



Use of Evidence Based Arguments in Standard Compliance

Managing Safety Case Relations to System Models

Andrzej Wardziński Gdańsk University of Technology, Poland ARGEVIDE sp. z o.o., Gdańsk, Poland

> SCSSS 2017 23 May 2016, Stockholm







Part 1: Use of Evidence Based Arguments in Standard Compliance

- 1. Evidence based arguments
- 2. Standard structure and requirements
- 3. Demonstrating compliance and making assessment
- 4. Managing standards

Part 2: Managing Safety Case Relations to System Models

- 1. References to the system context
- 2. System model
- 3. Establishing and maintaining relations





- Argument structure based on Toulmin's argument model
 - comply with ISO 15026 and OMG SACM
- Argument premises may be supported by evidence







Available argument notations:

- graphical notations (GSN, CAE)
- tabular notation
- hierarchical textual notation
 - TCL Trust Case Language
 - developed at Gdańsk University of Technology in 2007





Prescriptive vs. goal based standards



Use of evidence based arguments (assurance cases) is already required by some goal based standards

	Prescriptive standards	Goal based standards
Requirements of a standard	specify precisely what should be demonstrated	specify goals and allow different ways how it is achieved
How to demonstrate compliance	Provide evidence the requirement is satisfied	 Define strategy how the goal is achieved Justify the strategy is effective Provide evidence the strategy is followed



Conformance case



- Argument hierarchy can represent structure of a standard
 - directly or with mapping
- Leaves of the argument represent requirements of the standard
- Users can provide evidence to demonstrate compliance
- Argument can be extended with additional information like:
 - guidance for standard users
 - assessment procedures and criteria





OF TECHNOLOGY

Project Edit View Reports Help	Project: ASPICE 3.0 (ASPICE 3.0)		Andrzej Wardziński
 Project Edit View Reports Help ASPICE 3.0 template (ACQ decomposition) ASPICE Template for selected processes Scope of assessment Acquisition Process Group (ACQ) ACQ.3: Contract Agreement ACQ.3: Contract Agreement Accult 1 - Performed process Accult 1 - Performed process Accult 2 - Performed process Acquisition Process Base Practices AcQ.3.BP1: Negotiate the composition of Acquire and Ac	Project: ASPICE 3.0 (ASPICE 3.0) Close Filter: Hidden Rationales ork products e attribute Intract/agreement Ind duties //agreement for supplier capability monitoring //agreement agreement agreement esult to tenderers	Details Fact Name: Label: Tags: Tags: Font Negotiate [OUTCO NOTE 1: requirement between p process re	Andrzej Wardziński Negotiate the contract/agreement ACQ.3.BP1 Image: Second Sec
ACQ.12: Legal and Administrative Requirements	ts 🗸	Assessment	
_			

[21-05-2017 18:17:43] Node opened Negotiate the contract/agreement





Assessment scales



- 8
- Compliance for each requirement of the standard can be evaluated separately
- Different assessment methods can be used, for example:
 - Dempster-Shafer method permits to represent uncertainty (e.g. missing information)
 - SPICE is using 0..100 scale with four levels of compliance (N-P-L-F)
 - Rating scale is using number for evaluation
 - 3-value scale (noncompliant, partially compliant, compliant)





Assessment reporting



aSPICE report visy - Microsoft Eycel

9

Assessment results can be

- represented with a color scale
- reported to MS Excel, XML, PDF

			Plik	Narzędzia g	łówne V	Vstawianie	Układ strony	Formuły	Dane	Recenzja	Widok	Deweloper	Dodatki	Foxit Rea	ider PDF	Team	
				E16	- (°	f _∞ =JE	ŻELI.BŁĄD(W	VYSZUKAJ.P	IONOWO(E	28;'NOR-ST	A'!\$C\$2:\$F\$	3000;4;FAŁS	SZ);"×")				
			A	B C	D	E	F	G	Н	1	J	К	L	М	N	0	Р
A RD - NOR-STA ×			1 2	Proc	ess /	Attrib	oute										
← → C 🔒 https://services.argevide.com/rd		¶☆ =	3														
A Project View Reports Account Help		Log out	4	PA3.2 PA3.1													
SO 27001 Assessment	Details		6	PA2.2													
Close Filter: Hidden Argumentation Strategies	Assessment		7	PA2.1													
A5: Information security policies			8	PA1.1													
6 a C A6: Organization of information security	Assessment:					_	_ R	R	>	Inte	ø	70		≅ Q	Ξ.,	z	M
🜔 🗉 🗩 A7: Human resource security	8 / 10 (80%)			Process	gree	foni		Sys Quin Ana	Sys	Sys nteg Te	Sys Te	Se O	erifi	anag	Prot	Reg	Pro
🌜 🗉 💭 A8: Asset management				Attribute/	eme	torin	ation	tem Iysis	tem sign	tem fatio	icati ist	ality	catio	urati Jerne	Jem	linge	ject
🌗 🖻 💭 A9: Access control			9	Area	a	<u>د</u>	nts	nts		n	n	œ	5	Ion ent	ent "	ent	ent
A9.1: Business requirements of access control	Exclude from assessment	ete Assessment	10		ACQ3	ACQ4	SYS1	SYS2	SYS3	SYS4	SYS5	SUP1	SUP2	SUP8	SUP9	SUP10	MAN3
● = 💭 A9.2: User access management			11														
O Page A9.2.1: User registration and de-registration	4			1			1			1	1						
A9.2.2: User access provisioning			17 · (* •) =			ISO_27001	.xlsx - Microso	ft Excel								
A9.2.3: Management of privileged access rights			larzędzia główi	ne Wstawiani	e Układ st	rony Formu	Jły Dane	Recenzja	Widok D	eweloper			0.	- ® X			
A9.2.4: Management of secret authentication information of users	Fort Size	- I max	C. IL .						Form	atowanie waru	nkowe + 🛛 🖓 🖙	Wstaw - D	· A- m				
A9.2.5: Review of user access rights	T OIR		Calibri	* 11 * <i>F</i>	A A - 7		Ogoi	ne	Form	atuj jako tabelę	- ÷	Usuń 👻 🛃	, Zĩ m				
A9.2.6: Removal or adjustment of access rights		Wklej	BIL		<u>A</u> ·] ≡ :		· · · · · · · · · · · · · · · · · · ·	% 000 56	🐝 📑 Style	komórki *		Format * 🥥	* filtruj * zazna		-	100	60
All All All Sectors and an Illantian access and the		Schowek	9	Czcionka	G	Wyrównanie	G	Liczba	6	Style	K	omórki	Edycja		25	75	
A 2.4: System and application access control			A31	• (*	f _x									*	100	71	
A to. Cryptography		A	В	C		D	E	F		G	H	- I	J K		100	13	20
Alt: Operations security		1				ISO 2	27001 As	sessmer	nt Result	s					3 _	3	3
A a A13: Communications security		2													Pro	ana Re Ch	ana
Alta: System acquisition, development and maintenance		3													625	699	62
A15: Supplier relationships	A	pply 5						0% 10%	20% 30%	40% 50%	60% 70%	80% 90%	100%				
A 16- Information excurity incident management		6							1 1								
[04-11-2015 15:11:40] Node opened Management of privileged access rights		7	_		A5.	Information se	curity policies		1 1								
					A6. Organiz	ation of inform	ation security										
		10				A7. Human res	ource security										
						A8. Asset	t management										

🕱 | 🔜 | 🗳 + 🕲 - | A + 🖏 - | =



The Compliance Process Supported with Arguments



- Define structure of a standards (conformance case template)
- Plan your compliance project (start with an empty compliance case)
- 3. Provide evidence and compliance argument
- Make assessment (self assessment, certification assessment)
- 5. Report progress and level of the compliance
- 6. Maintain compliance



Applications



The approach has been applied by commercial users for standards:

- Hospital Accreditation Standards (NCQA, Poland)
- ISO 9001 Quality management systems
- ISO 14001 Environmental Management Systems
- OHSAS 18001 Occupational Health and Safety Management
- ISO 27001 Information Security Management
- IEC 62443 Security for industrial automation and control systems
- EN/IEC 61511 Functional safety Safety instrumented systems for the process industry sector
- ISO 26262 Road vehicles Functional safety
- ISO/IEC 17065 Conformity assessment Requirements for bodies certifying products, processes and services







- 12
 - Argumentation structure is easier to comprehend that traditional documentation of standards
 - users better understand the standard requirements
 - You can create an integrated compliance environment consisting of:
 - requirements of the standard
 - guidance, best practices, evidence samples
 - compliance evidence and descriptions
 - assessments and comments
 - The approach helps to maintain consistency in conformance projects
 - Online cooperation improves communication between organizations



Standards going electronic



- Traditional document structure of standards is
 - optimal for technical publication (and will not disappear)
 - not optimal for using it and for managing
- Standards logical structure and dependencies become more and more complex
 - maturity levels, SILs, EALs, process areas, practices, etc.
- Argumentation structure is a step in the right direction to represent logical structure of a standard
 - More advanced data structures may also be useful
- It helps to manage complex standards
- XML representation makes possible exchange of compliance information between systems and organizations



Managing Safety Case Relations to System Models



Safety argument in the context



- Argument context includes...
 - System structure, elements and their properties
 - Behaviour (events, processes)
 - Risk model (hazards, causes, safety requirements)
 - Environment structure and properties
 - System life cycle activities and artefacts
- A valid safety argument needs the context to be correct and consistent





How can the context be managed?



Informal references

- Use context names in argument elements
 - Example claim: Speed sensor S17 failure rate is below 10-6
- Distinct context elements
 - GSN Standard specifies a Context element
 - A context, presents a contextual artefact. This can be a reference to contextual information, or a statement.
- Model generated argument
 - Automatic safety argument generators ensure argument consistency with system models used.
- Direct references to system model elements







For the presented fragment of an argument:

- Claim1: Hazardous situation {H} is mitigated
 - Context1: Severity: {Sev}
 - Context2: Hazard {H} description
- Argument1: Argument strategy over hazard causes
 - Justification1: Hazard is mitigated by providing control measures for all its causes
 - [1..*] Claim1.1: Cause {C} is addressed by control measures

The goal is:

- to establish references to valid elements of the risk model
- to ensure referenced elements relations hold
 (e.g. we refer to causes of the hazard specified in the parent claim)
- to maintain correctness of the references and to be informed when it is challenged (e.g. elements of the risk model are modified)



The system metamodel



- 18
 - System metamodel defines an abstract schema for system models
 - It defines entities, attributes and relations
 - UML class diagram can be used to present a metamodel
 - Example:





Reference model



- System metamodel enables establishing references to:
- elements of a given type
- elements in a specified relation with context elements
- We extend the safety argument parameters with:
- a model type
- a selector which specifies an element type or relation
- Claim1: Hazardous situation {H:HModel:Hazard} is mitigated
 - Context1: Severity: {Sev:HModel:SeverityOfHazard(H)}
 - Context2: Hazard {H} description
 - Argument1: Argument strategy over hazard causes
 - Justification1: Hazard is mitigated by providing control measures for all its causes
 -] 🔽 [1..*] Claim1.1: Cause {C:HModel:CausesOfHazard(H)} is addressed by control measures



Interfacing safety argument to system models



An intermediary named *Model interface* can:

- provide information about system metamodel classes and relations
- give lists of elements which satisfy the reference requirement
- verify if a given element or relation is up to date





Uniform model interface



21



The minimal model of a model interface which permits to establish and maintain references to system models.





Pre-development phase steps

- 1. System metamodel specification
- 2. Model interface development
- 3. Argument pattern development
- Development phase steps
 - 4. System modeling
 - 5. Assurance case development (instantiation)
 - 6. System models and assurance case maintenance (iteration of steps 4 and 5)



Relations data



23

The relations data are maintained in:

abstract reference table

Pattern element id	Reference name	Model type	Element selector
Claim1 Context2	Н	HModel (the risk model)	Hazard
Context1	Sev	HModel (the risk model)	SeverityOfHazard(H)
Claim1.1	С	HModel (the risk model)	CausesOfHazard(H)

concrete(instantiation) reference table

Argument element id	Reference name	Model name	Model element id	Element name
C1 Ctxt2	н	PCAHazardTable.xml	H1	Air in line
Ctxt1	Sev	PCAHazardTable.xml	S1	Critical
C2	С	PCAHazardTable.xml	C1	Sensor failure to detect air bubble
C3	С	PCAHazardTable.xml	C2	Safety subsystem failure to stop the pump
C4	С	PCAHazardTable.xml	C4	Pump does not stop on request



1

±

Prototype solution



Prototype solution

- Manual specification of argument pattern parameters
- Prototype instantiation tool reads / writes SACM 1.1 arguments
- The model interface implemented for XML risk model and OSATE AADL models (partially)
- C1: Hazardous situation 'Air in line' is mitigated
 - Ctxt1: Severity: 'Critical'
 - Ctxt2: Hazard 'Air in line' description
 - A1: Argument strategy over hazard causes
 - J1: Rationale: Hazard is mitigated by providing control measures for all its causes
 - C2: Cause 'Sensor failure to detect air bubble' is addressed by control measures
 - C3: Cause 'Safety subsystem failure to stop the pump' is addressed by control measures
 - C4: Cause 'Pump does not stop on request' is addressed by control measures







Conclusions

- Uniform model interface is sufficient for establishing and maintenance of assurance case relations to system model
- Use of GUIDs in system models is effential for references maintenance

Further work

- Case studies for other types of models
- Verification function to detect model changes
- Maintenance of the instantiation reference tables
- Integration with SACM 2.0 (Terminology package)







- Uniform model interface will facilitate establishing and maintaining assurance case relation to system models
 - We expect this to be easier for safety engineers
- The established relations are:
 correct as they rely on directly on existing models
 up to date (this can be verified at any moment of time)
- System model changes can be propagated to the safety argument





Thank you for your attention

