ARGEVIDE

Challenges in providing support for management of evidence based arguments

Abstract. The paper considers selected challenges related to the application of evidence based arguments and maps them on the tool support level. In particular, we consider: communication and teamwork, handling large arguments, evidence management and integration, argument assessment and communication, change control and reporting, evidence reuse, user data security and argument portfolio management. Then we explain how these challenges have been responded to in NOR-STA, a tool supporting applications of evidence based arguments. The paper is based on our experience with applying NOR-STA to support conformance cases and assurance cases in more than 120 organizations.

1 Introduction

Using evidence-based arguments for demonstrating system safety and other properties is becoming a common practice in several sectors. Standards such as ISO 26262 (for automotive E/E systems) [1] and IEC 62278 (for railway) mandate the use of safety arguments [2]. In 2010, US FDA issued a draft guidance for use of safety argument for medical devices [3]. ISO 15026 describes general requirements for assurance case structure and its use in the system lifecycle [4].

Argument structures can be used not only for demonstrating system assurance but also for other purposes like demonstrating conformance [5] or comparative conformance monitoring [6]. Although the argumentation strategies may be different, depending on the business objectives (demonstrating and assessing assurance, achieving and/or certifying conformance and so on) and the domain of application (for instance nuclear, automotive, railway, medical), the underlying problems of argument representation, assessment, visualization and maintenance are common. This observation led us to the concept of NOR-STA, a generic platform of services supporting applications of evidence based arguments, on top of which different functional packages are located to support more specialized applications of arguments.

Our experience is based on more than 120 conformance and assurance case projects conducted in cooperation with different organizations. The conformance case applications included standards related to hospital accreditation, food chain safety, self-assessment and improvement of public offices, while assurance cases focused on trustworthiness of medical and lifestyle IT-based services, software security assessment and others. The argument sizes were different, up to about 2500 elements. The projects were carried out in the period of several years and the average project length was about one year. The NOR-STA software was improving and maturing in the process of these applications with the focus on optimizing the value added by the tool with a particular focus on large scale projects.

The objective of this paper is to review the challenges met by tools supporting management of evidence based arguments which we have identified during NOR-STA development and application. In the

following sections, we provide a brief description for each identified challenge and explain how NOR-STA responds to it.

2 Communication and teamwork

The challenge is to provide access to exact and up to date information and the related functionality for each participant of the process of developing and using an argument in accordance to the needs resulting from his/her role in the process and independent of his/her location.

While working on a project of the development and usage of an evidence based argument (for instance, a safety case) there is a need to share knowledge and the results of work with the teammates, consultants, customers and other experts. Delays in communication or using outdated information can lead to delayed or wrong decisions. The teams are often distributed and different team members play different roles (for instance, an external expert is invited to assess the argument).

2.1 Solution

- Make the tool a web-based application and deploy it to the users in accordance with the SaaS (Software-as-a-Service) model making it accessible by typical web browsers as shown in Fig. 1.
- Equip the data structures representing an argument with metadata to represent necessary attributes, change control related information, identification of data ownership and others.
- Provide mechanisms for implementation of Role Based Assess Control (RBAC) policies and assign user roles in accordance with their business needs.

2.2 *Effect*

The argument becomes a 'live' electronic document accessible from any place and at any time and can be directly and constantly used during the project. The team members can share access to the argument according to their roles and needs, and all see the same (current) version of the argument. External experts can be easily integrated to perform specific tasks.

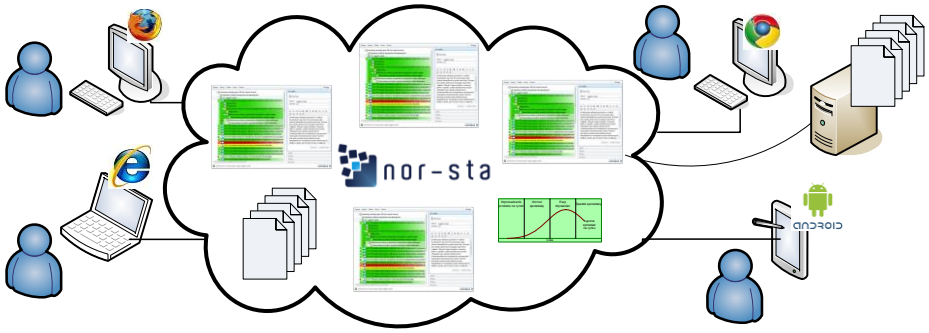

Figure 1. NOR-STA services offered as a cloud

3 Operating large arguments

The challenge is to provide for handling 'large' arguments, where the number of elements (claims, inferences, premises, evidence items) and their dependencies create the complexity exceeding the human cognition capacity.

In our practice we were dealing with arguments of up to 2500 elements. Such arguments are difficult to understand and the problem is even worse when you have to modify or update the argument. While working on an argument you often switch from a global view where you consider the overall structure of the argument to a local context where you work on a single argumentation step.

We all like diagrams and graphics but managing changes of a diagram can be quite difficult. Even adding a single element to an argument diagram may sometimes require its severe rearrangement. For instance, consider adding an additional argumentation strategy element to the argument shown in Fig. 2. Because of the limited page width, it would be almost impossible to place it at the same level as the elements S2 and _S5 despite that it belongs to the same level of logical hierarchy. And imagine such graphical representation of an argument diagram containing hundreds of elements. The structure of the graphical representation would then highly depart from the logical structure of the argumentation. The problems with visualizing arguments using notations based on graphical shapes were reported e.g. in [7, 8].

In addition, including the description inside the contour of an element leads to the situation where elements with longer descriptions are represented as bigger in comparison with the ones with more concise descriptions, which can create a false belief of differences in their meaning (for instance, consider the elements C5, C13 and C15 in Fig. 2).

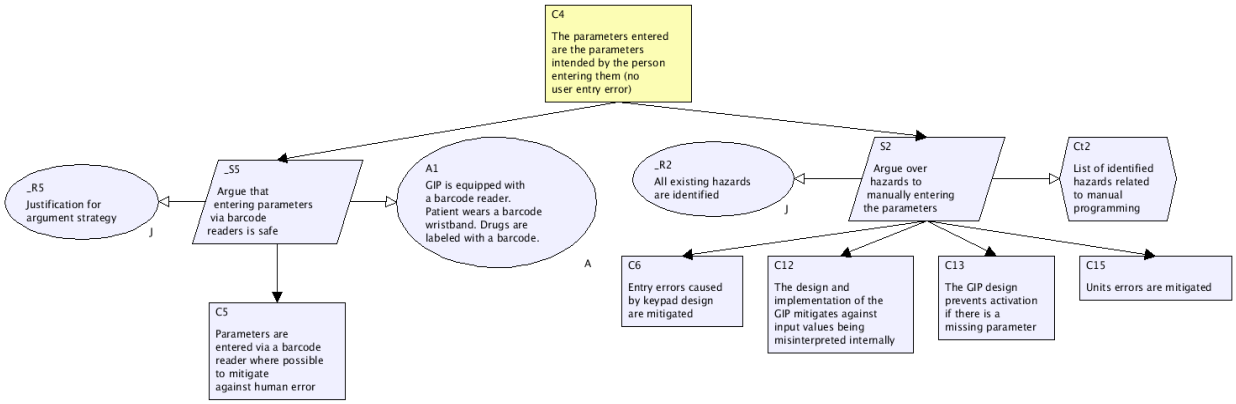

Figure 2. An example graphical representation of an argument

3.1 *Solution*

- Use the representation which is highly consistent with the logical hierarchy of argumentation even if the size of the argument grows significantly. Left-to-right representation of the hierarchy (compare Fig. 3) instead of top-down facilitates representing and traversing large hierarchies. It is particularly useful if the hierarchies are wide and with a relatively shallow depth (which can be often noticed for safety and assurance cases). This advantage of left-to-right representation is also exploited in many systems while representing file directories.
- Graphical symbols separated from textual descriptions of the elements - the texts explaining the meaning of argument elements are separated from the graphical representation of these elements (icons). As a result, the size of graphical representation does not depend on the length of the explanatory text related to the argumentation elements. Combined with left-to-right representation of the argumentation hierarchy, it provides for a structure that is compact and easy to read and traverse.

3.2 *Effect*

A compact and uniform (concerning the size of elements) representation of an argument where deeper layers of the hierarchy can be easily traversed; the argument can be reviewed by scrolling the display screen; the elements can be added/removed and their descriptions can be set to any length without affecting the argument visualization and without forcing the user to rearrange the structure.

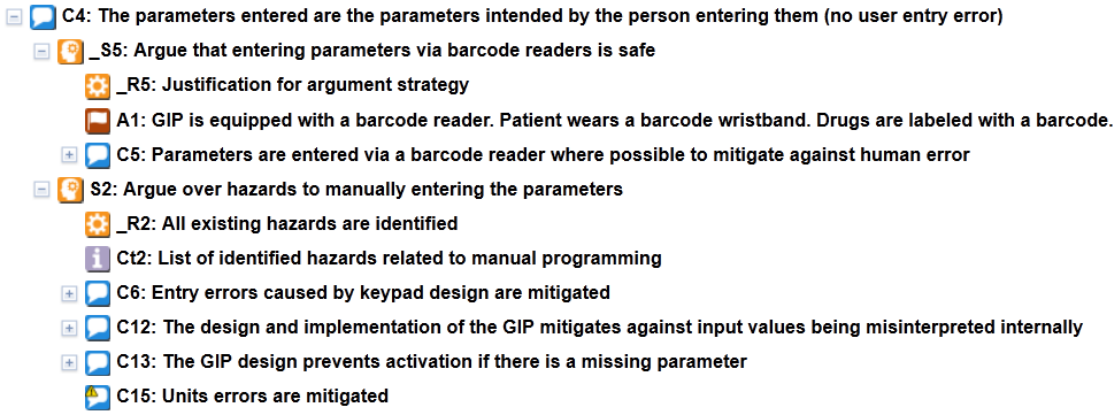

Figure 3. The argument from Fig. 2 expressed in NOR-STA

4 Managing the evidence

The challenge is to provide for argument integration with documents of different formats stored in different and often distributed repositories with a possibility of explicit pointing to the relevant section of the document (where the evidence is being located) and to enable attaching metadata describing the evidence.

The evidence 'grounds' an argument in the real world. The evidence can be produced during system life cycle activities and stored in dedicated repositories but it can also be collected from the external sources. The evidence may take a form of electronic documents of different formats. Typically, an evidence document is referenced just once but some documents can be referenced multiple times, in particular if different parts of the document (pages, sections, etc.) contain the material supporting different parts of the argument.

It is useful for an evidence document to be associated with metadata which characterize its contents, its status and role in the argument.

4.1 *Solution*

- Clear separation, at the syntactic level, of the argumentation and the evidence supporting the argumentation. In a given inference step, a premise can be another claim, an assumption or a fact (an assertion) to be demonstrated by the evidence. The evidence is being connected to the facts by explicit referencing. This clarifies the role the evidence plays in the argument (demonstrating a fact) and makes it visible in argument representation.
- Providing for referencing any electronic document at any place in the cyberspace as an evidence supporting argumentation. Consequently, any resource that pointed to by URL can be used as an evidence, for instance a document file, a web page, a dynamically generated report, even a live video stream.

- Providing dedicated and security protected evidence repository space - for users who do not want to involve themselves in management of the evidence repositories and at the same time allowing users to make references to their own repositories.
- Supporting access to selected repository types (for instance, SVN, FTP) to provide for a quick access to the evidence while argument browsing.
- Managing access rights to the evidence (CRUD).
- Providing a method for pointing to selected parts of an evidence document (for instance pages, sections, chapters).
- Establishing a set of metadata for describing evidence location, access rules, description, validity and other information useful for evidence management.

4.2 *Effect*

The evidence referred to from an argument can be stored in an arbitrary place and in an arbitrary format. Different evidence pieces can be stored in the same document, still with the possibility to be referred to individually. The access controls applied in the evidence repositories are being respected while accessing the evidence which provides for implementation of different access policies. Users who do not want to bother with managing evidence repositories can delegate this responsibility to the tool.

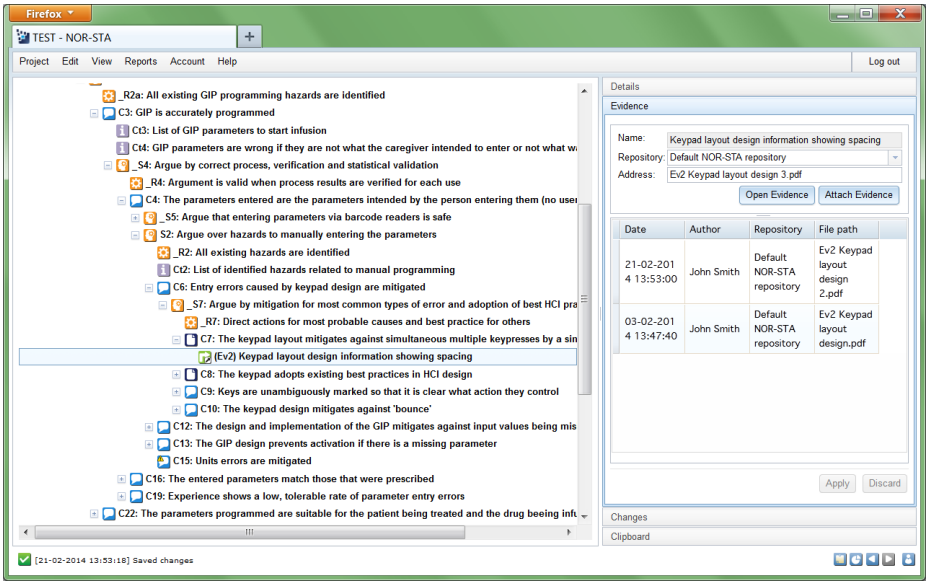

Figure 4. An example referencing to an evidence item (Ev2) and to its previous versions

5 Argument assessment and visualization

The challenge is to provide support for argument assessment (enabling the involvement of multiple assessors) with a possibility of applying different assessment models and to provide means for effective and efficient visualization of the assessment results.

Assessing the 'strength' of an argument needs human (expert) intervention. Depending on the application, it may also involve different methods of argument assessment. For instance, in our practice with conformance cases, we have learned that each of four standards we were coping with, had a specific conformance assessment method.

In relation to 'large' arguments (arguments of hundreds or even thousands of elements) we have identified an additional challenge: how and to which extent is it possible to automate the argument

assessment? And in particular how to provide for automatic aggregation of the local assessments (assessments of individual inferences and assessments of the support given by evidence to local claims) up the argumentation hierarchy? Such aggregation mechanism would help to solve the argument assessment scalability problem, having a potential for delivering more repeatable and more objective assessments results. It is also desirable that multiple assessors can issue their assessments which can then be visible to others and/or aggregated, if necessary.

Another challenge is how to represent argument assessment results to make them easily readable without the need to go to the technical details of the argumentation. This is particularly important while communicating the argument 'strength' up the decision-making hierarchies (in most cases, a decision maker will not be interested to go through the technical details; her/his main concern would be to learn if the argument is sufficiently 'strong' and where are its eventual weak parts).

5.1 *Solution*

- Visualization of argument assessment - the results of argument assessment are visualized by coloring the argument. This conveys an intuitive message about the argument 'strength' and provides for quick and easy identification of strong and weak parts of the argumentation without going into the technical details.
- Using diverse means of communicating the assessment results, in addition to argument structure coloring (for instance numbers, pie charts and so on) if more exact data is needed.
- Integration of different assessment mechanisms. Recognizing that there is not a single and universally accepted method of argument assessment, a mechanism of integrating customized assessment modules is needed (presently in NOR-STA five such modules are available).
- Supporting multiple independent assessments and their aggregation.
- Automatic aggregation of local argument assessment - one of the argument assessment methods presently implemented is based on the Dempster-Shafer theory and provides for automatic aggregation of local assessments issued by an expert [9]. This distinguishing feature provides for assessment of large arguments which would be difficult to perform manually (if possible at all).
- Inclusion of assessor's comments explaining her/his assessment decisions, which is especially important for work-in-progress assessment.
- Optionally, excluding an element from the assessment scope, if the element does not fit into the current context.

5.2 *Effect*

Argument coloring scheme (based on green, red, yellow and combinations of them) conveys the assessment results in an intuitive way (for instance, red = not accepted, green = accepted, yellow = uncertain) and indicates the weak and strong parts of the argumentation. Specific argument assessment methods can be switched on and off according to the needs. Explanatory information can be attached to provide for better understanding the assessor's intentions. Assessment of large arguments is greatly facilitated if the method based on Dempster-Shafer belief functions is applied.

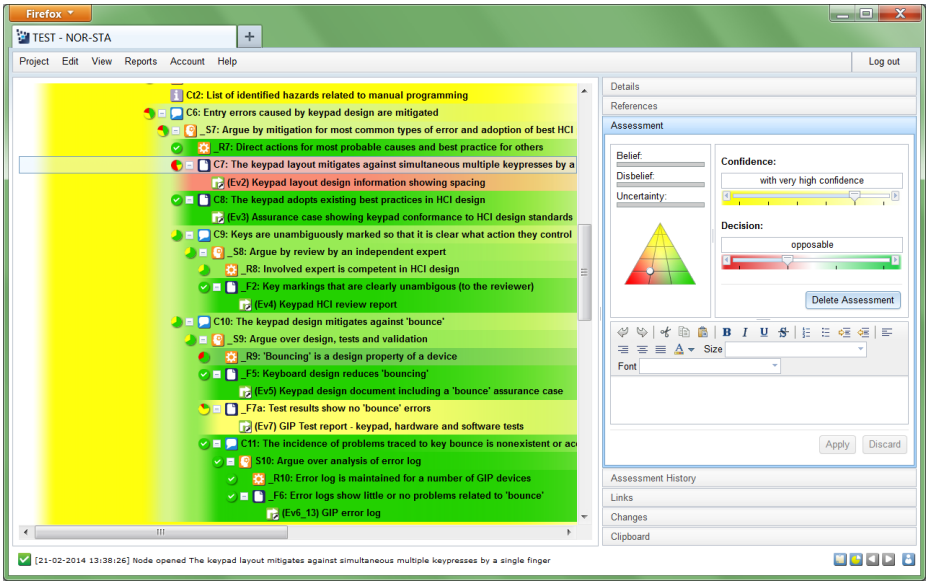

Figure 5. Argument assessment result (details of claim C7 are shown in the panel on the right)

6 Change control and state reporting

The challenge is to support the evolution of the argument structure and of the supporting evidence, to provide for forming baselines (checkpoints) that can be used for comparisons and possible rollback, and to generate reports of the argument state.

The arguments like safety cases, assurance cases and conformance cases have long lifecycles and they undergo many changes. For instance, it is recommended that a safety case is started at the concept stage of the system, evolves together with the system through the subsequent stages of its development (requirements, design, implementation, validation and so on) and then is maintained during system deployment and operation. During its lifecycle, an argument is often updated and re-evaluated. For instance, the conformance cases of hospital accreditation are updated at least each third year, before the next formal accreditation renewal. Meantime, they can be used to monitor the fulfillment of accreditation criteria and for periodic self-assessments. The scope of changes covers the argumentation structure, the evidence and the argument assessment. It is also expected that some states of the argument (for instance just after a formal accreditation) serve as a benchmark and the baseline for future evolution. The state of an argument needs to be reflected in various reports (including hard copies) to provide for argument status monitoring, audits and for archiving purposes.

The special case of a report is the export of the full argument contents. It can be needed for archiving purposes and sometimes to process the argument by other tools. Argument development and use should not be constrained by just a single tool and it is likely that the results are to be exported/imported to/from other tool environments. This need has been recognized by the recent work of OMG on the Argument Metamodel (ARM) [10] and some recommendations of ISO 15026 standard [4] are further steps in this direction.

6.1 *Solution*

- Maintaining the history of the argument, the evidence and the assessment results and providing for reviewing the historical values.
- Providing for change reporting, for instance the list of changes since a particular date.

- Providing for generating the ‘end of the day’ snapshot with the possibility to copy it as a starting point for an alternative argument.
- Report generation – in addition to maintaining an argument as an electronic on-line document which can be shared, edited, viewed and communicated according to the needs, a wide range of report generating functions is needed (covering different formats, like XML, doc, PDF and others).
- Web service interfaces are used to integrate with other systems for import (for instance, the evidence status and history) and export (for instance, arguments or assessments results). Standard XML formats for importing and exporting evidence and arguments are applied (like ARM and SACM).

6.2 *Effect*

The argument is accessible along the full line of its evolution with the user-defined milestones. A range of reporting functions address different needs of development monitoring and data archiving. An example Change List report is shown in Fig. 6. Migration to/from other tools is also possible.

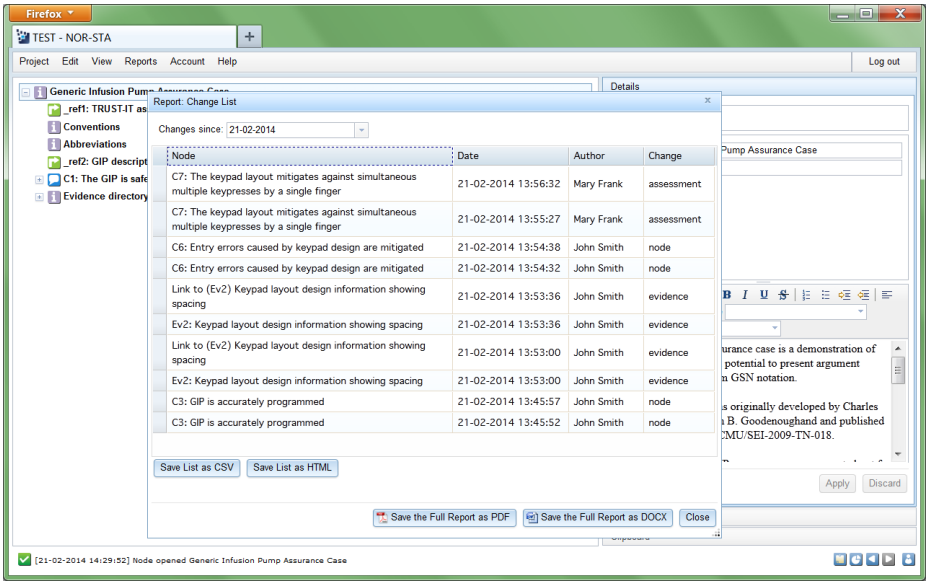

Figure 6. An example change report since a specified date

7 Argument reuse

The challenge is to improve argument development process efficiency and quality by re-using argumentation patterns and other assets proven to be effective.

The concept of reuse is widely accepted as a powerful means to save time and effort. With respect to the evidence based arguments it covers two aspects: reusing the argument (sub)structures and reusing the evidence. Both aspects are of great practical importance. For instance, reusing an argument structure saves the effort needed to ‘reinvent’ this structure and reusing the evidence provides for avoiding unnecessary redundancy and reduces the effort needed to maintain integrity of redundant objects.

7.1 *Solution*

- Conformance templates – while dealing with the conformance to standards we have successfully used the concept of a conformance template [11]. Such template is a skeleton of a conformance

argumentation which is derived from text of the standard. Then, the template can be reused by any organization willing to demonstrate the conformance which considerably saves time and effort. The tool provides for distinguishing the parts of the template which need further elaboration (in particular the expected evidence to be delivered) and for including additional information guiding a user.

- Argument patterns – argument patterns are argumentation schemes that can be reused as parts of many arguments. NOR-STA allows to store such patterns in a separate workspace (serving as a pattern catalogue) and to copy such schemes to the actual case under development.
- Reuse of argument fragments – the same structure can be repeated in several places of a given argument. NOR-STA provides for defining such a fragment once and then integrating it to different parts of the argument by explicit linking. The result is that that the argument becomes a DAG (Directed Acyclic Graph) instead of a tree which leads to more compact and easier to maintain argument structure.

7.2 *Effect*

Reusable argument assets can be defined and reused in different contexts by simple operations (copy-paste and linking). Additionally, while preparing such assets, additional explanations and hints how to reuse the asset in its target context can be provided by using special ‘information' elements which are not the part of the argumentation and are used to help the user and to make the argument more readable (an information element in an argument plays the same role as a comment in a software code).

8 Data security

The challenge is to provide means to meet security expectations related to both, argument structure and the evidence.

Users of the evidence based arguments have different expectations concerning protection of their data (argument structures, evidence, results of the assessment). These requirements can refer to the full CIA range (Confidentiality, Integrity, Availability). In our practice, we were facing users with low data protection concerns (publishing the argument in the Internet, integrating evidence from public websites), users having data protection among the highest priorities (the argument structure separated from the Internet, full control over evidence repositories with strict access control policies), and in the extreme case the users who in addition asked for formal security certification of our tool.

8.1 *Solution*

- As NOR-STA services can be used on-line, delivered from a cloud (in accordance with the SaaS model), the security of the entrusted data depends on the service provider. Presently such an operator is Argevide, a spin-off company of Gdańsk University of Technology [12]. In addition Argevide also offers a security protected space for evidence storage. In such case, security of the users’ data is based on the trust in the service provider (Argevide) and the trustworthiness is demonstrated by the Security Polices implemented by the provider. For the users who want to have more control over the evidence, it is possible to integrate the evidence documents from own repositories which remain under their exclusive security governance.
- Users with high security demands may use NOR-STA installed on a dedicated server in a protected local network. In such case they maintain full control over the installation and can implement their

own security policies (for instance, prohibiting access to Internet, installing specific security controls to access the argument, the evidence and so on).

8.2 *Effect*

The users can choose and adapt the form of NOR-STA services deployment to meet their security demands. In an extreme case the system can be installed on a standalone computer with no connection to the external world. It is also possible to install NOR-STA in the organisation's intranet with full control over the related security policies. In less demanding situations, a user can just access the NOR-STA cloud services with a browser on his/her personal computer connected to Internet. It is also possible to use evidence repositories which are fully controlled by the users' administrators.

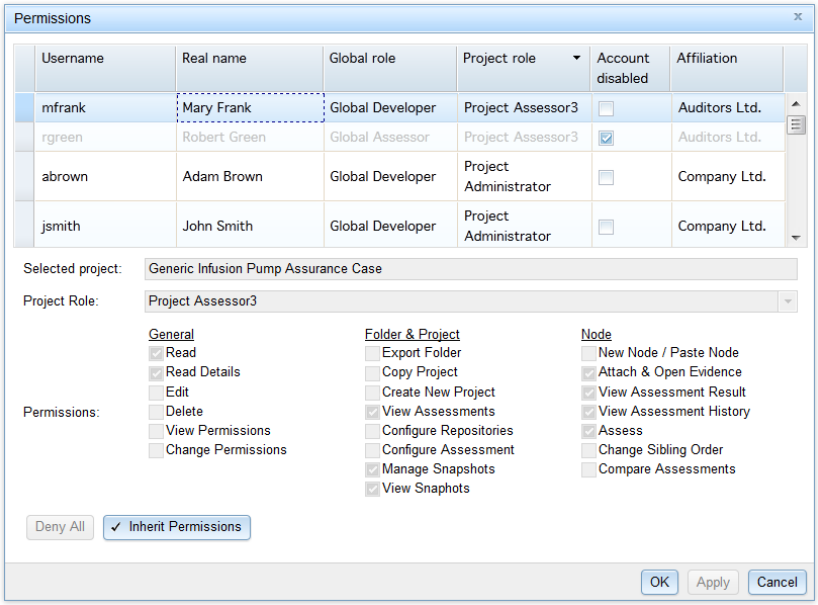

Figure 7. Assigning access rights to NOR-STA users

9 Argument portfolio management

The challenge is to support management of a set of arguments (argument portfolio) to provide for implementation of common policies with respect to these arguments.

In our practice we have faced different requirements concerning the management of the users of arguments and the management of the workspaces used to store the arguments (we call such workspaces 'projects'). These requirements ranged from a single project with just few users (in case of a small hospital aiming at accreditation) to hundreds or even thousands of users sharing access to tens or hundreds of projects (in case of a major consulting institution). The users work in different roles: administrators, editors, evidence managers, consultants, auditors and readers, which require different access permissions. With such a variety of requirements it is necessary that the tool provides for a high level of automation and a user friendly interface to manage the resulting complexity (user identities and access credentials, access roles, project access accountability, defining/opening/closing projects, and so on).

9.1 *Solution*

- One integrated environment for administering projects (workspaces for arguments and reuseable argument assets).
- Simple administration interface to administer users and user-to-project relationships. This interface is accessible at two levels: the platform administrator (allows to manage all projects and all users of a given NOR-STA installation) and package administrator (the administration function is restricted to projects and users within the scope of a given package purchased by the customer).
- Role based access control - NOR-STA implements the RBAC model for accessing projects and the associated evidence repositories. This helps in organizing the work on arguments in accordance to the business needs.
- Users and projects accounting – NOR-STA implements functions which collect data related to user-in-projects activity which provides for implementation of flexible billing schemes.

9.2 *Effect*

The administration functionality which is simple enough to be delegated to a user who either acquires a server license to install NOR-STA in his/her own infrastructure or prefers to have a package of projects and user accounts delivered by the NOR-STA cloud. This provides for creating a network of NOR-STA services operators who can deliver the services to their own customers. The accounting function provides for implementation of flexible billing schemes like pay-per-use, pay-per-project or pay-per-user.

10 Summary

Using the evidence based arguments in large scale calls for effective and efficient tool support capable to meet different technical and business needs. We have worked on methodological and technical problems of evidence based arguments for more than ten years in a series of R&D projects in international and local settings and with end users applying NOR-STA to support their business processes. In this process we were identifying different challenges related to the tool support for evidence based argument management which were then reflected in the NOR-STA tool during its development, following the evolutionary and incremental lifecycle model [13]. In this article we have enumerated some of these challenges which have particular relevance to cooperation, communication and sharing of large scale arguments in industrial settings.

The tool is undergoing further development, following the roadmap drawn from the requirements gathered from customer organizations and end users, as well as from the analysis of the needs of application domains (each year 2-3 major releases are deployed).

References

[1] ISO 26262-2:2011 Road vehicles – Functional safety – Part 2: Management of functional safety, International Organization for Standardization (2011)

[2] IEC 62278 Railway applications – Specification and demonstration of reliability, availability, maintainability and safety (RAMS), International Electrotechnical Commission (2002)

[3] Guidance for Industry and FDA Staff: Total Product Life Cycle: Infusion Pump - Premarket Notification [510(k)] Submissions, Food and Drug Administration (2010)

[4] ISO/IEC 15026-2:2011 Systems and software engineering – Systems and software assurance – Part 2: Assurance case, International Organization for Standardization (2011)

[5] Górski J., Jarzębowicz A., Miler J.: Validation of services supporting healthcare standards conformance, Metrology and Measurements Systems XIX (2), 269-282 (2012)

[6] Górski J., Jarzębowicz A., Miler J.: Comparative conformance cases for monitoring multiple implementations of critical requirements, In: Proc. of The 32nd International Conference on Computer Safety, Reliability and Security (Safecomp 2013), LNCS Vol. 8153, pp. 8-20 (2013).

[7] Weinstock, C. B., Goodenough J.B.: Towards an Assurance Case Practice for Medical Devices, Technical Note CMU/SEI-2009-TN-018, Software Engineering Institute, (2009).

[8] Holloway C. M.: Safety Case Notations: Alternatives for the Non-Graphically Inclined? 3rd IET International Conference on System Safety, Birmingham, UK, October 2008

[9] Cyra L., Górski J.: Support for Argument Structures Review and Assessment, Reliability Engineering and System Safety, Elsevier, 96, pp.26-37 (2011).

[10] Object Management Group, Argumentation Metamodel (ARM), 1.0 Beta (2010).

[11] Cyra L., Górski J.: SCF - a Framework Supporting Achieving and Assessing Conformity with Standards, Computer Standards & Interfaces, Elsevier, 33, 2011, pp. 80-95

[12] Argevide website, www.argevide.com

[13] Górski J. Witkowicz M.: Experience with instantiating an automated testing process in the context of incremental and evolutionary software development, e-Informatica Software Engineering Journal, Volume 5, Issue 1, pp. 51–63 (2011)

You can find more information on website: www.argevide.com.