원자력 발전소 안전 소프트웨어의 safety case pattern 작성을 위한 문헌 리뷰 기반의 pattern 작성 범위 분류

정세진○, 손준익, 유준범
건국대학교 컴퓨터 정보통신 공학과
{jsjj0728, sji6227, jbyoo}@konkuk.ac.kr

Classification of the scope of safety case patterns for nuclear power plants safety-related SW: A literature review

Sejin Jung○, Junik Son, Junbeom Yoo
Division of computer science and engineering, Konkuk university

요 약

안전 필수 시스템 (safety-critical system)은 그 사고가 사회에 큰 영향을 미치기 때문에 안전성을 우선하여 개발해야 한다. Safety case (안전 논증) 방법은 안전성이 중요한 소프트웨어/시스템 개발 시 수행하는 안전 관련 활동들 간의 논증 구조 작성을 통해 시스템이 용인되는 수준의 안전성을 갖추었는지 증명하는 기법으로 사용된다. Safety case pattern은 각 대상의 safety case 작성 시 반복되는 부분들을 효율적으로 재사용하기 위해 사용되며, 효과적인 패턴 제공을 위해서는 패턴의 구조뿐만 아니라 제공되는 패턴의 범위, 내용, 수준 또한 중요하다. 본 논문에서는 원자력 발전소의 안전 소프트웨어를 대상으로 한 safety case pattern 작성을 위해 기 제안된 다른 도메인의 패턴 내용을 바탕으로 safety case pattern의 작성 내용, 범위, 수준 등을 고려한 분류를 수행하고 제안한다.

1. 서 론

원자력 발전소와 같은 안전 필수 시스템(safety-critical system) 은 사고로 인한 영향이 인명, 환경 등에 크게 미치기 때문에 안전성 확보가 필수적이다. 안전 필수 시스템의 소프트웨어 또한 마찬가지로 신뢰성, 안전성 확보가 매우 중요하며, 관련 표준, 법규 등에서는 소프트웨어 생명주기에 걸쳐 안전성 분석, 검증, 형상관리 등을 수행하도록 규제하고 있다. 이 때 각각의 활동들로 인한 결과물들의 단순한 검토만으로는 개발, 검증, 안전성 분석, 형상 관리의 적정성 등을 파악하기 힘든 점이 있다. 이를 위해 여러 안전이 중요한 분야에서는 표준이나 규제를 통해 safety case (안전 논증) 방법론을 적용하여 안전 보증을 확인하도록 하고 있다.

Safety case는 시스템이 용인되는 수준의 안전성 (acceptably safe)을 갖추었는지를 구조적이고 명시적으로 표현하는 기법으로 직접적인 안전성 분석 기법은 아니지만, 안전성 분석의 한 갈래로 사용되고 있다. Safety case 작성에는 주로 GSN (Goal structuring notation) 표기법이 사용된다. 다음 <그림 1>은 GSN의 표기법에 대한 그림이다. <그림 1>의 notation을 용도에 맞게 goal과 strategy를 생성해 연결하는 식으로 구성된다. 이러한 safety case 작성시 유사한 시스템, 소프트웨어의 경우 같은 프로세스가 적용되는 경우가 많기 때문에 반복되는 사항들이 많이 나타나는 점을 확인 할 수 있다. Safety case pattern[2]은 반복적으로 나타나는 구조들을 효율적으로 재사용하기 위해 제안되었다. 또한 일정 수준 이상의 안전 논증 구조 작성을 보장하기 위해서도 사용될 수 있다. 현재 기존의 몇몇 연구들을 통해 도메인 별로 safety case pattern이 제안되고 있다. 하지만 원자력 발전소의 안전 소프트웨어를 대상으로 한 패턴의 연구는 부족한 상황이다.

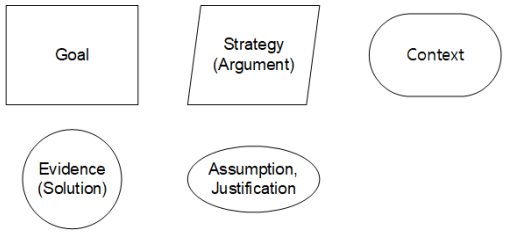

그림 1. GSN에서 사용되는 표기법

이러한 safety case pattern 작성에는 pattern language, GSN 기반의 structure 작성도 중요하지만 제공되는 내용 (contents) 또한 매우 중요하며, 어느 수준에서 제공할 것인가에 대한 고찰이 필요하다. 기 개발된 SW safety case pattern들을 확인해 보면 각 패턴들은 적용 범위, 적용 수준에서 여러 가지 차이점을 보인다. 이에 본 논문에서는 원자력발전소의 안전 소프트웨어의 safety case pattern 작성에 앞서 효과적인 safety case pattern 작성을 위해 기 개발된 safety case pattern들에 대해 리뷰를 수행하고, 이를 바탕으로 패턴의 작성 내용, 범위, 수준 등을 고려한 분류를 수행 및 제안한다. 또한 각 분류 별 패턴 작성시의 장단점에 대해 논한다.

2. Safety case pattern

Safety case pattern은 시스템/소프트웨어의 안전 논증 작성 시 자주 사용되는 공통된 구조를 재사용하기 위한 template으로 효과적인 사용을 위해 정형화된 카테고리가 중요하다. [2]에서는 디자인 패턴에서 자주 사용되는 항목들을 기반으로 safety case pattern language를 제안한 바 있다. Pattern language는 ‘name and classification,’ ‘intent,’

'also known as,' 'motivation,' 'applicability,' 'structure,' 'participants,' 'collaboration,' 'consequence,' 'implementation,' 'simple text,' 'known uses,' 'related pattern,' 으로 구성되어 제공되는 패턴을 명확하게 표현하도록 하였다. 특히 'structure' 항목은 GSN을 이용해 패턴의 주 내용을 제공한다. 또한 pattern이 실제 대상의 instance 화를 위해 multiplicity, uninstantiated entity, optional extension을 사용해 패턴들을 표현한다. <그림 2>는 safety case pattern의 structure 예제에 대한 그림이다 [3]. <그림 2>와 같은 구조를 제공해 필요에 따라 내용을 변경해 가면서 재사용 할 수 있도록 한 구조이다.

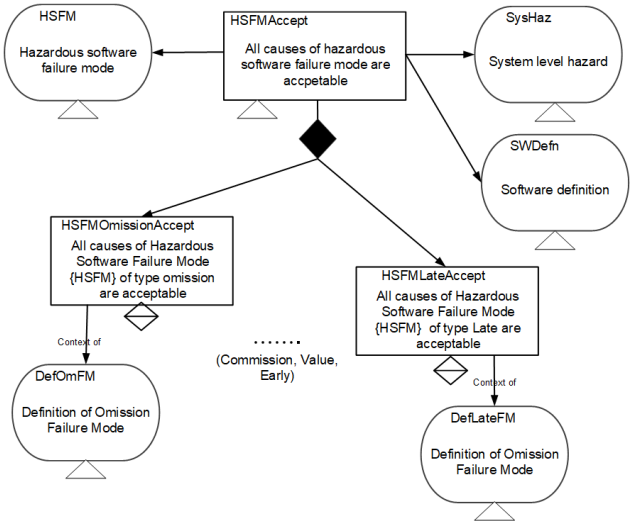

그림 2. Safety case pattern structure의 예제 [3]

3. Safety case patterns의 분류

3.1 Existed safety case pattern 조사 및 리뷰

기존에 SW 및 소프트웨어 기반 시스템을 대상으로 하는 safety case pattern에 대해 수행한 연구들이 존재한다. 각각의 패턴은 특정 도메인을 대상으로 한 패턴도 있지만 도메인과 상관없이 SW를 대상으로 하고 있는 패턴도 존재한다. 본 논문에서는 safety case pattern에 대한 키워드로 약 40 여 편의 논문을 수집하여 확인하였고, 해당 논문 들 중 동일한 저자의 유사한 논문 및 safety case 적용 논문, 연관성이 없는 논문을 제외한 논문들을 다음 <표 1>과 같은 카테고리로 분류할 수 있었다. 공간의 문제로 전체 논문들의 리스트를 포함하지 못하고 일부에 대해서 설명한다.

첫 번째는 패턴을 제안하거나 패턴 기반으로 safety case를 작성하는 연구들로 구성된 논문으로 13편을 확인할 수 있었다. <그림 2>는 1번 분류에 나타난 safety case pattern의 예제이다[3]. <표 1>에서 2, 3 번으로 분류된 논문들은 패턴을 직접적으로 제안하지는 않지만 argument type, argument scheme, safety case principle들을 통해 safety case 작성 시 유용한, 그리고 패턴화 될 수 있는 내용들을 제공하고 있는 논문이다. <표 1>에서 2, 3으로 분류된 논문들도 패턴을 작성함에 있어 고려할 만한 정보들을 제공하고 있지만 본 논문에서는 우선 1번으로 분류된 논문들에 대하여 논한다.

표 1. 각 논문들의 분류

분류	명세	비고
1. Safety case pattern and pattern-based approach for safety case	Safety case 작성을 위한 pattern 제안 및 적용에 관련된 논문	13 편
2. Pattern-based approach of safety/assurance argument	직접적인 safety case pattern 이외의 패턴 기반의 argument, assurance 작성을 위한 논문들	2 편
3. Other perspectives for software safety case (with simple pattern)	Safety case 작성을 위한 principle, review에 관련된 논문들	4 편

(기타 safety case application 관련 10여 편 및 safety case modelling, tool 등 pattern과 연관성이 떨어지는 논문 제외)

이처럼 각 논문들을 수집하여 확인하고, 패턴 사용 방법에 따라 분류를 수행하였다. Safety case pattern을 직접적으로 제안하는 1번 분류의 속한 논문들을 자세히 살펴보면, safety case의 instance를 생성할 경우에 필요한 instantiation 정도가 모두 다름을 확인할 수 있다. 즉, 각 패턴의 abstraction 수준이 다르다는 것으로 볼 수 있다. 다음 절에서는 이 점에 초점을 두어 논한다.

3.2 Safety case pattern의 작성수준 분류

앞서 확인한 바와 같이 각 safety case pattern의 structure를 살펴보면 언급된 내용과 수준에 차이점이 있어 instance 생성 시 필요한 정보 및 요구되는 것들이 달라진다. 본 논문에서는 원자력 발전소 안전 소프트웨어를 위한 safety case pattern 개발을 위해 선 작업으로 해당 abstraction 수준을 분류하고 각각의 장단점에 대해 확인하여 패턴 작성 시 기준을 생각해 보고자 하였다. 본 논문에서는 이러한 차이들을 4 단계로 카테고리화 하였다. 해당 분류들은 <표 2>를 통해 확인할 수 있다. 한 논문에 여러 카테고리에 해당하는 패턴들을 제안한 경우도 존재한다.

각각의 분류에 대해 살펴보면 우선 'Structural composition'으로 분류한 항목은 pattern을 개발 시에 goal, strategy의 내용보다 간단한 내용만을 포함하고, structure 구성을 위주로 초점을 두어 제공하는 패턴들이 포함되는 분류이다. 다음으로 'High-level contents composition'은 높은 수준에서 abstraction 되어 제공되는 패턴들을 의미한다. 예를 들어 'All hazards in SW should be mitigated' 와 같이 상위 수준의 abstraction된 내용을 제공하는 패턴들이 해당 카테고리에 속한다. <그림 3>은 해당 분류의 패턴에 대한 예제이다. 세 번째 카테고리인 'Concrete contents composition'의 경우는 두 번째와 유사하지만 대상 도메인 혹은 안전성 분석 활동 타입 등에 대해 구체적인 내용을 포함하고 있는 패턴들을 위한 분류이다. 예를 들어 위험 경감 관련 내용에서 포함되어야 할 hazard의 타입을 지정하는 것과 같이 낮은 수준의 abstraction만 적용되어 구체적인 내용들을 포함한 패턴들이다. <그림 2>의 패턴이 해당 분류에 속한다고

할 수 있다. 마지막 분류는 domain specific한 내용뿐만 아니라 대상에 구체적인 내용까지 포함해 safety case의 instance와 차이가 없는 패턴들을 포함하는 분류로 지정하였다. 해당 분류의 패턴[6]은 빈도가 적지만, 경우에 따라서는 필요에 따라 쉽게 사용할 수 있는 종류가 될 수 있다.

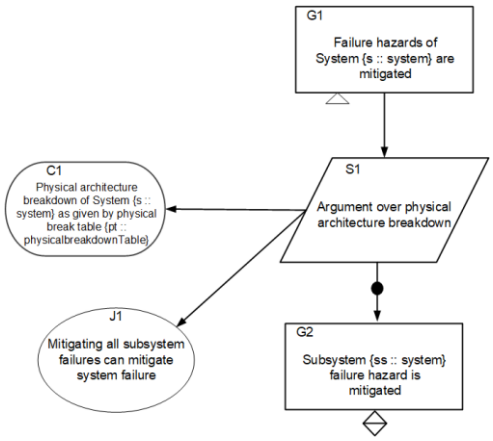

그림 3. [7]에 나타난 2 분류 패턴의 예제

표 2. 4 카테고리로 구분된 pattern들의 차이점

No.	Classification	Description	비고(편-개수)
1	Structural composition	Notation들의 structure만으로 구성	2-11 ([4][5])
2	High-level contents composition	높은 수준에서 abstraction된 내용만 제공 (*e.g. All hazard mitigation*)	10-36 ([5][7] 등)
3	Concrete contents composition	대상 domain에 specific한 내용을 포함한 abstraction (*e.g. failure type 정의*)	4-11 ([3] 등)
4	Detailed contents composition	가장 낮은 수준의 abstraction, safety case의 instance에 근접	1-3 ([6])

<표 2>를 통해 분류한 카테고리에서 detail 수준이 높아질수록 패턴을 적용할 때 작성자가 직접 작성해야 하는 instantiation contents가 적어진다. 가장 높은 수준의 detail을 포함한 패턴(abstraction이 낮은 패턴)은 safety case 와 동일한 수준까지도 작성되어 같은 safety case를 바로 사용하는 것과 다를 바 없어질 수 있다. Abstraction 수준이 높아져 2번 분류로 갈수록 분석가가 작성해야 하는 내용들이 많아지지만, 다양한 대상, 도메인에 사용할 수 있는 여지가 증가한다는 장점이 있다. 그렇기 때문에 잘 작성된 패턴이 있다면 여러 도메인에 쉽게 적용할 수 있다. 단점으로는 많은 수준의 내용이 분석가에 의해 판단되고 작성되어야 하기 때문에 패턴을 이용한 instance의 결과가 크게 차이 날수 있고, 재사용하여 일정 수준 이상의 safety case 결과를 생성하려는 의도가 약해질 수 있다. 3번 분류는 중간 정도의 내용이 포함되어 같은 수준의 프로세스, 표준, 규제를 통해 개발된 경우 반복해서 사용할 수도 있고, 일정 내용을 지속적으로 제공할 수 있다는 장점도 있다. 이처럼 수준에 따라 여러 abstraction을 생각 할 수 있으며, detail 수준에 따라서 패턴의 의도, 목적이 달라지고 장단점이 달라진다.

실제 패턴을 개발할 때 detail을 높이게 되면 패턴 개발과 instance 개발과 차이가 없어지게 되고, abstraction 수준을 높이게 되면 domain, target 과 떨어진 패턴을 개발하게 된다. 특히 detail 정도가 높은 패턴을 개발 시 높은 제약성으로 인해 해당 패턴의 재사용 가능성 여부에 심도 있는 고민이 필요해진다는 점이 있다. 패턴 개발 시 이러한 점에 초점을 맞추어 수준을 정할 필요가 있다. 특히 이런 점을 명확하게 정하지 않는다면 패턴 개발에 혼동을 불러 일으킬 수 있다.

원자력 발전소 시스템은 복잡한 시스템으로 다양한 소프트웨어가 사용된다. 해당 소프트웨어 들을 대상으로 하는 패턴에 대해 고려해 보면, 다양한 소프트웨어들이 사용되기 때문에 적정한 수준의 abstraction이 되어 여러 소프트웨어를 대상으로 할 수 있어야 한다. 따라서 4번 분류는 적합하지 않다고 할 수 있다. 원자력 도메인의 내용을 포함한 3번 분류의 패턴을 작성하거나, 몇몇 내용은 2번 수준에서 작성하는 것이 적합하다고 할 수 있다.

4. 결 론

본 논문에서는 기존에 개발되었던 safety case pattern 들에 대해 리뷰를 수행하고, 이를 바탕으로 패턴 작성의 범위, abstraction 수준 등을 분류하여 원자력발전소의 안전 시스템 소프트웨어를 대상으로 하는 safety case pattern 개발 시 적용할 수 있는 수준에 대한 분류를 수행 및 제안하였다. 본 논문에서 수행한 pattern의 작성 분류는 4 가지로 분류되었으며, 각 분류의 장단점에 대해 논하였다. 이처럼 수행한 작성 범위 분류를 통해 safety case pattern 개발 시 필요한 범위내에서 패턴을 개발하는데 도움이 될 것으로 기대한다.

Acknowledgement

이 논문은 2018년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(NRF-2017R1D1A1B03030065)

참고문헌

[1] IEEE, “IEEE Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations,” 20110

[2] T. P. Kelly, J. A. McDermid, “Safety Case Construction and Reuse using Patterns,” Safe Comp 97, pp.55-69, 1997

[3] R.A. Weaver, “The Safety of Software - Constructing and Assuring Arguments,” Ph. D. Thesis, University of York, 2003.

[4] Ewen Denney, Ganesh Pai, “Composition of safety argument patterns,” International Conference on Computer Safety, Reliability, and Security,” pp.51-63, 2016.

[5] T. P. Kelly, “Arguing safety - a systematic approach to managing safety cases,” Ph. D. Thesis, University of York, 1998.

[6] Robert Palin, Ibrahim Habli, “Assurance of automotive safety - A safety case approach,” International Conference on Computer Safety, Reliability, and Security, pp.82-96, 2010.

[7] Ewen Denny, Ganesh J. Pai, “Safety case patterns: theory and application,” NASA/TM-2015-218492, 2015.