**DRAFT**

Full paper published in
**Polish Journal of Environmental Studies**
**Vol. 17, no. 4C (2008)**

# Argument Strategies and Patterns of the Trust-IT Framework

Janusz Górski, Łukasz Cyra, Aleksander Jarzębowicz, Jakub Miler
Department of Software Engineering, Gdańsk University of Technology,
Narutowicza 11/12, 80-952 Gdańsk, Poland
e-mail: {jango, lukasz.cyra, olek, jakubm}@eti.pg.gda.pl

## Abstract

The paper concerns the methodological and tool support framework for trust cases called Trust-IT. The concept of trust case extends the well-established concept of safety case to make it applicable to a wider range of analyzed properties, acceptable evidence and implemented business models. In the paper, we focus on the strategy of trust case development, which depends on the property being analyzed in the trust case. We present the risk-driven and the standard-driven approach and propose a number of argument patterns. We also discuss the trust case development process and provide more details about available tool support. In conclusion we summarize the present status of the Trust-IT framework and the experience from its application as well as give plans for further development.

**Keywords:** trust case, Trust-IT framework, safety, security, privacy

## 1. Introduction

Trustworthiness of IT systems and services is becoming an issue of growing importance, in particular with respect to such properties as safety, privacy and security. Trustworthiness means that there are 'good' reasons for trusting that a given object possesses a distinguished property (or set of properties). We are interested in situations where the 'grounding' for trust is in an explicit analysis of trustworthiness, which can be verified and assessed. Safety is one of the properties for which we expect particularly high assurance and this expectation is reflected in the related standards, regulations and recommendations. In particular, it is commonly required that safety is justified in an explicit safety case [16, 19] and the methodological and tool support for safety cases has been established (e.g. [1, 14]).

The need to extend the conventional safety case approach has been recently recognized and the international effort has been organized into a series of Workshops for

Assurance Cases held in Washington DC, Florence, Ispra and Edinburgh. In our work we extended the idea of safety case to trust case [8] in several aspects discussed below.

We extend the scope of the properties addressed by the case. In our work we presently focus (in addition to safety) on privacy and security. Furthermore, we see the applicability of trust cases in virtually any situation where trustworthiness of a postulated statement is worth analyzing with explicit argumentation. This opens a very wide range of potential applications not necessarily restricted to technical domains.

We do not distinguish any particular criteria of an 'acceptable' case. This issue is to be decided by the trust case user and/or an expert acting on his/her behalf. Instead, we are focusing on general aspects of argumentation and the question of how to build a valid argument based on the available evidence. Consequently, the compellingness of a trust case becomes a subjective issue (we require that the trust case analyzes trustworthiness explicitly and the assessment of this analysis is left to the user). Nevertheless, we are still interested in the assessment of the compelling power of a trust case and we work towards providing an appropriate support to this task [4].

We assume that trust cases can be developed for open and distributed systems, which raises the issue of distributed ownership of the trust case, distributed evidential material and cooperative development of the case. We are researching into the trust case development and maintenance process understood as a distributed business process with distinguished goals, roles to be played, performed activities and managed resources. We distinguish different types of such processes which depend on the particular business objectives (examples are: system/service acceptance or standards conformity).

We recognize that a trust case is the means of communication to convey a message which serves to build user's trust to a system. We are developing tools for publishing trust cases in the Internet to make them accessible to a wide range of users.

The results of the above research are contained in the Trust-IT framework [7], which includes the language for trust cases, the trust case development methodology and the toolset which facilitates its application.

The paper introduces the notion of argument and language for trust case development. Next, we present the strategies which we applied while developing the trust cases as well as provide argument patterns encountered in trust cases. Then the process of trust case development is described stressing its collaborative nature and describing present tool support. In conclusions we list our present experiences with trust cases from different industrial and research projects and indicate the plans for the future.

## 2. Trust case language

The argument model in trust cases follows Toulmin's argument model [21]. It includes a *conclusion* to be justified and *premises*, which provide the basis for the inference. Following [13, 18, 22], we assume that the argument establishes a *conveyance* relationship between them. This relationship is established by the inference rule which is explained and justified by the warrant of the argument [21]. As we admit warrants which not necessarily are deductively valid, we admit *defeasible reasoning* [20] to be used in trustworthiness analysis.

A trust case has a tree-like structure and is composed of nodes of different types. The nodes are elements of the language for expressing trust cases. The syntax of the language is defined by the allowed parent-child relationships of nodes. The node types and their allowed relationships are shown in Figure 1, where an arrow represents that a node of a given type can

be a child of a node of the type pointed to by the arrow. The graphical symbols in the figure are those used in the trust case editing tool of the Trust IT framework.

The basic node type is the *claim* (denoted CL), which contains a concluding statement to be analyzed. A node of type *argument* (denoted Arg) links the claim to the corresponding premises and uses the *warrant* (denoted W) to explain the conveyance. A premise can be of the following types: an *assumption* (denoted As) represents a premise which is not further analyzed in the trust case; a claim represents a premise to be further analyzed by a more detailed argument; and a *fact* (denoted F) represents a premise which is obviously true or otherwise is supported by some evidence. The evidence is provided in external documents which are pointed to by nodes of type *reference* (denoted Ref).

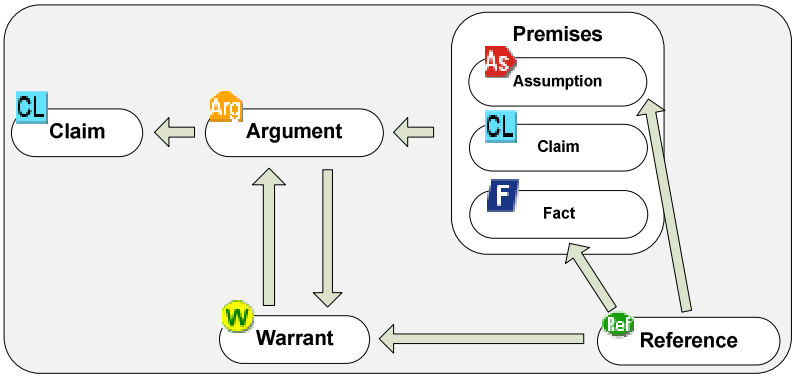


Figure 1. Trust Case meta-model

Additionally, everywhere in the argument tree an *information* node (denoted i), which contains explanatory information and does not constitute part of the argument, can be placed, and any node can support the information node.

## 3. Argument development strategies

Trust cases are developed top down, starting from a chosen aspect to be analyzed, the selection of which is outside the scope of *Trust-IT* and depends on the context within which the trust case is developed. For IT systems it is often safety (of e.g. an e-health service), privacy (of a user of a service), and security of sensitive assets.

Below we give an overview of two argument strategies which we apply to deal with these crucial aspects of trustworthiness while developing trust cases. Then we provide examples of some commonly utilized argument patterns.

### 3.1. Risk-driven strategy

While analyzing user's safety we take a risk related perspective assuming that trustworthiness increases if it is demonstrated that the risks related to the user's life and health are known and acceptable. Therefore, analysis of safety risks is a natural starting point and the trust case structure is shaped by the hierarchy of identified risks.

The analysis focuses on the system and the risks resulting from its use in the target environment and the risk management process (based on [11]) is implemented. The process is supported by an internet-based tool, RiskGuide, which supports risks identification, helps in risk rating, and finally publishes the risk assessment reports [15]. The risks are then examined from the mitigation and acceptance perspective. For each risk, a postulate that the risk has

been properly addressed becomes a claim to be included and supported in the trust case. The risks can often be arranged hierarchically according to the cause-consequence relationship, which is reflected in the trust case structure.

We were running an e-health case study of trustworthiness analysis for the system developed in PIPS project [17]. The system monitors its users' health related parameters and provides advice related to health and lifestyle. An example of risk driven decomposition of the trust case developed in the project is shown in Figure 2. The two generic safety risks considered were:

1. the system causes deterioration of the patient's health,
2. the system impairs the (otherwise natural) health improvement.

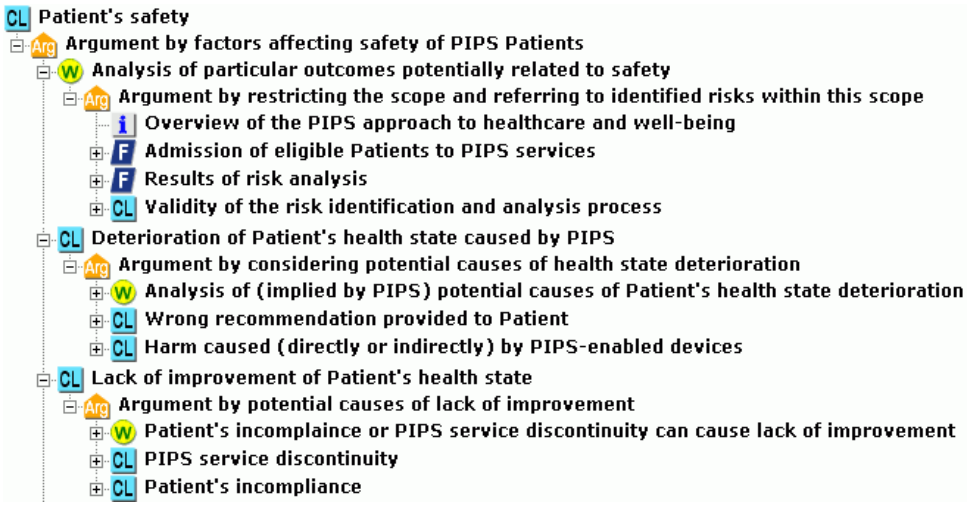


Figure 2. Example risk-driven analysis of safety

The claims on mitigation of these risks constitute the premises of the top claim. The risk analysis itself is referred to in the top argument's warrant. The risks were then analyzed for their causes, and the claims about their mitigation were included as premises.

### 3.2. Standard-driven strategy

While analyzing security and privacy we applied the approach driven by standards [3]. It comprised selection of relevant standards and demonstration of conformity.

We use *trust case templates* to demonstrate conformity with standards. The template is a generic trust case derived from the text of the norm with references to the text of the standard and to additional sources of information. The central component of the template is Template Argument Structure (TAS), which argues the conformity with the standard. TAS is decomposed recursively. At the top there is a claim stating that all the requirements of the standard have been fulfilled, which is supported by an argument referring to the claims postulating the fulfillment of particular requirements. This decomposition is continued to the level of individual requirements which has to be demonstrated by providing appropriate evidence. In addition, TAS includes auxiliary information which facilitates interpretation and demonstration of the requirements, as well as dependencies between requirements. Before applying it in the certification process, the template gets acceptance of a certified auditor.

We have developed a trust case template for ISO/IEC 27001:2005 [12] to support demonstration of conformity with recommended information security management practices.

We have also developed a template reflecting the privacy related recommendations derived from a set of relevant documents, including among others: Directive 95/46/EC [6], Directive 2002/58/EC [5] and HIPPA Privacy Rule [9]. A fragment of the template for ISO/IEC 27001 is given in Figure 3.

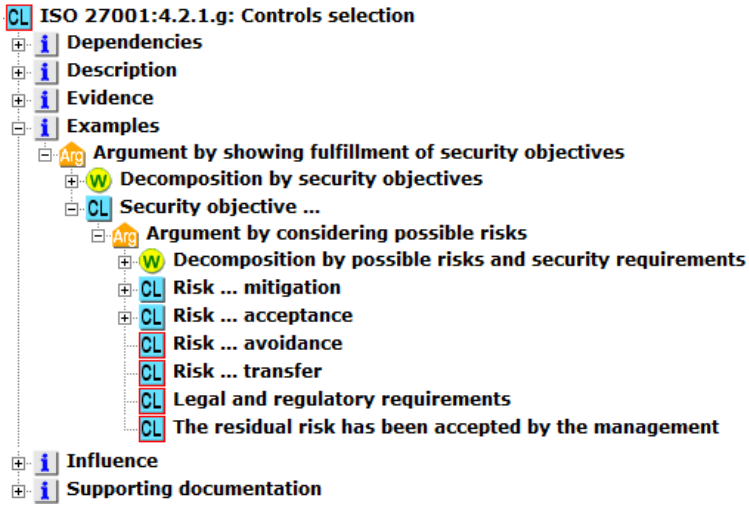


Figure 3. ISO/IEC 27001 standard template fragment

The above fragment contains an open claim representing the requirement related to controls selection from item *4.2.1.g* of the standard. The claim is supported by information nodes which trace to the interdependent requirements of the standard (*'Dependencies'* and *'Influence'* nodes), list the documents and other requirements of the standard used to demonstrate the fulfillment of the given one (*'Evidence'* node), contain example arguments (*'Examples'* node) and refer to other useful documents which can be helpful while achieving conformity (*'Supporting documentation'* node).

## 4. Argument patterns

The argument structure in the trust case is developed by referring to a knowledge base which stores a range of different argument patterns applied in trust cases. Examples of such argument patterns are given in table 1.

Table 1. Examples of argument patterns

| Pattern name | Premises | Conclusion | Warrant |
|---|---|---|---|
| Argument from expert opinion | 1. Person E is an expert in domain D<br>2. E asserts that A is true<br>3. A is about D | A | Assertions related to D, given by an expert in D are plausibly true |
| Argument from compliance with 'best practices' | 1. R is an accepted recommendation in a domain D<br>2. R is applicable to an object O in environment E<br>3. R focuses on establishing a property P<br>4. Object O in environment E is compliant with R | O in E exhibits P | R encompasses what is recognized as a good and recommended practice to achieve P of O in E. Therefore, compliance with R justifies that exhibiting P by O is plausibly true |
| Argument from | 1. System O is exploited in the | O in E is | U considers O in E as trustworthy if |

| | | | |
|---|---|---|---|
| risk analysis | environment E<br>2. U is a user of O<br>3. Risk analysis of O in E identified set R of risks potentially affecting U<br>4. Risk analysis was adequate<br>5. All risks from R are analyzed and if necessary mitigated in a demonstrable way | trustworthy for U | the risks potentially affecting U are identified and mitigated. Adequate risk management provides sufficient coverage of relevant risks and adequate risk mitigation supports risk acceptance by U. Therefore, trustworthiness of O for U is plausibly established |
| Argument from decomposition | 1. Object O has property Q1<br>2. Object O has property Q2 | O has property Q | Possessing properties Q1 and Q2 by O most likely is equivalent to possessing property Q by O. Therefore, possessing property Q by O is plausibly true |

## 5. Trust case development process

Five roles have been identified to be assigned to stakeholders for the trust case development and maintenance process:

- *Manager* – supervises and coordinates the process,
- *Developer* – builds trust case, attaches evidence, maintains trust case integrity,
- *Contributor* – supplies evidence and argumentation, reviews the trust case and provides feedback,
- *Assessor* – assesses the trust case against a defined set of criteria,
- *Viewer* – studies the trust case to find the argumentation supporting the trust aspects of her/his interest.

The process of trust case development involves all the stakeholders (users, auditors, regulatory authorities, professional bodies, system developers etc.) who for some reasons have justified interest in the trust aspects included in the trust case. Their active participation is crucial for the effectiveness of the trust case development process. The development is an iterative process of extending and validating the current trust case.

The evidence provided by contributors and maintained in a trust case is basically a set of documents, which are referred to from the facts. It can be of very diverse nature, e.g. excerpts from the system design documentation, review reports, test results, expert opinions and so on. In general, such documents say something about the system design, its development process, its verification and validation results or its target environment.

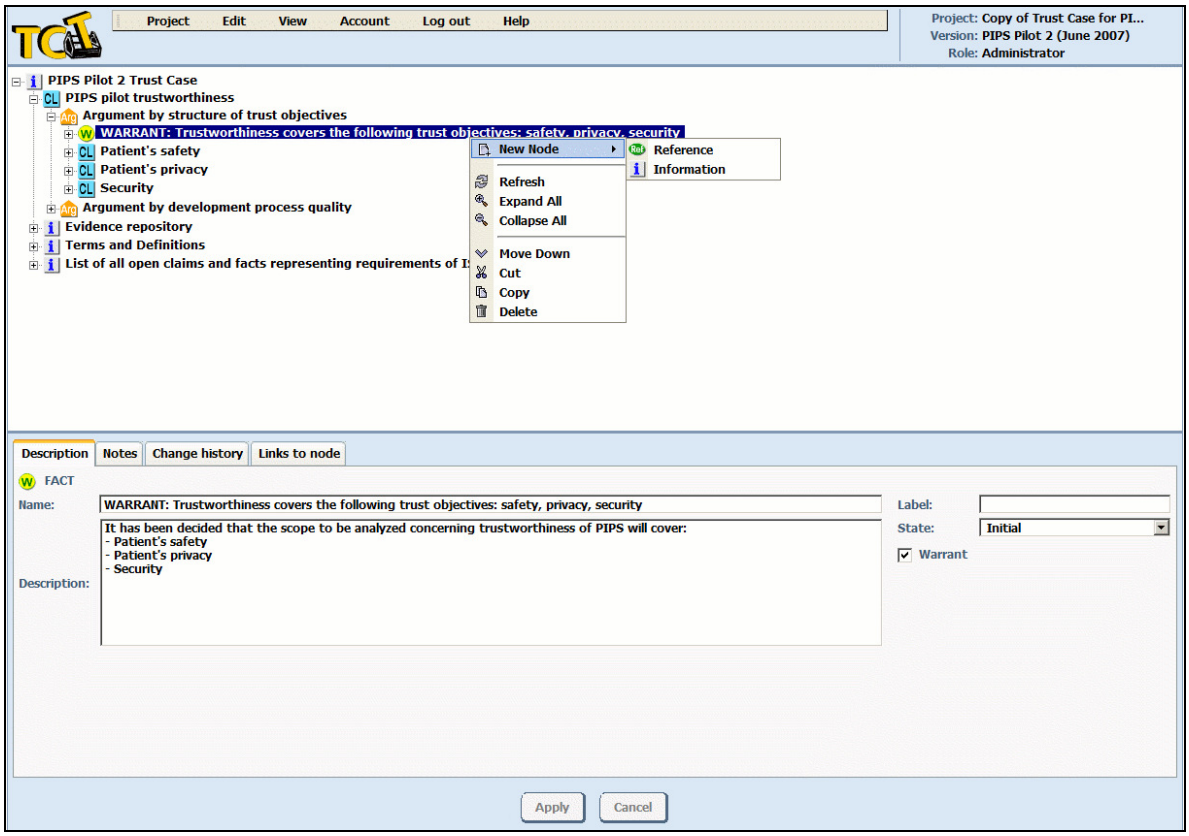


Figure 4. Sample trust case presented in TCT

To support the collaborative trust case development process we have created a dedicated software tool called TCT (Trust Case Toolbox). TCT is a rich internet application (RIA), which supports sharing of all trust case data stored in an on-line repository and provides for high level of interactivity while accessing and editing trust cases. Figure 4 presents a sample screenshot of TCT with a trust case tree in the upper part, the details of the selected node in the lower part of the screen and expanded context menu.

TCT supports a team of stakeholders, which can be distributed geographically, by providing access to the up-to-date version of the trust case. TCT also stores full information about the history of trust case development by recording each modification of a particular node and by providing versioning mechanism for creating snapshots of whole trust cases. The evidence is stored in a repository (centralized or distributed), which is integrated to the trust case via references. The references may also include security credentials, if required by the repository. The evidence must be kept consistent and up-to-date, which is presently addressed by adequate change management procedures implemented manually.

## 6. Conclusions

In this paper we reported our experiences with the development of trust cases. Trust case is an argument structure which explicitly shows how the available evidence and explicit assumptions support the postulated conclusion. We are developing a framework, called Trust-IT, to support development and application of trust cases in different business scenarios. This process involves a number of case studies and we use the feedback from those case studies to plan for the subsequent development steps.

The present scope of our case studies includes:

- A prototype system for drugs distribution and application in a hospital environment (5th EU FR Project DRIVE – 'DRugs In Virtual Enterprise'). The trust case focused on the safety objective and contained about 500 nodes.
- An open platform for e-health services (EU 6th FP Project PIPS - 'Personalized Information Platform for life & health Services' [17]). The main PIPS trust case covers safety, privacy and the conformity to a security management standard. It comprises now up to some 700 nodes.
- An embedded Wireless Sensor Network (WSN) based platform for health and life related services (EU 6th FP Project ANGEL – 'Advanced Networked embedded platform as a Gateway to Enhance quality of Life' [2]). The work on the ANGEL trust case focusing on safety, privacy and security is ongoing.
- Standards Conformity Framework (SCF) [3] which supports conformity with standards achievement and assessment. The framework applies trust case templates to represent the requirements which have to be fulfilled to demonstrate the conformity. Currently templates for ISO 14971, ISO/IEC27001:2005 and Common Criteria are available.

The plans for Trust-IT development include:

- Assessment of SCF in cooperation with a certifying institution and companies aiming at certification against ISO/IEC27001:2005 and possibly ISO9000 series of standards.
- Extending the TCT tool with the viewing mechanism providing different perspectives on a trust case depending on a user profile.
- Extending Trust-IT with a method of appraising the contents of a trust case by an expert to express his/her opinion on the 'strength' of the argumentation and the evidence.
- Investigating the issues related to the distributed ownership of the parts of a trust case (including the related evidence), in particular in relation to the protection of critical infrastructures.

More details about Trust-IT and the related research projects can be found at [10].

## Acknowledgements

This work was partially supported by the project ANGEL - 'Advanced Networked embedded platform as a Gateway to Enhance quality of Life' (IST project 2005-IST-5-033506-STP) and by the project PIPS – 'Personalized Information Platform for health and life Services' (Contract No. 507019 IST2.3.1.11 e-Health) within the European Commission 6th Framework Programme. The authors also appreciate the comments by Andrzej Wardziński.

## References

1. Adelard Safety Case Editor (ASCE), Adelard, UK, 2006, (http://www.adelard.com/web/hnav/ASCE/index.html).
2. ANGEL project website (http://www.ist-angel-project.eu).
3. Cyra Ł., Górski J.: *Supporting compliance with safety standards by trust case templates*, Proceedings of ESREL 2007 conference, Stavanger, Norway, 2007.
4. Cyra Ł., Górski J.: *Supporting expert assessment of argument structures in trust cases*, 9th International Probabilistic Safety Assessment and Management Conference, China, 2008.
5. Directive 2002/58/EC The processing of personal data and the protection of privacy in the electronic communications sector, 2002.

6. EU Directive 95/46/EC - The Data Protection Directive.
7. Górski J.: *Trust-IT – a framework for trust cases*, Workshop on Assurance Cases for Security - The Metrics Challenge, The 37th Annual IEEE/IFIP International Conf. on Dependable Systems and Networks, June 25 - 28, Edinburgh, UK, 2007.
8. Górski J., Jarzębowicz A., Leszczyna R., Miler J., Olszewski M.: *Trust Case: justifying trust in an IT solution*, Reliability Engineering and System Safety - Vol. 89 (2005), pp. 33-47.
9. HIPAA Privacy Rule, Health Insurance Portability and Accountability Act, Part 164, Subpart E - Standards for Privacy of Individually Identifiable Health Information, U.S. Department of Health and Human Services.
10. Information Assurance Group homepage (http://iag.pg.gda.pl/iag/).
11. International Standard ISO/FDIS 14971: Medical devices - Application of risk management to medical devices, ISO 14971:2000 Annex A - Medical Devices Safety Questionnaire.
12. ISO, ISO/IEC 27001:2005 Information technology – Security techniques – Information security management system – Requirements, 2005.
13. Katzav J., Reed C.: *On Argumentation Schemes and the Natural Classification of Arguments*, Argumentation 18 (2), 2004, pp. 239-259.
14. Kelly T., McDermid J.: *A Systematic Approach to Safety Case Maintenance*, Proc. SAFECOMP 1999.
15. Miler J.: *A service-oriented approach to the identification of IT Risk*, Proc. IEEE TEHOSS 2005 conference, Gdańsk, Poland, September 28-30, 2005.
16. Ministry of Defence, Directorate of Standardisation. Defence Standard 00-55 Issue 2: The procurement of safety critical software in defense systems.
17. PIPS project website (http://www.pips.eu.org).
18. Prakken H.: *AI & Law, Logic and Argument Schemes*, International Congress of Comparative Cultures and Legal Systems, Mexico City, 2004.
19. Railways (Safety Case) Regulations 2000 Health and Safety Executive.
20. Stanford Encyclopedia of Philosophy (http://plato.stanford.edu/entries/reasoning-defeasible/).
21. Toulmin S.: *The Uses of Argument*, Cambridge University Press, 1958.
22. Walton D.: *Justification of Argument Schemes*, Australian Journal of Logic, No. 3, 2005, pp. 1-13.