:object(?R2, ?S) ^ gsn:associatedWith(?R2, ?A) ^ gsn:assuranceClaimPoint(?R2, true)~~	~~?G2 is a Goal **AND** ?G2 is supported by ?S **AND** ?R1 is a Relationship With Confidence **AND** ?R1 has subject ?S **AND** ?R1 has predicate "supported by" **AND** ?R1 has object ?G1 **AND** ?R1 is associated with ?A **AND DO** (for all ?G2 make ?R2) **THEN** ?R2 is a Relationship With Confidence **AND** ?R2 has subject ?G2 **AND** ?R2 has predicate "supported by" **AND** ?R2 has object ?S **AND** ?R2 is associated with ?A **AND** ?R2 has an assurance claim point~~	with one or more Goals, then the Goal supported by the Strategy should have the same. *EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rules are defined instead.*
VII.45			gsn:Strategy(?S) ^ gsn:Goal(?G1) ^ gsn:supportedBy(?S, ?G1) ^ gsn:Goal(?G2) ^ gsn:supportedBy(?G2, ?S) ^ gsn:RelationshipWithConfidence(?R1) ^ rdf:subject(?R1, ?S) ^ rdf:object(?R1, ?G1) ^ gsn:associatedWith(?R1, ?A) ^ swrlx:makeOWLThing(?G2, ?R2) -> gsn:RelationshipWithConfidence(?R2) ^ rdf:subject(?R2, ?G2) ^ rdf:object(?R2, ?S) ^ gsn:associatedWith(?R2, ?A) ^ gsn:assuranceClaimPoint(?R2, true)	**IF** ?S is a Strategy **AND** ?G1 is a Goal **AND** ?S is supported by ?G1 **AND** ?G2 is a Goal **AND** ?G2 is supported by ?S **AND** ?R1 is a Relationship With Confidence **AND** ?R1 has subject ?S **AND** ?R1 has object ?G1 **AND** ?R1 is associated with ?A **AND DO** (for all ?G2 make ?R2) **THEN** ?R2 is a Relationship With Confidence **AND** ?R2 has subject ?G2 **AND** ?R2 has object ?S **AND** ?R2 is associated with ?A **AND** ?R2 has an assurance claim point	
VII.46	1:5.3.4 An ACP can be placed on the evidential relationship indicated by the 'SupportedBy' relationship between a goal and supporting evidence as illustrated in Figure 1:5-2.	48	-	-	This is already allowed in the Relationship class, and there are no restrictions preventing this.
VII.47	1:5.3.5 Where a single goal is supported by more than one item of evidence, the ACP applies across all 'SupportedBy' relationships in support of the goal and may be illustrated as shown in Figure 1:5-3.	49	~~gsn:Goal(?G) ^ gsn:Solution(?S1) ^ gsn:supportedBy(?G, ?S1) ^ gsn:Solution(?S2) ^ swrlb:notEqual(?S1, ?S2) ^ gsn:supportedBy(?G, ?S2) ^ gsn:RelationshipWithConfidence(?R1) ^ rdf:subject(?R1, ?G) ^ rdf:predicate(?R1, gsn:supportedBy) ^ rdf:object(?R1, ?S1) ^ gsn:associatedWith(?R1, ?A) ^ swrlx:makeOWLThing(?S2, ?R2) -> gsn:RelationshipWithConfidence(?R2) ^ rdf:subject(?R2, ?G) ^ rdf:predicate(?R2, gsn:supportedBy) ^ rdf:object(?R2, ?S2) ^ gsn:associatedWith(?R2, ?A) ^ gsn:assuranceClaimPoint(?R2, true)~~	~~**IF** ?G is a Goal **AND** ?S1 is a Solution **AND** ?G is supported by ?S1 **AND** ?S2 is a Solution **AND** ?S1 not equal to ?S2 **AND** ?G is supported by ?S2 **AND** ?R1 is a Relationship With Confidence **AND** ?R1 has subject ?G **AND** ?R1 has predicate supportedBy **AND** ?R1 has object ?S1 **AND** ?R1 is associated with ?A **AND DO** (for all ?S2 make ?R2) **THEN** ?R2 is a Relationship With Confidence **AND** ?R2 has subject ?G **AND** ?R2 has predicate "supported by" **AND** ?R2 has object ?S2 **AND** ?R2 is associated with ?A **AND** ?R2 has an assurance claim point~~	If for a given Goal, an ACP (and relevant things) exists for a relation with one Solution, then all other Solutions supporting that Goal should have the same. *EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rules are defined instead.*
VII.48			gsn:Goal(?G) ^ gsn:Solution(?S1) ^ gsn:supportedBy(?G, ?S1) ^ gsn:Solution(?S2) ^ swrlb:notEqual(?S1, ?S2) ^ gsn:supportedBy(?G, ?S2) ^ gsn:RelationshipWithConfidence(?R1) ^ rdf:subject(?R1, ?G) ^ rdf:object(?R1, ?S1) ^ gsn:associatedWith(?R1, ?A) ^ swrlx:makeOWLThing(?S2, ?R2) -> gsn:RelationshipWithConfidence(?R2) ^ rdf:subject(?R2, ?G) ^ rdf:object(?R2, ?S2) ^ gsn:associatedWith(?R2, ?A) ^ gsn:assuranceClaimPoint(?R2, true)	**IF** ?G is a Goal **AND** ?S1 is a Solution **AND** ?G is supported by ?S1 **AND** ?S2 is a Solution **AND** ?S1 not equal to ?S2 **AND** ?G is supported by ?S2 **AND** ?R1 is a Relationship With Confidence **AND** ?R1 has subject ?G **AND** ?R1 has object ?S1 **AND** ?R1 is associated with ?A **AND DO** (for all ?S2 make ?R2) **THEN** ?R2 is a Relationship With Confidence **AND** ?R2 has subject ?G **AND** ?R2 has object ?S2 **AND** ?R2 is associated with ?A **AND** ?R2 has an assurance claim point	
VII.49	1:5.3.5 (...) This representation may also be applied where a goal is supported by multiple goals without a strategy being explicitly represented.	49	~~gsn:Goal(?G) ^ gsn:Goal(?S1) ^ gsn:supportedBy(?G, ?S1) ^ gsn:Goal(?S2) ^ swrlb:notEqual(?S1, ?S2) ^ gsn:supportedBy(?G, ?S2) ^ gsn:RelationshipWithConfidence(?R1) ^ rdf:subject(?R1, ?G) ^ rdf:predicate(?R1, gsn:supportedBy) ^ rdf:object(?R1, ?S1) ^ gsn:associatedWith(?R1, ?A) ^ swrlx:makeOWLThing(?S2, ?R2) -> gsn:RelationshipWithConfidence(?R2) ^ rdf:subject(?R2, ?G) ^ rdf:predicate(?R2, gsn:supportedBy) ^ rdf:object(?R2, ?S2) ^ gsn:associatedWith(?R2, ?A) ^ gsn:assuranceClaimPoint(?R2, true)~~	~~**IF** ?G is a Goal **AND** ?S1 is a Goal **AND** ?G is supported by ?S1 **AND** ?S2 is a Goal **AND** ?S1 is not equal to ?S2 **AND** ?G is supported by ?S2 **AND** ?R1 is a Relationship With Confidence **AND** ?R1 has subject ?G **AND** ?R1 has predicate "supported by" **AND** ?R1 has object ?S1 **AND** ?R1 is associated with ?A **AND DO** (for all ?S2 make ?R2) **THEN** ?R2 is a Relationship With Confidence **AND** ?R2 has subject ?G~~	If for a given Goal, an ACP (and relevant things) exists for a relation with one Goal, then all other Goals supporting that Goal should have the same. *EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rules are defined instead.*

				~~**AND** ?R2 has predicate "supported by" **AND** ?R2 has object ?S2 **AND** ?R2 is associated with ?A **AND** ?R2 has an assurance claim point~~	
VII.50			gsn:Goal(?G) ^ gsn:Goal(?S1) ^ gsn:supportedBy(?G, ?S1) ^ gsn:Goal(?S2) ^ swrlb:notEqual(?S1, ?S2) ^ gsn:supportedBy(?G, ?S2) ^ gsn:RelationshipWithConfidence(?R1) ^ rdf:subject(?R1, ?G) ^ rdf:object(?R1, ?S1) ^ gsn:associatedWith(?R1, ?A) ^ swrlx:makeOWLThing(?S2, ?R2) -> gsn:RelationshipWithConfidence(?R2) ^ rdf:subject(?R2, ?G) ^ rdf:object(?R2, ?S2) ^ gsn:associatedWith(?R2, ?A) ^ gsn:assuranceClaimPoint(?R2, true)	**IF** ?G is a Goal **AND** ?S1 is a Goal **AND** ?G is supported by ?S1 **AND** ?S2 is a Goal **AND** ?S1 is not equal to ?S2 **AND** ?G is supported by ?S2 **AND** ?R1 is a Relationship With Confidence **AND** ?R1 has subject ?G **AND** ?R1 has object ?S1 **AND** ?R1 is associated with ?A **AND DO** (for all ?S2 make ?R2) **THEN** ?R2 is a Relationship With Confidence **AND** ?R2 has subject ?G **AND** ?R2 has object ?S2 **AND** ?R2 is associated with ?A **AND** ?R2 has an assurance claim point	
VII.51	1:5.3.6 An ACP may also be associated with an 'InContextOf' relationship as illustrated in Figure 1:5-4. This enables a confidence argument to support the contextual relationship.	49	-	-	This is already covered above.

Dialectic Extension

id	Item in GSN Community Standard	Page(s)	Item in GSN Ontology	Simplified Item in Ontology	Reason(s) for in-/exclusion
VIII.1	1:6.1.1 A Dialectic process in its simplest form is the investigation of truth. Applied to Assurance Cases, dialectics add strength to arguments by comparing options, testing truth, logically disputing and constructively criticising. The use of a dialectic process provides a framework for creating, challenging and questioning Assurance Cases through the discovery and identification of doubt, which can be depicted and the residual doubt exposed.	50	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inDoubt"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#AssuranceCase"/> </owl:DatatypeProperty>	inDoubt domain AssuranceCase	
VIII.2	1:6.2.2 GSN defines dialectic uses of the following core elements: • Goal; • Solution	50	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#challenges"> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdfs:domain rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Solution"/> <coreOrExtension>Dialectic Extension</coreOrExtension> <rdfs:label xml:lang="en">challenges</rdfs:label> </owl:ObjectProperty>	challenges domain (Goal or Solution)	
VIII.3	1:6.2.3 An additional dialectic specific relationship is provided: • Challenges	50		challenges a ObjectProperty	
VIII.4				challenges coreOrExtension "Dialectic Extension"	
VIII.5				challenges label "challenges"	
VIII.6	1:6.2.4 GSN defines a status that may be assigned to elements and relationships: • Defeated	50	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#defeated"> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension>Dialectic Extension</coreOrExtension> <rdfs:label xml:lang="en">defeated</rdfs:label> </owl:DatatypeProperty>	defeated a DataProperty	
VIII.7				defeated range boolean	
VIII.8				defeated coreOrExtension "Dialectic Extension"	
VIII.9				defeated label "defeated"	
VIII.10	1:6.2.6 The definitions below apply to all the other 'forms' of goals and solutions defined within the GSN Extension Tables throughout the standard for the normative definition i.e. Instantiable (represented within curly brackets), instantiated, undeveloped, public/private, away, as applicable.	50	-	-	There are no restrictions on elements with true property values for the indicated properties, so this should apply by default.
VIII.11	**Table 1:6-1 – Definition** A goal, (core element) can be used in a dialectic context to assert a challenge to part of the argument.	51	<owl:Axiom> <owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#challenges"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2000/01/rdf-schema#domain"/> <owl:annotatedTarget rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <skos:definition xml:lang="en">A goal, (core element) can be used in a dialectic context to assert a challenge to part of the argument.</skos:definition> </owl:Axiom>	<challenges domain Goal> definition "A goal …"	
VIII.12	**Table 1:6-1 – Definition** A solution, (core element) can be used to present a reference to an evidence item that asserts a challenge to part of the argument.	51	<owl:Axiom> <owl:annotatedSource rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#challenges"/> <owl:annotatedProperty rdf:resource="http://www.w3.org/2000/01/rdf-schema#domain"/> <owl:annotatedTarget rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Solution"/> <rdfs:label xml:lang="en">A solution, (core element) can be used to present a reference to an evidence item that asserts a challenge to part of the argument.</rdfs:label> </owl:Axiom>	<challenges domain Solution> definition "A solution …"	
VIII.13	**Table 1:6-1 – Definition** Defeated Element decorator symbol, rendered as a cross ('X') superimposed on a GSN element. This indicates that the element is defeated	51	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#defeated"> <rdfs:domain> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/>	defeated domain (Artefact Reference or Claim)	
VIII.14				defeated range boolean	
VIII.15	**Table 1:6-1 – Definition**	51		defeated renderedAs "cross ('X') …"	

VIII.16	The Defeated decorator can be applied to any of the GSN elements.		<rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> </owl:unionOf> </owl:Class> </rdfs:domain> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <renderedAs>cross (‘X’) superimposed on a GSN element or relationship</renderedAs> <skos:definition xml:lang="en">This decorator symbol indicates that the element is defeated. The Defeated decorator can be applied to any of the GSN elements.</skos:definition> </owl:DatatypeProperty>	defeated definition “This decorator …”	
VIII.17	1:6.2.7 Table 1:6-2 provides the definition and rendering of relationships for use in the dialectic extension. This declares a relationship between a source element (the entity responsible for making the challenge) and a target element. The arrow points to the target. An additional dialectic decorator is also provided.	51	-	-	Direction of the arrow is determined automatically.
VIII.18	**Table 1:6-2 – Definition** Challenges, rendered as a dashed line with an open arrowhead, allows a Challenge to any GSN entity to be documented.	51	<rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#challenges"> <renderedAs>dashed line with an open arrowhead</renderedAs> <skos:definition xml:lang="en">Allows a Challenge to any GSN entity to be documented.</skos:definition> </rdf:Description>	challenges renderedAs “dashed line …”	
VIII.19				challenges definition “Allows a …”	
VIII.20	**Table 1:6-2 – Definition** Permitted connections are: goal-to-any element, solution-to-any element, goal-to-any relationship, solution-to-any relationship,	51	<owl:ObjectProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#challenges"> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> <rdfs:range rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> </owl:ObjectProperty>	challenges range Claim	GSN elements are either Claims or Artefact References.
VIII.21				challenges range ArtefactReference	
VIII.22				challenges range Relationship	
VIII.23			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#challenges"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	Goal challenges only (ArtefactReference or Claim or Relationship)	Restrictions are added to ensure only permitted connections are added. This can be simplified in the next version by putting this restriction on the Defeater class, and then requiring first that all Defeaters (i.e., Goals or Solutions that challenge other elements) are categorized accordingly.
VIII.24			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Solution"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#challenges"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> </owl:Class>	Solution challenges only (ArtefactReference or Claim or Relationship)	
VIII.25	**Table 1:6-2 – Definition** Defeated Relationship decorator symbol, rendered as a cross (‘X’) superimposed on a GSN relationship. This indicates that the relationship is defeated	52	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#defeated"> <rdfs:domain> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> </owl:unionOf> </owl:Class> </rdfs:domain> </owl:DatatypeProperty>	defeated domain (Artefact Reference or Claim or Relationship)	Adding annotations to more advanced domain and range expressions leads to an error, so these statements are added as description.
VIII.26			<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#defeated"> <schema:description xml:lang="en">Defeated Relationship indicates that the relationship is defeated. The Defeated decorator can be applied to any of the GSN relationships.</schema:description>	defeated description “Defeated Relationship …”	
VIII.27	**Table 1:6-2 – Definition**	52			

	The Defeated decorator can be applied to any of the GSN relationships.		</owl:DatatypeProperty>		
VIII.28	1:6.3.2 The dialectic extension can be applied to any existing goal structure that complies with the other applicable normative parts of this standard. These may be in progress or deemed to be complete. Any updates that are required to refactor the structure in order to continue the dialectic process are similarly covered by this standard.	52	-	-	
VIII.29	1:6.3.3 A dialectic challenge, can be levied against any part of a goal structure, referred to here as the target of the challenge.	52	-	-	Target is not included, because it is covered by rdf:object in a Relationship where rdf:predicate is “challenges”
VIII.30	1:6.3.4 A challenge must be levied against the appropriate aspect of the goal structure. For example, it is all too easy to place challenges against a solution (evidence) which is actually valid in its own right, when it is the inference of its use that should be challenged. In such a case, the impact of any resultant defeat on the rest of the goal structure will be unclear and may lead to an invalid goal structure.	52	<rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#challenges"> <skos:note xml:lang="en">A challenge must be levied against the appropriate aspect of the goal structure.</skos:note> </rdf:Description>	challenges note “A challenge …”	Because there are no rules that dictate how to ensure the “appropriate” element is challenged, this instruction is added as a note for the user.
VIII.31	1:6.3.5 Counter evidence (via a solution) or an evidenced counter argument (via a goal) can be used to support a challenge to any element in a goal structure e.g. goal, solution, strategy, context, assumption, justification including those that are extended by the other extensions to GSN.	52	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Defeater"> <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> </owl:Class>	Defeater subClassOf Claim	This part is interpreted as covering the Defeater (*Dialectic Extension*, as subclass of Claim) and RelationshipWithConfidence (*Confidence Argument Extension*, as subclass of Relationship), so as subclasses, they inherit their link to the challenges object property. Modules and Patterns are not GSN elements as such, and thus not included.
VIII.32			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#RelationshipWithConfidence"> <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> </owl:Class>	RelationshipWithConfidence subClassOf Relationship	
VIII.33	1:6.3.6 Figure 1:6-1 depicts a dialectic challenge to a goal that results in defeat. The dialectic challenge within this structure asserts that if the evidence referred to in Solution CSn1 is valid, this is sufficient to establish that the claim in Goal G1 in the original structure is successfully challenged. Thus, a challenge to a target element is documented by identifying the counter evidence that makes this challenge.	53	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#valid"> <rdfs:domain> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> </owl:unionOf> </owl:Class> </rdfs:domain> </owl:DatatypeProperty>	valid domain (Claim or Artefact Reference)	
VIII.34			gsn:valid(?S, true) ^ gsn:Solution(?S) ^ gsn:challenges(?S, ?E) -> gsn:defeated(?E, true)	**IF** ?S is valid **AND** ?S is a Solution **AND** ?S challenges ?E **THEN** ?E is defeated	
VIII.35	1:6.3.7 In Figure 1:6-1, the challenge made by the evidence presented in solution Sn1 is valid and the claim presented in goal G1 is defeated. The defeat is depicted by the defeated decorator, which is applied to indicate that goal G1 is no longer valid and so presents a claim left as defeated in the goal structure.	53	gsn:defeated(?E, true) -> gsn:valid(?E, false)	**IF** ?E is defeated **THEN** ?E is <u>not</u> valid	
VIII.36	1:6.3.8 Figure 1:6-2 depicts a dialectic challenge to a goal. The dialectic challenge within this structure asserts that if the claim presented in Goal CG1 is true then this is sufficient to establish that the claim in Goal G1 in the original structure is in doubt. Thus, a challenge to a target element is documented by identifying a claim that asserts a challenge.	53	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inDoubt"> <rdfs:domain> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#ArtefactReference"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#AssuranceCase"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Claim"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"/> </owl:unionOf> </owl:Class> </rdfs:domain> <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/> <coreOrExtension>Dialectic Extension</coreOrExtension> <rdfs:label xml:lang="en">in doubt</rdfs:label> <skos:definition xml:lang="en">The dialectic challenge within this structure asserts that if the claim presented in Goal CG1 is true then this is sufficient to establish that the claim in Goal G1 in the original structure is in doubt.</skos:definition> </owl:DatatypeProperty>	inDoubt a DataProperty	InDoubt is not the same as invalid, therefore it is a new datatype property.
VIII.37				inDoubt domain (Artefact Reference or Assurance Case or Claim or Relationship)	
VIII.38				inDoubt coreOrExtension “Dialectic Extension”	
VIII.39				inDoubt label “in doubt”	
VIII.40				inDoubt definition “The dialectic …”	
VIII.41			gsn:Goal(?G) ^ gsn:challenges(?G, ?E) ^ gsn:true(?G, true) -> gsn:inDoubt(?E, true)	**IF** ?G is a Goal **AND** ?G challenges ?E **AND** ?G is true **THEN** ?E is in doubt	
VIII.42	1:6.3.8 (…) The challenge is complete only once an argument to support the assertion is developed and evidenced and so a counter argument is formed.	53	gsn:Goal(?G) ^ gsn:challenges(?G, ?E) ^ gsn:inDoubt(?E, true) ^ gsn:undeveloped(?G, false) -> gsn:defeated(?E, true)	**IF** ?G is a Goal **AND** ?G challenges ?E **AND** ?E is in doubt **AND** ?G is <u>not</u> undeveloped **THEN** ?E is defeated	

VIII.43 VIII.44	1:6.3.9 Counter evidence (via a solution) or an evidenced counter argument (via a goal) can be used to challenge any relationship in a goal structure i.e. SupportedBy, InContextOf, Challenges, including those that are extended by the other extensions to GSN.	53	~~<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Relationship"> <rdfs:subClassOf> <owl:Restriction> <owl:onProperty rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#predicate"/> <owl:allValuesFrom> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#supportedBy"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#inContextOf"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#challenges"/> </owl:unionOf> </owl:Class> </owl:allValuesFrom> </owl:Restriction> </rdfs:subClassOf> </owl:Class>~~	~~Relationship predicate only (supportedBy or inContextOf or challenges)~~	Although extensions are meant to be included, it is not clear how a challenge would interact with the Modular Extension (e.g., substituted by), Argument Pattern Extension (e.g., instantiation of) or Confidence Argument Extension (e.g., associated with). *EDIT 21-02-25: Punning disabled because of conflict with SWRL rules.*
			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#challenges"/>	challenges a Class	Object properties are intentionally asserted as classes (“punning”), to allow connection to Relationships via rdf:predicate.
VIII.45 VIII.46			~~gsn:challenges(?A, ?B) ^ swrlx:makeOWLThing(?B, ?R) -> gsn:Relationship(?R) ^ rdf:subject(?R, ?A) ^ rdf:predicate(?R, gsn:challenges) ^ rdf:object(?R, ?B)~~	~~**IF** ?A challenges ?B **AND DO** (for every ?B create ?R) **THEN** ?R is a Relationship **AND** ?R has subject ?A **AND** ?R has predicate “challenges” **AND** ?R has object ?B~~	Triples containing “challenges” are automatically reified. *EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rules are defined instead.*
			gsn:challenges(?A, ?B) ^ swrlx:makeOWLThing(?B, ?R) -> gsn:Relationship(?R) ^ rdf:subject(?R, ?A) ^ rdf:object(?R, ?B)	**IF** ?A challenges ?B **AND DO** (for every ?B create ?R) **THEN** ?R is a Relationship **AND** ?R has subject ?A **AND** ?R has object ?B	
VIII.47	1:6.3.10 Figure 1:6-3 depicts a dialectic challenge to a SupportedBy relationship that results in defeat. This is documented by identifying the evidence referred to by Solution CSn1 that asserts this challenge. Thus, a successful challenge to a target relationship is developed by applying counter evidence, similarly to Section 1:6.3.6.	53, 54	-	-	This rule is already represented in 1:6.3.6.
VIII.48	1:6.3.11 In Figure 1:6-3, the evidence presented in solution CSn1 defeats the SupportedBy relationship. The defeat is depicted by the defeated decorator, which is applied to indicate that the SupportedBy relationship is no longer valid and so is presented as defeated in the goal structure.	54			
VIII.49	1:6.3.12 As the inference between a goal and its supporting goal is indivisible, it is only possible to challenge the inference relationship in its entirety. A challenge cannot be made directly to multiple SupportedBy relationships, so challenges to this inference require a strategy to be inserted.	54	~~gsn:Goal(?G1) ^ gsn:Goal(?G2) ^ gsn:Goal(?G3) ^ swrlb:notEqual(?G1, ?G2) ^ gsn:supportedBy(?G3, ?G1) ^ gsn:supportedBy(?G3, ?G1) ^ gsn:challenges(?D, ?R) ^ rdf:subject(?R, ?G3) ^ rdf:predicate(?R, gsn:supportedBy) ^ rdf:object(?R, ?G1) ^ swrlx:makeOWLThing(?G3, ?S) -> gsn:valid(?R, false) ^ gsn:valid(?D, false) ^ gsn:Strategy(?S) ^ gsn:challenges(?D, ?S) ^ gsn:supportedBy(?S, ?G1) ^ gsn:supportedBy(?S, ?G2) ^ gsn:supportedBy(?G3, ?S) ^ skos:note(?S, “Change needed!”) ^ gsn:valid(?S, false)~~	~~**IF** ?G1 is a Goal **AND** ?G2 is a Goal **AND** ?G3 is a Goal **AND** ?G1 is not equal to ?G2 **AND** ?G3 is supported by ?G1 **AND** ?G3 is supported by ?G2 **AND** ?D challenges ?R **AND** ?R has subject ?G3 **AND** ?R has predicate “supported by” **AND** ?R has object ?G1 **AND DO** (for every ?G3 create ?S) **THEN** ?R is not valid **AND** ?D is not valid **AND** ?S is a Strategy **AND** ?D challenges ?S **AND** ?S is supported by ?G1 **AND** ?S is supported by ?G2 **AND** ?G3 is supported by ?S **AND** ?S has note “Change needed!” **AND** ?S is not valid~~	Instead of just evaluating when a defeater erroneously covers only a part of a composite relationship, it is also possible to: 1. partially rectify this by adding a strategy and the relevant properties with respect to other elements, and then 2. indicate where the user should make remaining changes. *EDIT 21-02-25: Punning disabled because of conflict with SWRL rules. New rules are defined instead.*
VIII.50			gsn:Goal(?G1) ^ gsn:Goal(?G2) ^ gsn:Goal(?G3) ^ swrlb:notEqual(?G1, ?G2) ^ gsn:supportedBy(?G3, ?G1) ^ gsn:supportedBy(?G3, ?G1) ^ gsn:challenges(?D, ?R) ^ rdf:subject(?R, ?G3) ^ rdf:object(?R, ?G1) ^ swrlx:makeOWLThing (?G3, ?S) -> gsn:valid(?R, false) ^ gsn:valid(?D, false) ^ gsn:Strategy(?S) ^ gsn:challenges(?D, ?S) ^ gsn:supportedBy(?S, ?G1) ^ gsn:supportedBy(?S, ?G2) ^ gsn:supportedBy(?G3, ?S) ^ gsn:valid(?S, false)	**IF** ?G1 is a Goal **AND** ?G2 is a Goal **AND** ?G3 is a Goal **AND** ?G1 is not equal to ?G2 **AND** ?G3 is supported by ?G1 **AND** ?G3 is supported by ?G2 **AND** ?D challenges ?R **AND** ?R has subject ?G3 **AND** ?R has object ?G1 **AND DO** (for every ?G3 create ?S) **THEN** ?R is not valid **AND** ?D is not valid **AND** ?S is a Strategy **AND** ?D challenges ?S **AND** ?S is supported by ?G1 **AND** ?S is supported by ?G2 **AND** ?G3 is supported by ?S **AND** ?S is not valid	

VIII.51	1:6.3.12 (...) Figure 1:6-4 depicts a dialectic challenge to a multiple SupportedBy relationship that results in defeat. If in the left-hand goal structure (a) the supporting-goals G2 and G3 are considered not sufficient and suitable to support goal G1 and a challenge to this inference is achieved by inserting strategy S1 below goal G1 and applying the challenge to the new strategy, as in the right-hand goal structure (b). The defeat is depicted by the defeated decorator, which is applied to indicate that the strategy S1 is no longer valid.	54	-	-	This rule is already represented in 1:6.3.6. There is no indication that the defeated status propagates across the upstream and downstream “supportedBy” goals.
VIII.52	1:6.3.13 Figure 1:6-5 depicts a challenge to a Challenges relationship that is documented by identifying a claim represented by goal CG2 that asserts a challenge. Thus, the original challenge can itself be challenged by forming an evidenced counter argument (similarly to Section 1:6.3.8 above). The doubt raised has yet to be resolved.	55	-	-	Given the open world assumption of OWL, and no rules which constrain this, there is no need to represent the "can be" axioms.
VIII.53	1:6.3.14 A challenge may be countered and so may itself be subject to further challenge. A countering challenge may be made to a preceding challenge by challenging: the inference of the challenge (via the Challenges relationship) as in Figure 1:6-5;	55	-	-	
VIII.54	1:6.3.14 (...) counter evidence (via the associated solution); a counter claim (via the goal);	55			
VIII.55	1:6.3.14 (...) or any part of a supporting evidenced counter argument that supports the counter claim.	55			
VIII.56	1:6.4.2 In a dialectic context, the goal statement is expressed to make a claim that asserts a challenge to a part of the argument.	56	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Defeater"> <skos:note xml:lang="en">The goal and solution statements should be clearly expressed such that the crux of the challenge is unequivocally communicated. Thus, the link between the part of the argument that is being challenged (target) and the dialectic element (source) is self-evident.</skos:note> </owl:Class>	Defeater note “The goal ...”	Because it is difficult to automatically ensure that the Defeater’s statement is relevant to the statement of its target, this note is left for the author.
VIII.57	1:6.4.3 In a dialectic context, a solution references evidence that challenges part of the argument.	56			
VIII.58	1:6.4.4 The goal and solution statements should be clearly expressed such that the crux of the challenge is unequivocally communicated. Thus, the link between the part of the argument that is being challenged (target) and the dialectic element (source) is self-evident.	56			

Part 2

id	Item in GSN Community Standard	Page(s)	Item in GSN Ontology	Simplified Item in Ontology	Reason(s) for in-/exclusion
IX.1	2:11.2.2 Dialectic can be used as a prefix: • 'Dialectic argument' - the outcome of using dialectic thinking or process	115	<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#argumentType"> <rdfs:range> <rdfs:Datatype> <owl:oneOf> <rdf:Description> <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#List"/> <rdf:first>dialectic</rdf:first> <rdf:rest> </rdf:Description> </owl:oneOf> </rdfs:Datatype> </rdfs:range> </owl:DatatypeProperty>	argumentType range (confidence or dialectic or risk)	
IX.2	Footnote 6: A dialectic element can sometimes be referred to as a 'defeater', though it does not necessarily result in defeat.	115	<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Defeater"> <rdfs:label xml:lang="en">Defeater</rdfs:label> </owl:Class>		"Defeater" is added as a concept, although it is only used as an additional class.
IX.3	• 'Dialectic element' - the source of challenge being applied[6]	115	<skos:altLabel xml:lang="en">Dialectic Element</skos:altLabel>		"Dialectic Element" is an alternative label, since "defeater" is a clearer term with similar semantics in other standards (e.g., CAE). If needed, "dialectic" can be redefined as a data property.
IX.4	2:11.3.1.2 A defeater (goal or solution) can challenge any element in a goal structure, e.g. goal, solution, strategy, context, assumption, justification.	117	<skos:definition>The source of challenge being applied; can be directed at any part of an argument. A defeater (goal or solution) can challenge any element in a goal structure, e.g. goal, solution, strategy, context, assumption, justification.</skos:definition>		
IX.5			<owl:Class rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Defeater"> <owl:equivalentClass> <owl:Class> <owl:intersectionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Defeater"/> <owl:Class> <owl:unionOf rdf:parseType="Collection"> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Goal"/> <rdf:Description rdf:about="http://www.semanticweb.org/momcilovic/ontologies/2024/1/gsn#Solution"/> </owl:unionOf> </owl:Class> </owl:intersectionOf> </owl:Class> </owl:equivalentClass> </owl:Class>	Defeater EquivalentTo (Defeater and (Goal or Solution))	Defeater is an "emergent class" of a goal or solution that challenges another element, and elements should be classified as such when they fulfill these conditions.
IX.6			gsn:challenges(?A,?B) -> gsn:Defeater(?A)	**IF** ?A challenges ?B **THEN** ?A is a Defeater	
IX.7			-	-	Currently, there is no solution for excluding individuals from being members of the Defeater class when they are no longer challenging another element.
IX.8			<skos:note xml:lang="en">Membership of this class is only meant to be inferred, not asserted! Please do not add individuals to this class manually (i.e., assertions).</skos:note>		This note is added to deter users from adding standalone defeaters (without referring to a goal or a solution), or interferring with the SWRL rules.

Archive

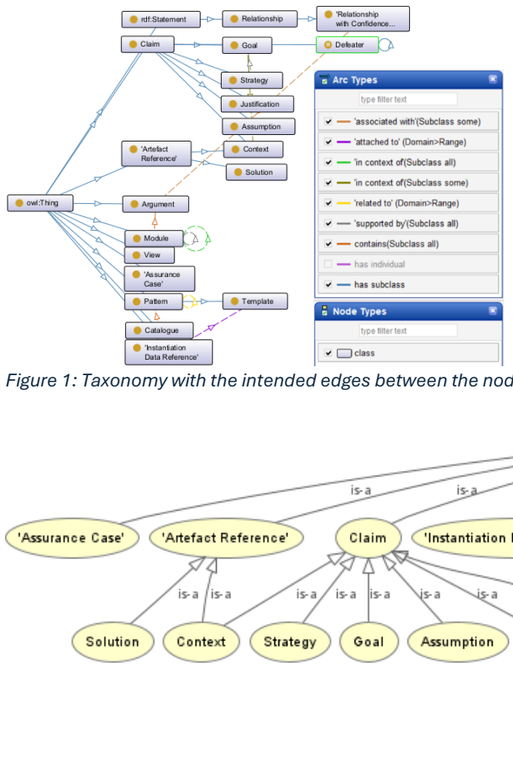


Figure 1: Taxonomy with the intended edges between the nodes.

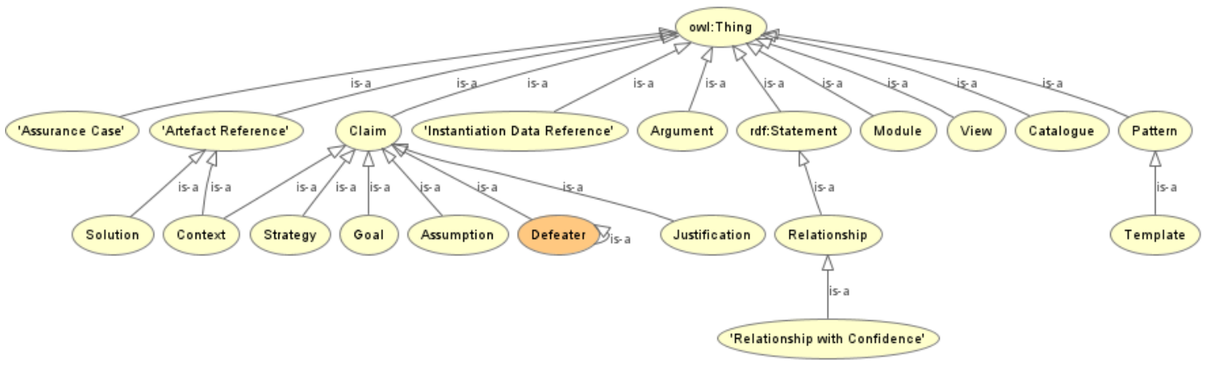


Figure 2: Taxonomy of classes (i.e., nodes).

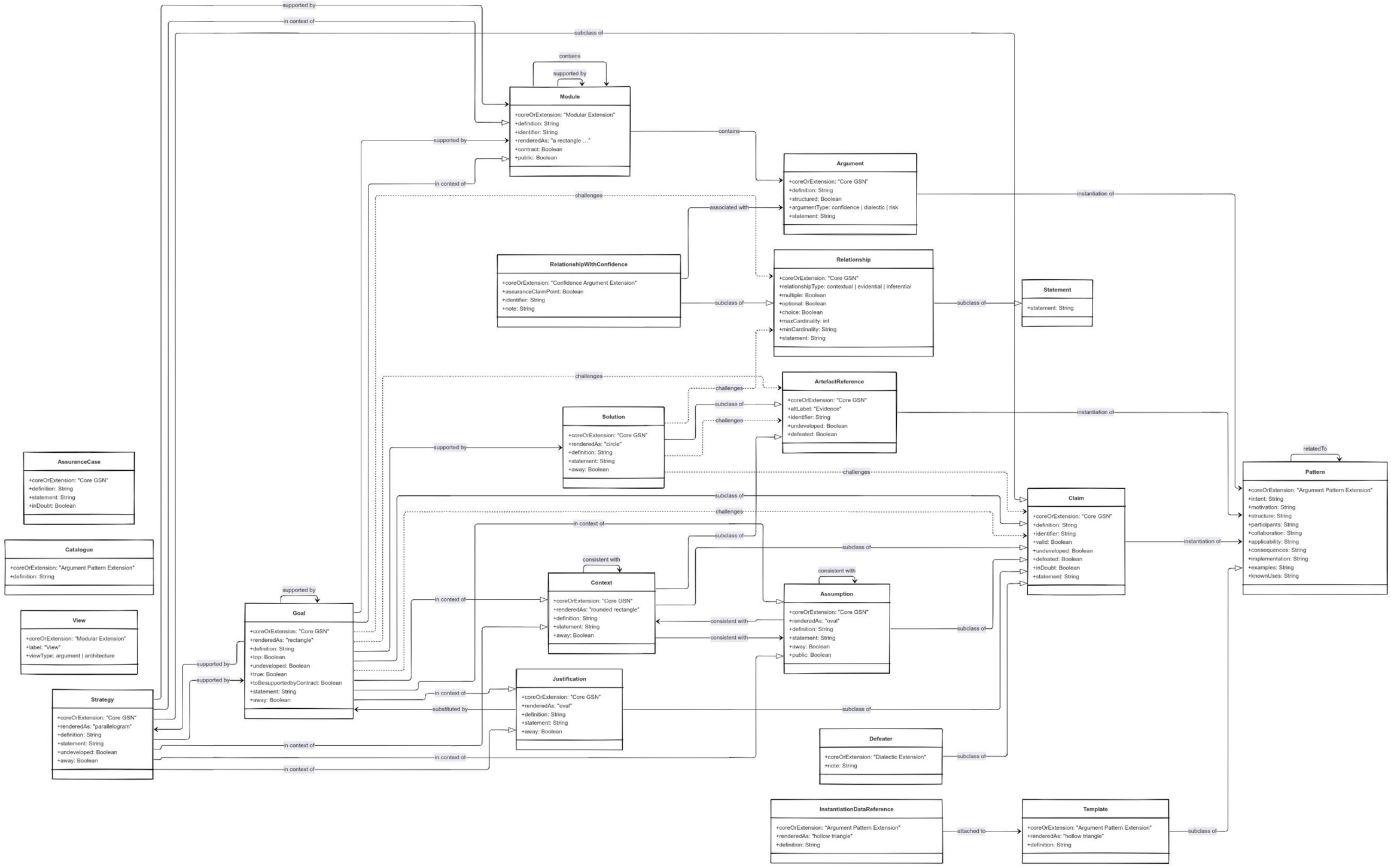


Figure 3: Class diagram of the entire ontology (rules are omitted); generated using Mermaid Charts[15].

[15] https://www.mermaidchart.com/