Software Engineering Institute

Towards an Assurance Case Practice for Medical Devices

Charles B. Weinstock
John B. Goodenough

October 2009

TECHNICAL NOTE
CMU/SEI-2009-TN-018

Research, Technology, and System Solutions Program

http://www.sei.cmu.edu

Carnegie Mellon

This report was prepared for the

SEI Administrative Agent
ESC/XPK
5 Eglin Street
Hanscom AFB, MA 01731-2100

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

This work is sponsored by the U.S. Department of Defense. The Software Engineering Institute is a federally funded research and development center sponsored by the U.S. Department of Defense.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

Table of Contents

List of Figures

Acknowledgments

The authors would like to acknowledge the assistance of the following, all of whom added greatly to our ability to carry out this work.

At the U.S. Food and Drug Administration

- Richard Chapman
- Brian Fitzgerald
- Raoul Jetley
- Paul Jones

At Medtronic

- Sherman Eagles
- Patty Krantz

At the University of Pennsylvania

- David Arney
- Insup Lee
- Oleg Sokolsky

At Children's Hospital of Pittsburgh

- Donna Flook
- Andrew Urbach

Abstract

Software technology enables an increasing percentage of medical device functionality, leading to much more complex systems and presenting a challenge to regulators charged with evaluating device safety and effectiveness. An approach to evaluating claims of safety increasingly used in Europe and elsewhere is the safety assurance case. Much like a legal case, the assurance case lays out an argument and supporting evidence to show that safety claims are valid.

This technical note explores the use of assurance cases for justifying claims of medical device safety. It illustrates the use of the assurance case on a particular type of medical device—the infusion pump. This example serves as a basis for discussing issues surrounding the introduction of assurance cases into the medical device community, which includes both manufacturers and the U.S. Food and Drug Administration.

1 Introduction

The medical device industry finds itself moving inexorably in the direction of so many other industries—an ever-increasing percentage of device functionality is provided by software. The industry is beginning to experience the problems that arise when products that were formerly mostly hardware become significantly dependent on software for their safe and effective operation. The increasing complexity of medical device software raises new questions about how manufacturers and regulators are to gain confidence in the safe operation of such software-dominated devices.

The current practice for ensuring safety focuses on process evaluation—assessing manufacturer—compliance with safety regulations and standards. Some in the industry have recently shown an interest in having device assurance practices be more focused on demonstrating product-specific device safety rather than on gathering indirect process data showing that design and production practices are sound. Because assurance cases are product focused, the U.S. Food and Drug Administration (FDA) and some manufacturers are considering their use as a means of gaining more confidence in the safety of medical devices and in expediting the approval process.[1]

The Carnegie Mellon® Software Engineering Institute (SEI) began considering the assurance case as a method of software assurance in 2004. Since then, interest in the technique has become widespread. An international community, including the SEI, has been researching the issues involved with developing security assurance cases and has held several workshops on the subject. The Object Management Group (OMG) has established a working group in the area,[2] and the International Organization for Standardization (ISO) is considering a standard (15026) that includes assurance cases. The US Department of Homeland Security's *Build Security In* website contains discussions regarding the use of assurance cases.[3]

The SEI began talking with the FDA on the subject of assurance cases in 2005-2006. By late 2006, Advamed, an advocacy group for the medical device industry, became interested in the subject and invited us to talk to them in early 2007. Since then they have held workshops on the subject[4] and are talking about additional activities. Other workshops have been held by the University of Pennsylvania,[5] Massachusetts General Hospital,[6] the University of Minnesota,[7] and the FDA. Two independent projects have spun out of manufacturer and FDA interest in exploring product-focused assurance: a pacemaker "grand challenge" project,[8] based upon information provided by a device manufacturer, and a National Science Foundation (NSF)-sponsored project to specify and assure a generic infusion pump, being done by the University of Pennsylvania and the FDA

1 "Approval" has a specific formal meaning when used by the FDA. In this note, we only use this term in its common, informal sense.

® Carnegie Mellon is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

2 http://swa.omg.org

3 https://buildsecurityin.us-cert.gov/daisy/bsi/articles/knowledge/assurance.html

4 http://www.advamedmtli.org/go.cfm?do=Page.View&pid=131

5 http://www.iqpc.com/ShowEvent.aspx?id=79066

6 http://rtg.cis.upenn.edu/hcmdss07/index.php3

7 http://www.umsec.umn.edu/events/Summer-Software-Symposium-2008

8 http://sqrl.mcmaster.ca/pacemaker.htm

[Arney 2008]. The purpose of the pacemaker project is to challenge the assurance community to come up with a formally assured pacemaker design and breadboard implementation. The purpose of the generic infusion pump project is to develop a reference design for infusion pumps. This reference design will be used to research issues in verification and validation of embedded systems and will also be useful for manufacturers of infusion pump devices. We adopted this reference design as input to the work reported in this technical note.

This technical note presents issues concerning the use of assurance cases for medical devices, using a generic class of infusion pumps as a focal example. We provide an introduction to goal structured assurance cases in Section 2, where we introduce Goal Structuring Notation (GSN) and provide a brief tutorial on its use. Section 3 describes the generic infusion pump, which we use as an example throughout this report. In Section 4, we present examples of assurance cases dealing with infusion pump safety issues. In Section 5, we discuss factors to be considered in conducting a review of assurance cases. In Section 6, we discuss assurance case development issues and factors relevant to increased use of assurance cases in the medical device industry. In that section we talk about standards, current FDA approval practices, and possible changes that could result in assurance cases being used by device manufacturers. We summarize our results and discuss possible future work in Section 7.

2 An Introduction to the Goal Structured Assurance Case

An assurance case is somewhat similar in form and content to a legal case. In a legal case, there are two basic elements. The first is evidence, be it witnesses, fingerprints, DNA, and the like. The second is an argument given by the attorneys as to why the jury should believe that the evidence supports (or does not support) the claim that the defendant is guilty (or innocent). A jury presented with only an argument that the defendant is guilty, with no evidence that supported that argument, would certainly have reasonable doubts about the guilt of the defendant. A jury presented with evidence without an argument explaining why the evidence was relevant would have difficulty deciding how the evidence relates to the defendant.

The goal structured assurance case is similar. There is evidence (e.g., test results) that a property of interest (e.g., safety) holds. Without an argument as to why the test results support the claim of safety, an interested party could have difficulty seeing its relevance or sufficiency. With only a detailed argument depending upon test results to show that a system was safe, but in the absence of those test results, again it would be hard to establish the system's safety. A goal structured assurance case, then, specifies a claim regarding a property of interest, evidence that supports that claim, and a detailed argument explaining how the evidence supports the claim.

In our case, the top-level claim is "The Generic Infusion Pump (GIP) is safe." From that claim flows an argument that supports the top-level claim. The argument consists of one or more subsidiary claims that, taken together, make the top-level claim believable. These lower level claims are themselves supported by additional claims until finally a sub-claim is to be believed because evidence exists that clearly shows the sub-claim to be true.

To develop the GIP assurance case and make it reviewable by others, we adopted the Goal Structuring Notation (GSN) developed by Tim Kelly and his colleagues at the University of York in the United Kingdom [Kelly 1998]. This notation has been used successfully in many safety cases and is ideally suited for our work. Figure 1 shows a short assurance case developed in GSN. In it the top-level claim is labeled "Claim," the argument consists of the sub-claims "Claim 1" (that is supported by some evidence) and "Claim 2." Claims are phrased as predicates; they are either true or false. Evidence nodes are stated as noun phrases. Other elements shown in the sample are

- the diamond under "Claim 2," which indicates that the claim requires further development
- the triangle under "Evidence 3," which indicates that the evidence is parameterized and needs to be instantiated in an actual case
- the diamond within the link between "Claim 1" and "Evidence 1" and "Evidence 2," which indicates (as labeled) that either "Evidence 1" or "Evidence 2" applies or both do
- the parallelogram labeled "Strategy," which is meant to be a guide to the reader as to how the argument is structured
- a rounded rectangle labeled "Context"
- an oval with an "A" under it labeled "Assumption"

The last two elements provide explanatory information about the claim to which they are attached.

These are not the only elements to a goal structured assurance case, but they represent those used in the GIP assurance case that we discuss later in this note.

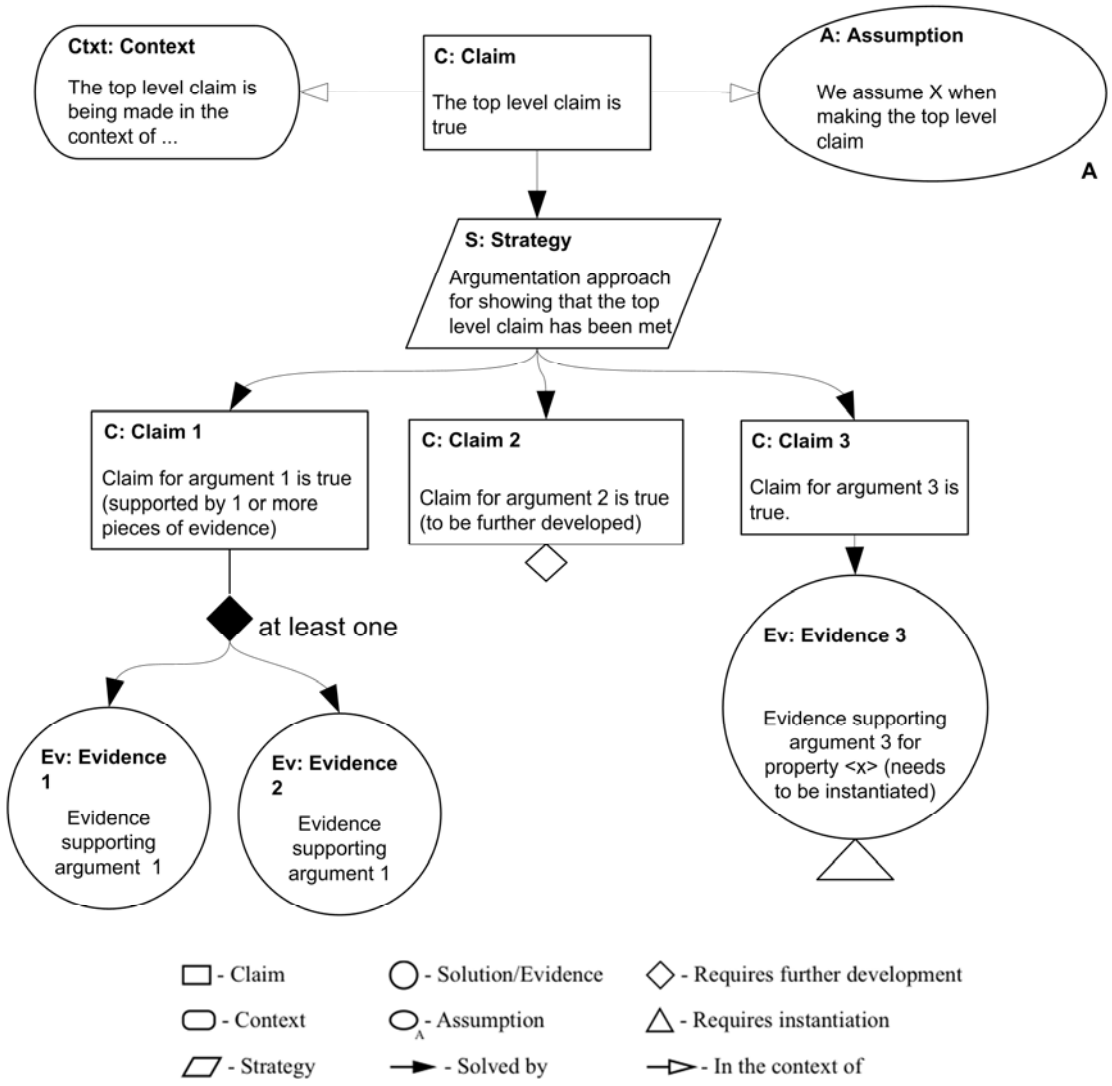

Figure 1: Example GSN Argument

3 The Generic Infusion Pump

An infusion pump delivers fluids into a patient's circulatory system with the intention of providing continuous or periodic controlled delivery of medicine. Pumps can be relatively simple (e.g., infuse at a constant rate until the fluid is used up) or extremely complicated (e.g., infuse at varied rates, from varied sources, over a period of time). Regardless of complexity, all pumps require input programming to control the rate and duration of infusion. Once these parameters are set, the pump should correctly deliver the specified dosage through the tubes connecting it to the patient. Figure 2 is a picture of a complex infusion pump with the keypad and interface used for programming.

Figure 2: An Advanced Infusion Pump

Ensuring the correct delivery of the prescribed drug to the patient is critical, so the pump manufacturers strive to design the pump interface to minimize mistakes made in setting medication parameters. Current generation pumps include drug libraries (established by the health care provider) that contain recommended and absolute minimum and maximum doses for a wide range of drug uses. For example, the patient's weight is a typical parameter that (some) drug libraries take into consideration because it can have an effect on the appropriate dosage for some medications. Another determinate of proper dosage is the pump's clinical application. The dosages allowed in pediatric applications differ significantly from the dosages allowed in adult or geriatric applications. A (smart) infusion pump will check input parameters against the drug library and trigger a warning if any are outside the recommended dose range.

Correct parameter entry is so important that advanced pumps come with barcode readers to read drug information off the bag or cassette that the drug comes in to avoid data entry errors. Some pumps also read patient information from a bar-coded wristband.

Other factors can affect patient safety as well. For instance, tubes can and do become obstructed (occlusion), e.g., due to a kink in the drug delivery tubing. Or, air can get into a delivery line, causing a bubble that will result in serious injury or death to the patient. The pump must be able to detect these (and other) anomalous conditions and respond appropriately.

Some pumps, designed to deliver analgesic medications such as morphine to the patient, have a feature that allows the patient to request an additional infusion (bolus dose) of the drug by push-

ing a button. These pumps must ensure that the patient cannot receive an overdose if he or she pushes the button too frequently.

For this work, we initially planned to use a design for a GIP developed by Insup Lee and his team at the University of Pennsylvania with FDA assistance. We ended up using that design (and its associated requirements and hazard analysis) as inspiration for the (imaginary) pump we used in this case. We're using this pump rather than a real infusion pump to avoid proprietary issues that might restrict the usefulness and distribution of our results. We've assumed that the pump we are assuring includes a drug library and readers that obtain drug and patient information from barcoded labels.

4 Creating the GIP Assurance Case

There are basically two approaches for structuring a safety assurance case: (1) focus on identifying safety requirements, showing that they are satisfied, or (2) focus on safety hazards, showing that they have been eliminated or adequately mitigated. The approaches are not mutually exclusive—to show that a safety requirement is met one often has to show that hazards defeating the requirement have been eliminated or mitigated—but each has a different flavor. Each type has a role to play in developing an assurance case.

4.1 Structuring an assurance case

Because regulators and manufacturers are used to stating requirements and then ensuring that they are satisfied, top-level claims in an assurance case often have a requirements flavor (e.g., "The GIP is safe," which might be decomposed into subclaims that the GIP is electrically safe, clinically safe, etc.). Typically, safety requirements arise from an understanding of hazards that need to be addressed; each safety requirement, if satisfied, mitigates one or more hazards. But if the case just addresses safety *requirements*, the link to the hazards mitigated by the requirement can be lost; it can become difficult to decide if the requirement is adequate to address the underlying hazard(s).

To see how a focus on requirements can obscure underlying hazards, let's consider an example. Infusion pumps typically contain batteries so a patient can walk to the bathroom or around the hospital floor. An obvious hazard is loss of battery power; one might therefore state a safety requirement aimed at helping to ensure that the pump is plugged into an electrical power source prior to battery exhaustion. Such a requirement might be worded as follows:

> *When operating on battery power, visual and auditory alarms are launched at least 10 minutes prior to battery exhaustion but no more than 15 minutes prior.*

To demonstrate that this claim holds for a particular infusion pump, we could provide test results showing that warnings are raised at least ten minutes prior to battery exhaustion but no more than fifteen minutes prior. In addition, we could present arguments showing that we have confidence in such test results because the structure of the tests has taken into account various potential causes of misleading results. For example, since the battery discharge profile changes depending on the age of a battery, we would need to show that all the tests were run with a mixture of new and well-used batteries. Similarly, since the electrical load might affect the time to battery exhaustion, we would need to show that the tests were run with different electrical loads on the pump.

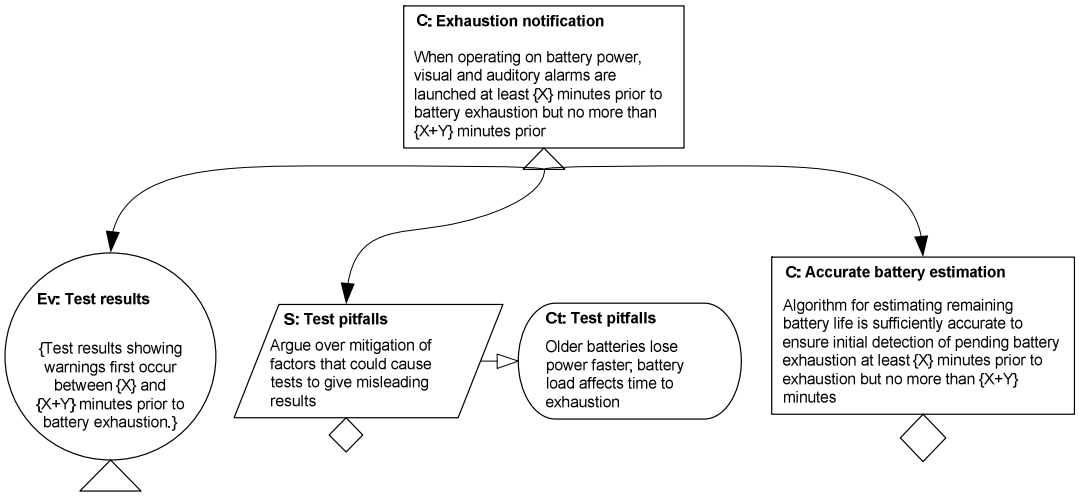

Figure 3: Confirming that a Safety Requirement has been Satisfied

We can represent the safety requirement as a claim in an assurance case together with the evidence and other arguments needed to argue that the requirement is satisfied (see Figure 3). In this figure, the top-level claim is the safety requirement, parameterized to allow for different warning margins. (Parameterization is indicated by the triangle at the bottom of the claim rectangle and by the use of {} to show the parameters that need to be instantiated when this fragment is used for an actual pump.) The test results and testing pitfalls are also shown in the diagram, although we have not expanded on the evidence or other argumentation needed to show that the testing pitfalls have been adequately mitigated. In addition, to increase confidence in the validity of the main claim, we complement the test results and mitigation of test pitfalls with a claim asserting the accuracy of the algorithm used to estimate remaining battery life. The combination of testing results and algorithm analysis makes the case stronger than if just test results alone are presented. To support the algorithm-accuracy claim, a manufacturer might reference design studies and literature as well as an analysis of the actual design.

Such tests and analysis are fine for demonstrating that the requirement is satisfied. But from a safety viewpoint, we have little documentation about what hazard the requirement is mitigating. In addition, how do we know that 10 minutes is the appropriate warning interval for every clinical setting? Is 10 minutes enough time for someone to respond to the alarm? Will the alarm be heard in every possible setting? How accurate does the measure of remaining power need to be (e.g., is it unacceptable if the alarms are launched when 20 minutes of power remains)? How does this requirement fit with other safety requirements? In short, to fully understand and validate the requirement, we need to establish the larger context within which the requirement exists. Figure 4 gives a possible context within which this requirement could sit.

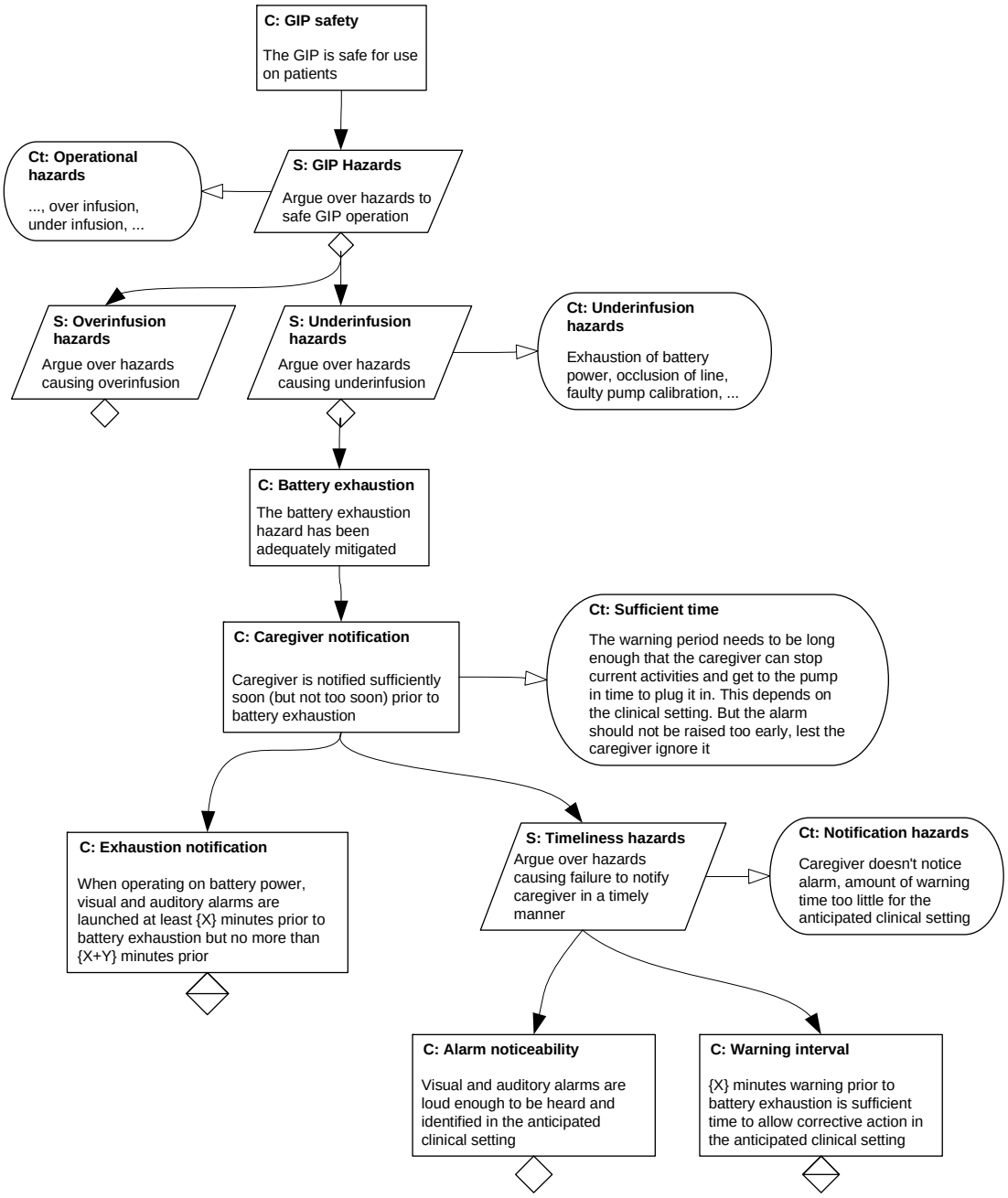

Figure 4: Context for Raising an Alarm about Impending Battery Exhaustion

Figure 4 starts with the relevant top-level claim: "The GIP is safe for use on patients." The case continues by stating the overall argumentation approach being taken, namely, arguing over the hazards that would impair safety if they were not mitigated or eliminated. This approach is stated in the strategy element: "S: GIP Hazards." Among the operational hazards listed in the context element "Ct: Operational hazards" are overinfusion and underinfusion. We propose to argue these hazards separately, as shown by the separate strategy elements for overinfusion and underinfusion. Since loss of battery power would lead to pump shut-down, this is a hazard that falls in the

class of underinfusion hazards. We state a claim that this hazard has been adequately mitigated ("C: Battery exhaustion").[9]

The proposed mitigation for battery exhaustion is to notify a caregiver in a timely manner that the battery is about to shut down. This is shown in the case by making the claim "C: Caregiver notification." The claim talks of notifying the caregiver "sufficiently soon," but not "too soon." The associated context element ("Ct: Sufficient time") explains the constraints on what constitutes "sufficiently soon" and "too soon." We are now in a position to state the safety requirement about when an alarm needs to be raised, "C: Exhaustion notification"; we understand the context within which it sits. In addition, we can now readily deal with other hazards not addressed directly by the safety requirement; namely, we can consider whether the warning time is sufficient to allow caregiver response and also whether the alarm is sufficiently noticeable that the caregiver will be unlikely to ignore it.[10]

Taken altogether, the exhaustion mitigation and notification claims establish the context and validity of what was originally an isolated safety requirement. Whether all this argumentation is necessary depends on the importance of dealing with battery exhaustion and the extent to which the industry has a standard way of dealing with it. Less argument (and evidence) is needed to support mitigations of less important hazards or where there is industry and regulatory consensus on adequate ways of addressing a particular hazard.

A benefit of focusing on safety requirements is that stating the safety requirements and demonstrating that they have been met seems straightforward from a user/regulatory viewpoint. But a safety assurance case that just addresses whether safety requirements are met will focus primarily on what tests and test conditions or other analyses are considered sufficient to show the requirements are met. The case is likely to be less convincing when considering whether all relevant hazards have been accounted for, because the reasoning leading from the hazards to the requirement is not necessarily part of the case.

Another problem with a pure requirements-based approach is that it can be difficult to specify fault-tolerant behavior. For example, consider a high-level requirement such as "The GIP delivers the prescribed amount of the prescribed product." Satisfying this requirement would certainly seem to satisfy a higher level claim that the GIP is safe. But the requirement, as stated, implies that the GIP *always* delivers the right amount of the right product, and clearly, there are factors outside the GIP's control that can prevent this from happening (e.g., the GIP has no way of recovering from a user entering an unprescribed delivery rate when programming the GIP). Similarly, if an infusion line is occluded, the GIP has no way of clearing the line. From a safety viewpoint, we want to ensure the GIP minimizes the chances of harming the patient. Stating a claim that is

---

[9] There are other underinfusion hazards; the diagram indicates that these are to be developed later by putting a diamond on the "Underinfusion hazards" strategy element.

[10] The case supporting the "Alarm noticeability" claim could be fairly complex, since the total variety of alarms and indicators needs to be considered, as well as the fact that some alarms are more important than others. One could ask: Is the device safer if the auditory alarm is louder or more urgent when the remaining battery life is less than five minutes? Less than three? And what happens when there are competing alarms? Which one gets the highest alarm signal? Is the overall alarming strategy for the device consistent with user interface standards for alarm signaling? Will the alarm for an important condition be loud enough to be heard over competing alarms or the sounds of other equipment (e.g., in a room where there are many devices connected to a patient)? All these issues can be raised and dealt with in the expansion of the "Alarm noticeability" claim.

unachievable in the real world doesn't allow the case to adequately address safety hazards and their mitigations.

From a safety argument perspective, instead of focusing on safety requirements, *per se*, it is more convincing to state (and satisfy) hazard mitigation claims. For example, a claim such as "The possibility of delivering an incorrect dose has been mitigated" allows the assurance case to discuss the possible hazards to incorrect delivery and then to explain the mitigation approaches, which can include raising alarms to cause a human intervention.

4.2 The GIP Assurance Case and Discussion

Our original goal was to produce a complete assurance case for a GIP, but given our limited resources, we decided to limit our case to a key aspect of the GIP—its programming by the caregiver. This section of the report presents our assurance case for programming the GIP.

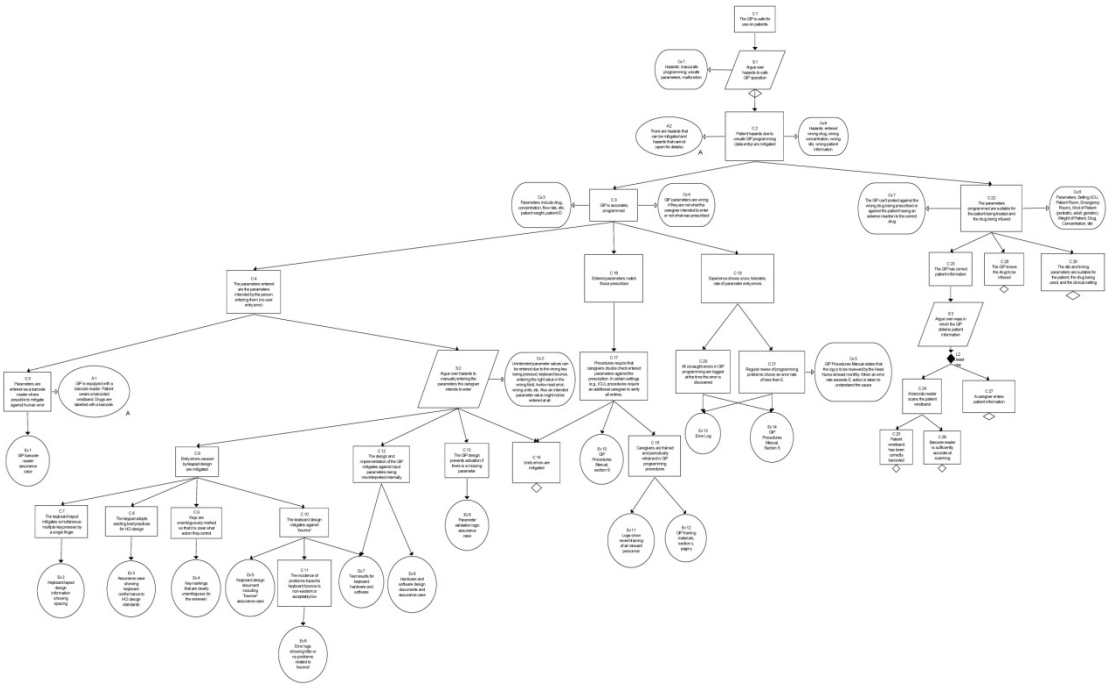

Figure 5: The GIP Assurance Case—Bird's Eye View

Figure 5 shows an overview of the GIP programming assurance case. It is not easily viewed on one sheet of paper, even a very large sheet of paper. To make the case more reviewable, we've broken it into smaller segments.

Figure 6 is the start of the assurance case. It presents the top-level claim that the GIP is safe for use on patients. The strategy element S1 shows that we've decided to argue over the possible hazards to patient safety. In this case, we've identified two possible hazard classes: (1) the GIP may be programmed inaccurately or unsafely, and (2) the GIP may malfunction.

To make the assurance case more readable, we will generally enumerate hazards in a context element when they are first identified. While we've only shown three hazards in Ct1, there could be others. For instance, if the GIP sits on a top-heavy stand, it could fall over and injure the patient. We've chosen to ignore other hazards since we're only dealing with unsafe programming in this assurance case.

At this point the argument divides into two. On the right in Figure 6 is a claim that "Patient hazards due to GIP malfunctions are mitigated" (C30). This argument needs further development, not shown here, as indicated by the diamond that is underneath it. The more interesting claim for our purposes is that "Patient hazards due to unsafe GIP programming (data entry) are mitigated" (C2). It is worth noting that there are some hazards to the programming of the GIP that cannot be mitigated—at least not with programming safeguards. For instance, if the physician prescribes *amoxicillin* when meaning to prescribe *acenocoumarol*, there is no way for the GIP to second-guess the prescription. This is captured in the assumption (A2).

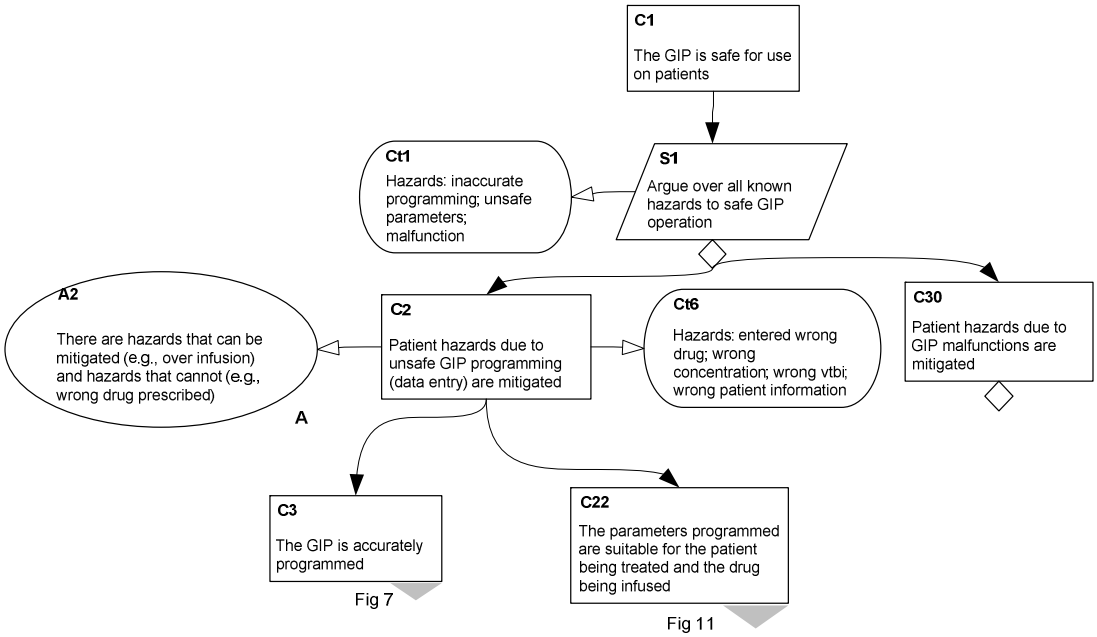

Figure 6: The GIP Assurance Case—The GIP is Safe

Although there are several hazards listed in the context element Ct6, they can be reduced to two classes of hazards and dealt with independently. The first (C3) is that the GIP is programmed accurately (for the right patient, with the right drug, etc.) and the second (C22) is that the parameters are suitable (i.e., safe) for the patient and drug. Each of these is dealt with separately and (in this report) in another figure, as indicated by the shaded triangle underneath the claim.

We'll begin with claim C3, "The GIP is accurately programmed," which is shown as Figure 7. This claim is supported by three claims: the parameters entered are those that were intended by the person entering them (C4), the entered parameters match those that were prescribed (C16), and experience shows that parameter entry errors are infrequent (C19). We can believe claim C2 (that hazards to GIP programming are mitigated) because the GIP has been designed to allow for accurate data entry as shown in Figure 9 on page 14, and we have historical evidence that shows a low and tolerable rate of programming error. This argument is also used as part of the argument for mitigating the hazard of entering data that differs from that which was prescribed.

The argument for the low, tolerable rate of programming error depends upon the first two pieces of evidence in the GIP programming assurance case, namely the log kept of GIP programming errors and the procedures manual for programming the GIP that calls for keeping and reviewing the error log. Note that the argument for the entered data matching what was prescribed (in Figure

8) also depends upon procedures—in this case a double check by another caregiver, when appropriate, as well as an ongoing training program to ensure that such procedures are followed.

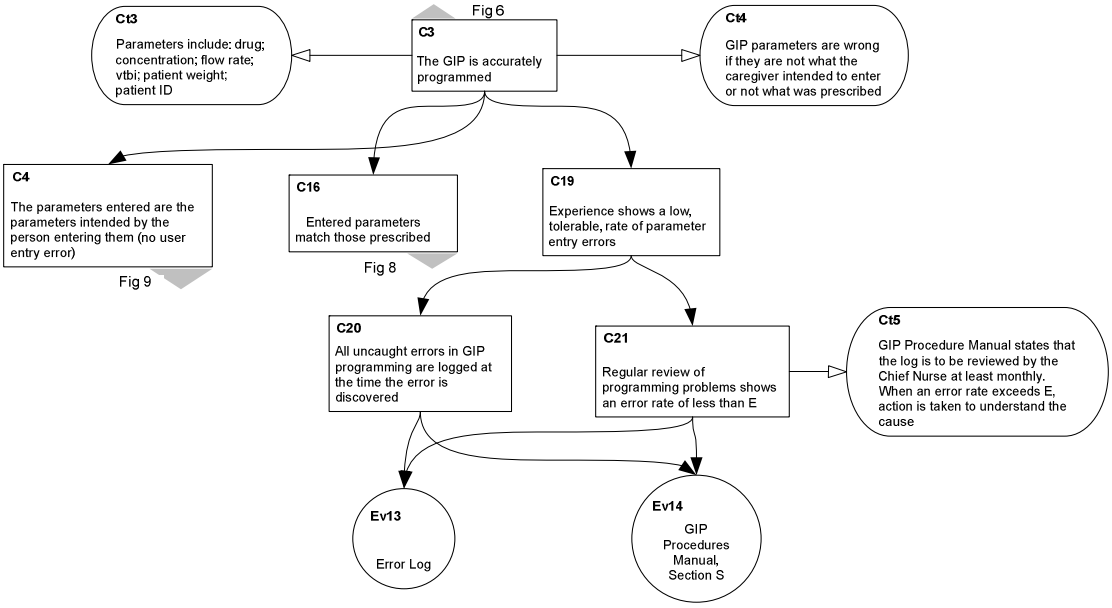

Figure 7: The GIP Assurance Case—Accurately Programmed

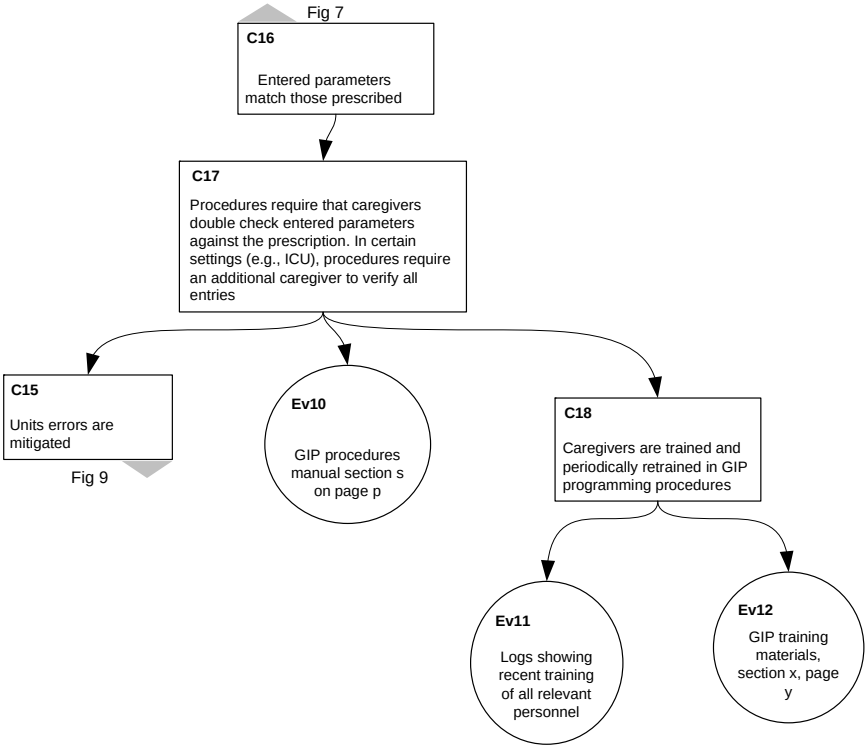

Figure 8: The GIP Assurance Case—Parameters Match

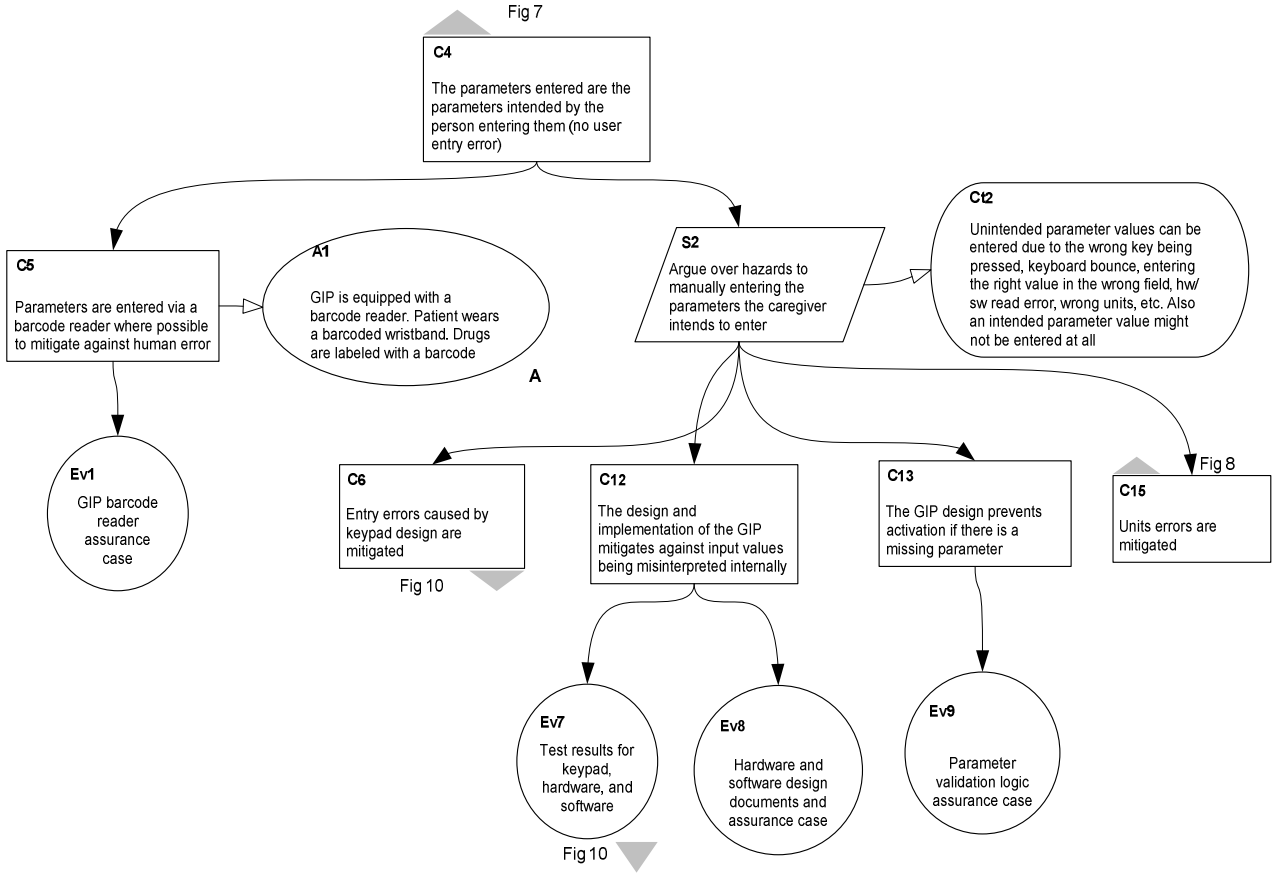

Figure 9: The GIP Assurance Case—Intended Data

Figure 9 expands the argument that the data actually entered into the GIP is what the caregiver intended to enter—that is, that there are no data entry errors (C4). Again there are two parts to the argument depending upon how data is entered. The preferred (these days) means of entering drug and patient information is to scan a barcode on the drug pouch or syringe or the patient's wrist-band, as shown in C5. This eliminates a source of human error but requires an assurance case for the barcode reader and analysis of how often the reader fails. The second part of the argument deals with additional hazards that must be dealt with when a human is entering the data (S2 and its supporting claims). These hazards (enumerated in Ct2) include keyboard issues, GIP read logic errors, the failure to input a value, and an error in entering the proper units (milliliters instead of micro liters for instance.)

Figure 10 shows how the keypad is designed to mitigate entry errors. The argument relies on the fact that the design mitigates against multiple keys being pressed by a single finger (C7), that the human-computer interface (HCI) is appropriately designed (C8), that the key labels (soft or hard) are unambiguous (C9), and that the keypad design avoids the problem of "bounce" (C11).

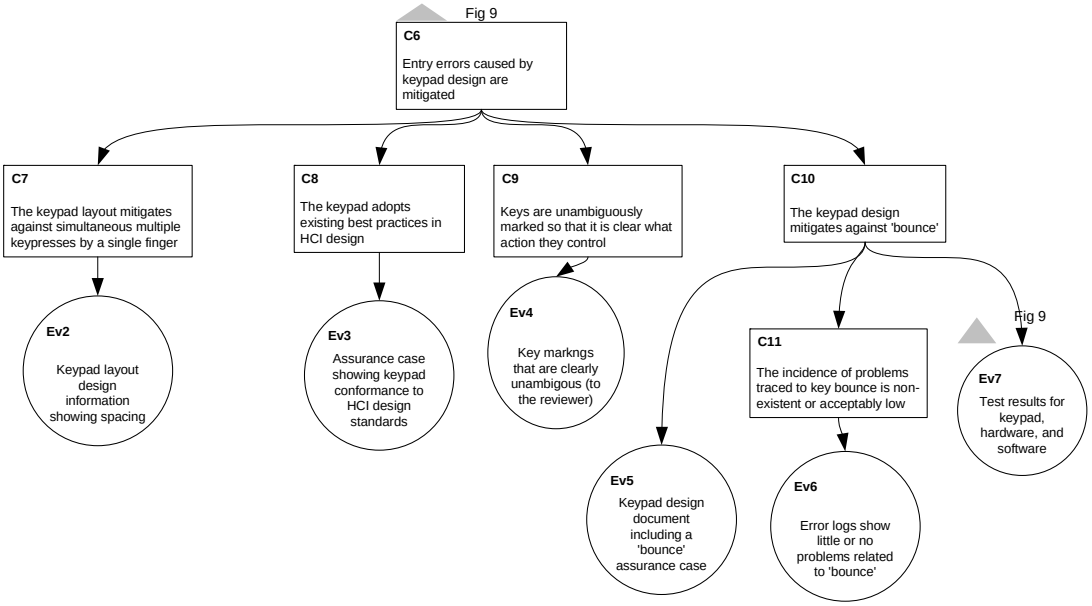

Figure 10: The GIP Assurance Case—Keypad Design

Having structured an argument about whether the GIP has been programmed accurately, we can refer back to the main claim in Figure 6 on page 12 (The GIP is safe for use on patients) to address the other leg of the argument—the parameters programmed are suitable for the patient and drug being used (C22). The argument for this is shown in Figure 11.

Since this claim is about the data being provided to the GIP, not programming the GIP, we have not fully fleshed out its argument. However, if we did it would be along the lines of stating that the GIP has correct information about the patient (possibly scanned off the wristband), that it has built-in information about the drug being infused, and that the parameters set for volume to be infused (vtbi) and timing are suitable for the patient/drug combination.

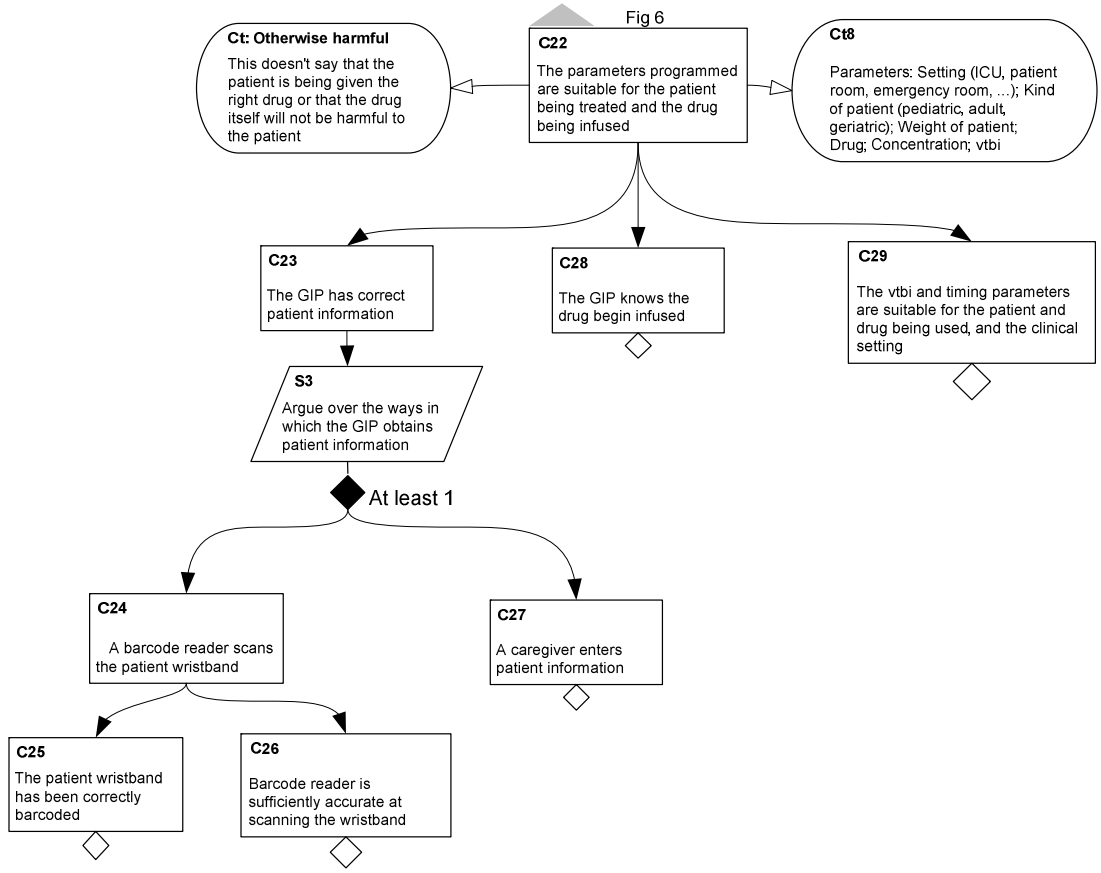

Figure 11: The GIP Assurance Case—Suitable Parameters

5 Reviewing Assurance Cases

Adopting assurance cases necessarily requires that there be a way to review them. This section discusses factors to be considered in conducting a review. A good discussion of reviewing assurance cases can be found in "Reviewing Assurance Arguments—A Step-by-Step Approach" [Kelly 2007].

In order to accept an assurance case, the reviewer must be able to understand it, be convinced that its arguments are sound and supported by the evidence, and believe that the case adequately addresses all relevant assurance issues.

It is up to the device manufacturer to ease the reviewer's task as much as possible. GSN helps in this regard, but only if it is used properly. Proper use of GSN requires that the case make liberal use of the annotations provided by the context, strategy, assumption, and justification nodes. It also requires that there be no major leaps from one step to the next as the reader is led through the case.

The first step in the review process involves checks to verify that the case is structurally complete and that node phrasing is correct. One check is to see that every node can be traced back to the top-level claim (that is, there are no "dangling" nodes or sets of nodes) and that each "leaf" node is either evidence or a reference to some previously reviewed assurance case. In addition, a second check is to ensure that claims are expressed as simple predicates and that evidence is a noun phrase (not stated as a claim). Checking that claims and evidence nodes are correctly phrased guards against confusion when later considering the substance of the case.

Once the structure and node phrasing of the case have been verified, the reviewer should consider whether the argument is persuasive. An argument is persuasive if each claim follows from the claims or evidence supporting it. Persuasiveness is best achieved when the associations between claims are obvious to a reasonably knowledgeable reviewer. An argument loses persuasiveness to the extent it requires the reviewer to spend time deciphering how strongly a claim is supported by its sub-claims or evidence.

The persuasiveness of a case can be weakened in various ways, including:

Incompleteness

A good assurance case cannot be selective in the arguments and evidence it presents.

- Incomplete argument: if an argument strategy is to break a claim into subcases and argue each of the subcases separately, the argument is defective if all the subcases are not actually addressed. For example, if a claim states that all hazards are addressed, the ensuing argument must actually address all identified hazards
- Incomplete claims: a claim is defective if it is stated too narrowly to be of interest (e.g., a system is shown to be safe only under a subset of conditions unlikely to be experienced in operational use)
- Incomplete evidence: evidence can be defective in various ways. A case is stronger to the extent that it argues against the presence of such defects (e.g., consider the discussion of usability evidence in Section 6.3).

Robustness Issues

- Interdependent arguments/evidence: if a claim is supported by independent arguments/evidence (e.g., by test results and by modeling analysis), the claim is more likely to hold, since a defect in one branch of the supporting argument will not impair the validity of the other branches. To the extent that proposed supporting arguments are not independent, the claim is more weakly supported than it might at first appear.

Fallacious Arguments

- Indirect effect: the connection between the supporting argument/evidence and a claim is only indirect (e.g., historical evidence that some hazard is rare or never happens is only indirect evidence that the hazard is adequately controlled "by the usual measures")
- Circular reasoning: occurs when an argument is structured so that it reasserts its claim as a premise or defines a key term in a way that makes the claim trivially true.

For a further discussion of fallacious arguments in assurance cases see "A Taxonomy of Fallacies in System Safety Arguments" [Greenwell 2006].

6 Towards an Assurance Case Practice for Medical Devices

The FDA's Center for Devices and Radiological Health (CDRH) is responsible for regulating firms that manufacture, repackage, re-label, and/or import medical devices sold in the United States. Medical devices generally require approval from the FDA prior to being marketed in the U.S., but the level of review required varies with the class of device.

Some software is more critical than other software. Software whose failure can lead to serious injury or death, for instance, is reviewed more stringently than other software. For the most critical software, the FDA looks at materials including:

- a description of the software: an overview of the features and the operating environment
- a device hazard analysis including both system and software hazards
- a complete set of requirements at the engineering level of detail. The requirements are expected to be unambiguous and realistic with respect to tolerance levels. They should include assumptions about the boundaries of use of the device.
- an architecture design chart
- the software design specification that shows how the requirements have been met and how the hazards identified in the hazard analysis have been mitigated
- a description of the software development environment including the critical issue of change management
- verification and validation documentation including test results at various levels (e.g., unit, integration, system level, etc.)
- revision history
- unresolved anomalies

This review can yield a huge amount of evidence, all of which must be considered within the required guidelines for timeliness.

6.1 Reviewing Critical Medical Device Software with Assurance Cases

The current approval process is time consuming, complex, and potentially inconsistent. The complexity of medical devices is growing and standards don't cover all of the relevant aspects. FDA reviewers (and the manufacturers themselves) have difficulty identifying all of the important technological risks. Since technological risk is an increasing portion of the overall risk of a medical device, the level of expertise required of FDA personnel and the effort they must expend to do their job is increasing.

A goal-structured assurance case holds promise to make the process less daunting by providing a means for the FDA examiner to understand just what beneficial properties the manufacturer is claiming for the device and how the evidence shows that the device is safe and effective. Instead of having to work through piles of evidence with little to no guidance, an assurance case provides the examiner with a structure that is easier to follow.

With an assurance case as a guide, the examiner can focus on how specific evidence supports various safety claims. Because the argument relating evidence to claims is explicit in an assurance case, it should be possible to evaluate a submission accompanied by an assurance case more quickly and accurately than one submitted in the current fashion.

There are major challenges to the adoption of assurance cases by the device industry and the FDA.

- Working with manufacturers, the FDA will have to define a goal-structured assurance case approval process. Issues in defining the process include determining
  - how much evidence is enough
  - how the evidence is to be used
  - who "owns" the evidence (given any trade secrets that it may contain)
  - how to physically submit both the evidence and the assurance case that relies on it[11]

  Arguments need to consider standards and regulatory guidance. The validity of the argument should not depend upon the experience of the reviewer.
- The FDA must provide a level playing field for device approval. If some manufacturers use assurance cases and others don't, the FDA must ensure fair evaluation in either case. In the words of one person at the FDA, optional adoption creates a "digital divide." To this we might add that forced adoption without advanced industry buy-in is likely to fail.

An approach to blazing a trail towards assurance case adoption is currently being discussed by the FDA, the medical device industry, and other interested parties. It involves developing specific regulatory guidance on assurance case submittal and use, together with a pilot process for early adopters. One possible way to ease into the use of assurance cases is to apply them initially to commercial off-the-shelf (COTS) components used in a wide variety of medical devices.

6.1.1 Easing the Review Process

As device manufacturers begin to develop assurance cases for their products, inevitably they will develop a library of domain-specific assurance case patterns (see Section 6.3). The use of these previously reviewed patterns can simplify the review process. The reviewer needs only to check that the pattern has been instantiated properly and that the evidence is appropriate, without delving into the intricacies of the actual argument being presented.

6.1.2 Reapproval

The use of assurance cases in the approval process provides an important benefit if there is ever a need to reapprove the device due to a design change or improvement. It is likely that only certain pieces of evidence and/or assumptions will be affected by the change or improvement. In this case only those portions of the assurance case affected by the new design choices will require review—a potentially important source of cost and time savings.

[11] Many of these issues exist under current practice. The introduction of the assurance case may simply require minor modifications to that practice.

6.2 The Value to the Device Manufacturer

Assurance cases are useful tools for the device manufacturer regardless of whether the FDA begins to use them in the approval process. Here are a few potential benefits:

- An assurance case fully documents the device being manufactured, leading to more confidence in the quality of the device and making it more likely that the design will be understood as new personnel are brought on board.
- Used properly, the assurance case is developed in parallel with development of the medical device. As a result, the manufacturer has more insight into product quality earlier in the development cycle and can take less expensive corrective action if problems begin to surface.
- The opportunities for reuse of a design documented with an assurance case are significantly greater than for a device developed without assurance cases. This is especially true if assurance case patterns (discussed in the next section) are developed.
- Reapproval is made simpler even if the FDA never sees the assurance case behind a device. The case can guide the manufacturer directly to the artifacts that need to be concentrated on for the approval process, avoiding a waste of time on irrelevant ones.

6.3 Medical Device Archetypes and Patterns

Tim Kelly [Kelly 1998] has pointed out that:

> *whereas the detail of the safety arguments within the safety case is likely to change from instance to instance (being based on specific evidence), there is often commonality between the structures of argument used in safety cases. This is observed to be particularly true for safety cases within the same domain (e.g., aero-engine control or nuclear power plant design). This can be attributed to the stability of the certification requirements, forms of evidence used and maturity of knowledge in these domains.*

Kelly goes on to define the concept of a safety case pattern—a template (with usage instructions) that captures acceptable ways of structuring generic safety arguments. For medical device assurance cases, it would be helpful if a set of agreed argumentation patterns were available for use by medical device manufacturers and reviewers. If such patterns were provided for different aspects of a particular device's assurance case, they would help to show manufacturers and reviewers how to make effective use of assurance case technology. Since the patterns would provide examples, a barrier to adopting GSN would be reduced.

The FDA already provides guidance documents on the information needed in a submission. Much of this information can be recast in the form of an assurance case pattern. Because such patterns deal with fairly specific device safety issues and represent exemplary practice, we call them archetypes. For example, one section of a document on medical device use safety [Kaye 2000] provides information on user-interface testing issues and concerns. The following extract lists factors that a reviewer (or manufacturer) should consider when reviewing (or assembling) clinical test results pertaining to the safety of a user interface:

> *Certain characteristics of clinical evaluation research should be carefully considered when the intent is to demonstrate safety and effectiveness of device use:*
>
> – *Device users involved in manufacturer-sponsored studies might be biased,*

- *Device user-participants might not accurately represent the population of intended device users. (They are often more capable, motivated, or informed than intended users in general. Some users could have been involved in the development of the device.)*
- *Personnel who collect data might overtly or inadvertently help users use the device,*
- *The training received by users participating in evaluations could be more recent or more extensive than what would be reasonably expected for actual users.* [Kaye 2000]

This information could be represented in an assurance case archetype, as seen in Figure 12. This archetype is generic—the evidence element ("User Test Results," at the top of the figure) is to be instantiated with appropriate test results. (The little triangle at the bottom indicates an instantiation is required, and the words in braces indicate the nature of what is to be instantiated.)

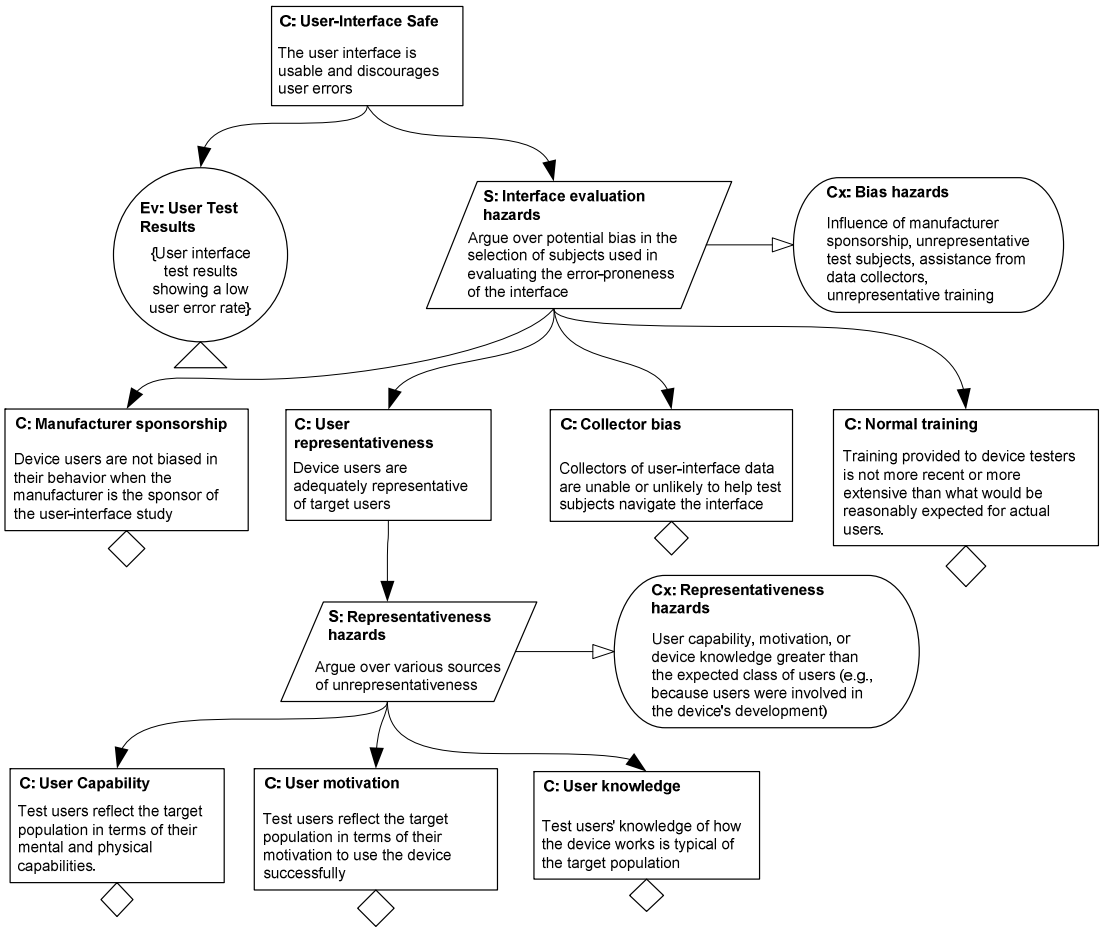

Figure 12: Capturing a Safety Archetype in a Goal-Structured Assurance Case

The archetype states a claim about the safety of the user interface. Part of the support for this claim is user test results. The argument recognizes that these test results could be biased unless the testing is properly controlled. The extract from Kaye and Crowley [Kaye 2000] indicates possible ways bias could arise, with unstated implications of how to control for bias. The assurance case example (Figure 12) shows both the possible biases and how they can be demonstrated not to arise. As shown in the figure, the argumentation strategy is to address, in turn, possible reasons why the test results might be biased, and then, in the rest of the argument, show how these potential evidentiary defects have been controlled. For example, consider the claim "User representa-

tiveness." One source of bias (as noted by Kaye) is an unrepresentative set of users, so we need to show that the set of actual test users is sufficiently representative. To do so, we state an appropriate claim: "Device users are adequately representative of target users." To show that this claim holds, we need to show that none of the various ways in which users could be unrepresentative holds for the particular submission under consideration. We then look at the various "hazards" to representativeness, and in turn, state a claim that each hazard has been addressed. Further refinement of the "User Capability" claim would, for example, provide evidence showing that users were randomly selected from a medical facility, with proper attention to ensure that the selection was truly (or sufficiently) random. If we expanded this figure, we could indicate alternative forms of representativeness evidence that would likely be considered satisfactory.

Figure 12 captures the guidance from the document [Kaye 2000] in a fashion that helps both the manufacturer and the FDA examiner (especially if the diagram is further extended to indicate the types of evidence that would be considered satisfactory in support of one of the claims).

The argument structure shown in Figure 12 is an example of a more general "Hazards to Evidence" pattern that we have described in a separate report [Goodenough 2008]. This more general pattern says that, since any evidence supporting a claim is potentially defective in some way, the quality of the argument supporting the claim is improved if potential types of defects are addressed explicitly to show why these evidentiary defects are unlikely to be present. We have simply instantiated this evidentiary defect pattern with respect to the user interface test results, using the discussion in Kaye and Crowley [Kaye 2000] to determine what defects need to be taken into consideration. A further exegesis of Kaye and Crowley [Kaye 2000] and a variety of other safety-related guidance documents and standards could provide a useful portfolio of medical device safety archetypes showing how potential defects in evidence need to be addressed and the impact of such defects on overall safety conclusions.

From the FDA reviewer's perspective, an assurance case containing a pattern or archetype that has already been successfully used in a submission makes it possible to concentrate on the evidence submitted without necessarily examining the whole argument, leading to more time-savings and more confidence in the approval. The reviewer could, if necessary, refer to that pattern to see how the evidence specifically supports the claim. But the fact is, given that the pattern has been used successfully before, the examiner only needs to ensure that the assumptions the case was created under still hold—at which point he only needs to consider the individual pieces of evidence. For the keyboard design, the list of evidence would include the HCI assurance case, test results, documentation that the "bounce" problem had been dealt with, and similar items. This listing of evidence seems very similar to the process-based checklist except that each item of evidence fills a specific need in the assurance case pattern that it supports.

6.4 Tooling

This report contains assurance cases developed using two tools—one an informally supported set of Visio macros and the other a commercially supported tool (the Assurance and Safety Case Environment (ASCE™)), available from the UK firm Adelard.[12] Figure 5 was prepared with the commercially supported ASCE tool. The other figures in this report were prepared using Visio macros made available to us.

[12] See http://www.adelard.com/web/hnav/ASCE/index.html for more information about ASCE.

The ASCE tool has been used on some very large projects [Di Nucci 2008] and includes features for hyperlinking elements of the case to complete documents or lengthy extracts from documents. Instead of just referencing documents in the case, the documents can be directly linked into the case, providing complete backup documentation directly with the case.

7 Concluding Thoughts

In this technical note, we've considered a practice of using assurance cases in the development and approval of medical devices. We developed portions of a possible safety assurance case for a GIP to help demonstrate the issues of using assurance cases in this domain. Working with industry, academia, and the FDA, we addressed some of the important issues surrounding the possible adoption of assurance cases by the medical device community.

Adopting assurance cases into the FDA approval process is going to take work by interested parties. There are forward-thinking manufacturers, personnel at the FDA, and people at the industry advocacy organization, AdvaMed, who appear ready and willing to go to some effort to make this happen. This relatively small group can't do it by itself. Adoption will take buy-in by a significant number of device manufacturers, and for this to happen they must be educated. AdvaMed, for its part, is taking a role in this by having sessions on assurance cases at workshops that it sponsors.

It seems to the authors that there are two activities that should be undertaken to ease the transition of assurance cases into the medical device community. The first is to increase industry awareness as to what assurance cases are and what benefits might be derived from their use. A viable approach to this would be to write a series of articles aimed at trade publications in different industries, exploring the assurance case approach and what adoption could mean to those industries. Artifacts used in the articles should be industry related. For this to be effective, the articles would have to be co-authored by people in the industry. The second activity would be to create and publish a series of FDA-approvable[13] archetypes for different kinds of medical devices.

[13] Approvable but not approved because the FDA has not yet adopted assurance cases as an allowed submission. Having a set of complete archetypes with their supporting materials would serve as useful guidance to the industry.

References

URLs are valid as of the publication date of this document.

[Arney 2008]
Arney, David, Jetley, Raoul, Jones, Paul, Lee, Insup, & Sokolsky, Oleg. *Generic Infusion Pump Hazard Analysis and Safety Requirements* (University of Pennsylvania Technical Report MS-CIS-08-31). University of Pennsylvania, October 2008.

[Di Nucci 2008]
Di Nucci, Simon. "Assurance Operational Systems—A Safety Case Study." *Proceedings of the Systems and Software Technology (SSTC) Conference 2008*. Las Vegas, Nevada (USA), April 29-May 2, 2008. http://sstc-online.org/2008/index.cfm?fs=pres&aid=1936&ld=530

[Goodenough 2008]
Goodenough John B. & Weinstock, Charles B. *Hazards to Evidence: Demonstrating the Quality of Evidence in an Assurance Case* (CMU/SEI-2008-TN-016), in preparation, 2008. http://www.sei.cmu.edu/library/abstracts/reports/08tn016.cfm

[Greenwell 2006]
Greenwell, W. S., Knight, J. C., Holloway, C. M., & Pease, J. J. "A Taxonomy of Fallacies in System Safety Arguments." *Proceedings of the 2006 Institution of Engineering and Technology (IET) Irish Signals and Systems Conference (ISSC 2006)*. Dublin, Ireland, June 2006. IEEE Computer, 2006. http://www.cs.virginia.edu/papers/paper-issc06-fallacies-as-printed.pdf

[Kaye 2000]
Kaye, Ron & Crowley, Jay. "Medical Device Use-Safety: Incorporating Human Factors Engineering into Risk Management: Identifying, Understanding, and Addressing Use-Related Hazards." U.S. Department of Health and Human Services, Food and Drug Administration, Center for Devices and Radiological Health. July 18, 2000. http://www.fda.gov/MedicalDevices/DeviceRegulationandGuidance/GuidanceDocuments/ucm094460.htm

[Kelly 1998]
Kelly, Timothy Patrick. "Arguing Safety—A Systematic Approach to Safety Case Management." PhD diss., University of York, Department of Computer Science, 1998.

[Kelly 2007]
Kelly, T. P. "Reviewing Assurance Arguments—A Step-by-Step Approach." http:// www-users.cs.york.ac.uk/~tpk/dsnworkshop07.pdf (2007)

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
(Leave Blank)	October 2009	Final

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
Towards an Assurance Case Practice for Medical Devices	FA8721-05-C-0003

6. AUTHOR(S)
Charles B. Weinstock and John B. Goodenough

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

8. PERFORMING ORGANIZATION REPORT NUMBER
CMU/SEI-2009-TN-018

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
HQ ESC/XPK
5 Eglin Street
Hanscom AFB, MA 01731-2116

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12A DISTRIBUTION/AVAILABILITY STATEMENT	12B DISTRIBUTION CODE
Unclassified/Unlimited, DTIC, NTIS	

13. ABSTRACT (MAXIMUM 200 WORDS)

Software technology enables an increasing percentage of medical device functionality, leading to much more complex systems and presenting a challenge to regulators charged with evaluating device safety and effectiveness. An approach to evaluating claims of safety increasingly used in Europe and elsewhere is the safety assurance case. Much like a legal case, the assurance case lays out an argument and supporting evidence to show that safety claims are valid.

This technical note explores the use of assurance cases for justifying claims of medical device safety. It illustrates the use of the assurance case on a particular type of medical device—the infusion pump. This example serves as a basis for discussing issues surrounding the introduction of assurance cases into the medical device community, which includes both manufacturers and the U.S. Food and Drug Administration.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Assurance case, goal structured notation, GSN	36

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL