

A map is worth a thousand words: Reliability of Electronic Evidence Diagrams

With a case study on a large NHS criminal case

Harold Thimbleby
harold@thimbleby.net
ORCID ID 0000-0003-2222-4243

July 30, 2025

Abstract

Almost all evidence today originates in digital systems or is processed through digital tools. Unfortunately, computer evidence used in courts is often problematic: computers are not reliable; computers may be configured, managed and operated poorly; computer evidence may be disorganised, complex and copious; the evidence may be affected by hardware failures, cyberattacks, and so on. Furthermore, in an investigation, digital forensic standards may not be upheld. The UK's current Common Law presumption that computer evidence is reliable further encourages miscarriages of justice because parties are unable to properly examine and check the computer evidence.

This paper proposes a visually-based approach to help better understand, communicate, and improve the quality of computer evidence, using Reliability of Electronic Evidence Diagrams (REEDs). The approach records and helps analyze the need for specific evidence, helping critique and improve the quality of what is already thought to be known. REEDs can also help manage IT systems prior to and regardless of possible legal action, and hence help improve the reliability of computer evidence used in investigations.

This article uses a large NHS criminal trial as a case study. The Police and the NHS's lack of awareness of their computer management failures and forensic problems emphasizes the importance of approaches like REEDs. Similar problems will face other organizations in other cases, so REEDs are likely to have broad applicability.

Table of contents (to support review)

1	Introduction to Reliability of Electronic Evidence Diagrams	3
1.1	Typical problems of computer evidence	3
1.2	How can REEDs be used?	4
1.3	Criticisms of diagram-based methods	4
1.4	A map is worth a thousand words	5
1.5	What does a REED consist of?	5
2	The Princess of Wales Hospital case study	6
2.1	The initial position	7
2.2	Representing the case study with a REED	7
2.3	Example narratives for a REED	7
2.4	Raising and resolving evidential gaps and problems	8
2.5	Updating a REED with new evidence	12
2.6	Using highlighting	12
2.7	Unresolvable disagreements	13
3	Learning points from the case study	13
3.1	On reliability	13
3.2	For the police: improve digital forensics literacy	14
3.3	For the Princess of Wales Hospital	14
3.4	For the NHS and other organizations more generally	15
3.5	For Abbott, manufacturers, and regulators	16
4	The future of REEDs	16
4.1	Why current REEDs are simple	16
4.2	Making REEDs easier to use	16
4.3	REEDs could be treated like car MOT certificates	17
5	Conclusions	17
6	Acknowledgements	18
7	References	19
	— ONLINE APPENDICES —	21
A	Alternative notations for REEDs	21
B	Semantics of REEDs	21
C	A tool for REEDs	22
C.1	Key benefits of a tool	22
C.2	REED identification	22
C.3	Drawing diagrams	23
C.4	Highlighting nodes	23
C.5	Strings	24
C.6	Notes	24
C.6.1	Cross-referencing nodes and evidence	25
C.6.2	Combining HTML and \LaTeX notes	25
C.7	Using versions to override properties	26
C.8	Summary of further features	27
C.9	Command-line features of the prototype REED tool	27
D	Additional appendix references	28

1 Introduction to Reliability of Electronic Evidence Diagrams

This document introduces the *Reliability of Electronic Evidence Diagram* (abbreviated REED) as a tool to help support understanding and critiquing computer evidence. REEDs are a simple visually-based approach. This paper will argue that REEDs have important properties:

- REEDs help overcome the “inequality of arms” whereby one party (usually the defence) does not have adequate access to or understanding of the computer systems and data processing such that they are not in a position to ask critical questions.
- REEDs make usable contributions even when fully informed IT expertise may not be readily available.
- REEDs are effective in pre-proceeding discussions and for courts actively managing cases [10,20].
- Providing REEDs to a court is not burdensome; it follows that being unable or unwilling to provide REEDs (or similar) might be taken to imply that the computer systems and evidence have not been managed adequately, and therefore that evidence from them is more prejudicial than probative [4].
- REEDs may also be very helpful in managing operational computers.
- By supporting informed critique of complex evidence, REEDs can help improve the quality and reliability of evidence.
- REEDs are currently supported by a prototype tool, which makes drawing, checking and managing them very easy. (The tool was used to draw all REED diagrams in the present paper.)

1.1 Typical problems of computer evidence

Mere assertions that computer evidence is reliable because there have been no detected problems are not credible, as failing to detect problems does not show that there are no problems but that the auditing was itself unreliable, as there are always problems to detect! Indeed a common prosecution argument (e.g., as used in the Post Office Horizon trials [2,9]) is that as the computer systems have run reliably, perhaps for thousands of transactions, then (they argue) the defendant’s claims of computer problems are implausible. This is an elementary fallacy; the prosecution needs to show that the computer systems were reliable under the specific conditions the defendant experienced.

Figure 2 is the common, idealized view that makes it look like the IT is a coherent, simple system. This view is consistent with the naïve Common Law presumption. In reality, though, any IT system has hidden complexity and numerous external influences that potentially undermine its reliability. Factors include: cyberattack, operator error, bugs, networks, cloud system problems, hardware faults, faulty maintenance, incorrect configuration, bad data. In hospital environments, there are additional factors such as lack of interoperability, middleware (and multiple versions of middleware), multiple standards partially implemented, poor user training, running obsolete code, code written by unqualified programmers (e.g., poorly programmed Excel spreadsheets), incomplete data because of legacy reliance on paper records, and more. In short, a real hospital IT system is very unlikely to be reliable, and possibly unable to provide reliable evidence for use in litigation. Furthermore, few people will understand the IT system and the reliability of its evidence.

Without support to better interpret and challenge computer evidence injustices are likely to follow. Hence this paper’s proposal of REEDs to help visualize and document how IT systems process evidence, and they are essential to help understand how electronic evidence is formed, routinely processed, and — critically — how evidence may be accidentally or intentionally lost or compromised. Using REEDs encourages disclosing detailed information about electronic evidence, and for parties to reach consensus over relevant details. Indeed, if an adequate REED (or equivalent) is not disclosed, the presumption arguably should be that the operators or owners of the computer system do not know enough about the system and its management — implying that any evidence generated by it is unreliable and should be inadmissible.

The classic review of computer evidence is [12].

1.2 How can REEDs be used?

Preliminary REEDs will typically be drawn up by the parties to litigation in the normal course of their work or by expert witnesses, and then shared and discussed to mutually fill in details to create a shared REED, which will be disclosed to all parties. Any remaining areas of uncertainty should lead to further investigations or calling of witnesses. Complex proceedings may require several REEDs, and will require resolution of inconsistencies between various REEDs.

In some cases involving computer evidence, the prosecution will assume the computer evidence is correct (following the Common Law presumption [2]), but the defendant needs to find out how the computer systems have failed. Courts do not like such ‘fishing expeditions’ yet IT systems are exceedingly complex, and getting to the root issues in a case may in practice require fishing of some sort.

I have built a fully working prototype tool for creating and managing REEDs. It is described in more detail in Appendix C.¹ The tool makes REEDs from textual description written in a normal document and creates a diagram and linked evidence narrative. The tool ensures it is easy to perform updates without making a mess of the diagram or its cross-referencing to the narratives. For instance, the tool numbers each part of the diagram for easy reference, and it automatically keeps the numbers consistent with the relevant sections in the narrative. Hence as the experts discuss and adjust their REEDs, the tool takes away the drudgery, and also checks for errors like overlooking providing narrative for some node. REEDs are easy to email and edit, and do not require any drawing skills.

1.3 Criticisms of diagram-based methods

A powerful criticism of diagram-based approaches is that the diagrams may be merely “PowerPoint Engineering,” a derogative term introduced by Sir Haddon-Cave [7] in his critical analysis of the total loss of an RAF Nimrod aircraft and all 14 crew and specialists on board in Afghanistan in 2006. PowerPoint Engineering is creating diagrams that deceptively look sophisticated but which conceal engineering oversights. Since REEDs are a flexible concept, misuse of REEDs can indeed lead to the problems of PowerPoint engineering Haddon-Cave noted, but the emphasis with REEDs is on iterative cooperation where all parties in legal proceedings jointly review, critique and contribute to the developing REEDs and narrative descriptions. Authors and contributors to REEDs will include independent expert witnesses competent in computer science. The social process therefore ensures the REEDs are a constructive, insightful contribution to the proceedings and avoids PowerPoint engineering.

A method called Theory of Change (ToC) is widely used across UK government departments and agencies to plan and demonstrate public value [5]. The Government promotes Theory of Change as a way of clarifying ideas for achieving effective change and for reaching consensus on the ideas [5]. Central to the ToC approach, like REEDs, is working cooperatively on the diagrams to agree and make issues explicit, and in particular to help notice critical steps that have not yet been considered. There is an interesting comparison to be made with the Theory of Change and this article’s ideas for REEDs. (Section 3.3 uses ToC to present NHS learning points from this article.) Theory of Change, like REEDs, is represented as a diagram of nodes and arrows with connected narratives, with a social process to find and fix problems and to reach consensus. Overall, ToC is a similar approach to REEDs, and its success (and the research behind that success) is an argument supporting the diagrammatic-collaborative process that REEDs also use.

REEDs will normally be used with a checklist, so that parties can confirm that relevant issues have been addressed, *and evidence is provided that they have been addressed* [6]. Examples of typical issues are as follows:

- Testing protocols;
- Known error logs;
- Records of updates and reasons (as are recorded in repositories such as GitHub);
- Management of operators;
- Management of cybersecurity;
- Forensic management of evidence;

¹ It is envisaged that all appendices for this paper will be purely online, however the appendices are included as part of the present paper to facilitate refereeing.

- External IT audits;
- Specific warranties that the computer systems are fit for purpose;
- Note that standard software engineering resources [15] and international standards provide further topics.

In a competent professional environment, all relevant documents will already have been computerized, and hence accessing relevant evidence will be a simple matter of sharing links or emailing to make evidence available. Indeed, if an organization has already had an IT audit, relevant documents will already have been bundled and emailed to the auditors.

As any competent developer or user of computer systems will follow good practice, the electronic copies of the relevant information will be trivial to provide. Indeed, absence of or reluctance to disclose details such as those listed above should be taken as a flag that the computer evidence is unreliable.

1.4 A map is worth a thousand words

Evidence about computer systems and their data can be overwhelming and, worse, it is impossible to see if details are missing or over-simplified. Details are often omitted because it is easier and, anyway, nobody will notice.

A common phrase is “a picture is worth a thousand words” but we do not often think about what that really means.

Take a road map as an example: the map is a picture of all routes from anywhere to anywhere, plus lots of details — from the locations of post offices to national monuments, contours and tidal ranges, gridlines and more. If a map was solely a written document, it would be overwhelming, but when it is a pictorial map, we can pick out any route we need to follow to get to our destination without being overwhelmed with irrelevant detail. Because a map is a picture, we can see details that are relevant to our journey without having to sift through details in other parts of the map.

REEDs provide a map for evidence, and are particularly for complex evidence, such as that originating in connection with computer systems. Although a few symbols are used in REEDs, the key contribution of REEDs is the thorough cross-referencing to sections of textual narrative (possibly with illustrations). Someone reading a REED can go from a symbol on the map of the REED to one or more paragraphs of text explaining it, and from any paragraphs back to the points on the REED map connecting them to the evidence pathways, which are arrow routes on the diagram.

Except in the very simplest of cases, in conventional textual and verbal evidence *some or all* evidential paths will be invisibly missing or over-simplified. REEDs fix this problem because, like a geographical map, a REED diagram is a map showing all possible evidential pathways of interest cross-referenced to detailed narratives, something conventional sequential evidence can never make explicit because it would be overwhelming to properly document all possible evidential pathways. Additionally, with tool support as described in this paper, REEDs can also be checked automatically as providing all relevant documentation.

1.5 What does a REED consist of?

A REED is a controlled document that has a date, version number, and responsible author(s).

1. A REED document may state that the computer system and all other sources and processing of electronic evidence are ‘approved devices’ (such as speed cameras), and therefore no further details justifying the reliability of the systems will normally be required, beyond statements of truth. At a minimum a REED diagram should be provided to show *how* the device or devices are used and managed, who is responsible for them, and to make explicit the trail of evidence pathways back to reliable sources.
2. In other cases a REED consists of:
 - (a) One or more diagrams showing the major sources and processing of information leading to evidence presented, including relevant sources of information *not* presented in evidence. (Examples are provided in this paper.)
 - All sources, processors, and destinations of evidence, including humans, are called *nodes* in the diagrams. Many processors of information will be computer systems.

- Nodes are connected by *arrows* showing the flow of evidence data.
- (b) There is a cross-referenced narrative analysis for each node (and optionally each arrow), at least explaining the purposes and possible limitations of its generating or processing evidence.
3. It is expected that a computer tool is used to help prepare REEDs, and such tools will check the consistency, completeness, and integrity of the REEDs. In particular, a tool will assure that relevant narrative analysis has been provided for each node. The prototype tool used in this paper produces a digitally signed certificate that it has checked compliance.
 4. It is possible that parties cannot produce or agree on a common REED ready for a pre-trial hearing. This problem should be addressed in court *before* any reliance is made of electronic evidence, since without a worked-out narrative consensus which a REED would represent implies that the electronic evidence itself and/or the people managing it are unreliable, or at least the electronic evidence and evidence related to it should be presumed unreliable until the discrepancies or other problems have been resolved.

2 The Princess of Wales Hospital case study

This article presents a case study based on the Princess of Wales Hospital 2014 NHS criminal case [16],² where 73 nurses were suspended when it was thought that they had neglected patient care. Subsequently, five nurses were prosecuted: three nurses pleaded guilty and received custodial sentences; two pleaded not guilty and went to trial.

The case may be briefly summarized [4, 8, 16, 17]:

One of the side-effects of stroke is it can affect diabetes control and can make a sufferer unable to recognize symptoms and control their blood sugar. Accordingly, regular blood glucose monitoring is an essential part of the care of such patients. Nurses working on a specialist stroke ward at the Princess of Wales Hospital, Bridgend, used a handheld blood glucose monitoring device, the XceedPro blood glucometer manufactured by Abbott.

Using the XceedPro, a nurse scans their own staff barcode ID and the patient's barcode, and after a blood reading is then taken with the XceedPro the nurse also records the details on paper.

The measurement data is also stored in the XceedPro until it is docked, and the data is then automatically uploaded to the hospital's database, PrecisionWeb. PrecisionWeb also records other details such as the glucometer's battery level and the temperature of the ward. Ultimately, the data will populate the patient's electronic patient record.

Under normal circumstances, the nurses' written records and the PrecisionWeb database records will be consistent with each other.

There were 130,978 blood glucose readings (together with dates, times, ward numbers, patient and nurse identifiers, etc) submitted as evidence. It was alleged that the defendants (and the other 68 nurses not prosecuted) had fabricated blood sugar readings to write up in the paper records, so the nurses would have been criminally negligent as the stroke patients had reduced capacity. The prosecution points to the fact that the nurses had written up paper records that did not correspond to anything recorded in XceedPro glucometers nor on the PrecisionWeb database. The situation is illustrated in figures 1 and 2.

The expert witnesses pointed out that the evidence was not complete. For instance, PrecisionWeb maintained a 'reject folder' of problematic data (a reject could occur if a nurse ID had been scanned in place of a patient ID, as might happen if the patient's barcode was damaged, just to get the XceedPro to work). In the second week of proceedings, this reject folder was finally produced in court: it contained 18,546 rejected records, but none of the rejected records matched the missing data. The statistical distribution of the reject data very strongly suggested some form of sudden manipulation and (I argued) it was not possible to countenance an explanation consistent with the data being reliable.

In many ways, the case study is a small version of the Post Office Horizon scandal [2, 19]; in both, miscarriages of justice were exacerbated by the absurd attitude that computer systems are reliable black boxes, and that evidence generated by computer can be presumed correct.³

I was an expert witness in this case, and I explained and distributed figures 2 to 4 in court over several days during my cross-examinations. These diagrams (plus my evidence) were a

²R v Cahill, R v Pugh, 14 October 2014 at Cardiff Crown Court (T20141094, T20141061). The judge's ruling has been published [4].

³The metaphor "black box" implies one cannot see any of the details inside the box.

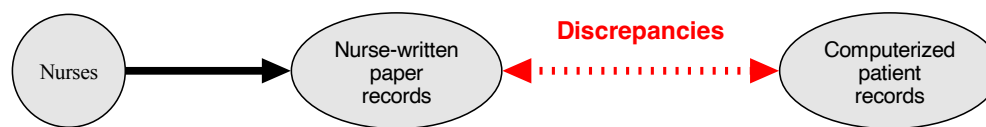


Figure 1. Nurses’ handwritten paper records and the hospital computer records were inconsistent; in particular the computer evidence showed that nurses had not performed some of the tests that they had written up in their handwritten notes. The fact that computer evidence “showing” something may not be reliable evidence had not occurred to the prosecution, and was denied when I raised the possibility, even despite the implausible assumption that 73 nurses had all been negligent and dishonest in their record keeping in exactly the same way.

successful prototype of the more formal REED proposed in this article. More background and further details of examples discussed in this paper are available at <https://harold.thimbleby.net/reeds>

Because of the technical analysis presented and clearly communicated by my prototype REED approach, the prosecution called Nick Reece, a PrecisionWeb support Specialist employed by Abbott Diabetes Care.

Mr. Reece had been asked by the hospital to help when the PrecisionWeb database faced problems in 2013. In court, he said he had visited the Princess of Wales hospital on a date when data had disappeared from the relevant databases. He admitted deleting it. The judge concluded that the prosecution evidence was unreliable and was therefore excluded [4]. The prosecution response was to offer no evidence. In consequence, the nurses who had pleaded not guilty were acquitted.

It is noteworthy that the hospital and the prosecution had not previously considered that the explanation of the discrepancies was or might have been that computer data had been deleted rather than the written nurse records had been fabricated by 73 nurses (somehow colluding to do so consistently). When I raised such possibilities in pre-proceedings discussion, the prosecution position was that the nurses “were all in it together” — which they asserted without any evidence, and without having explored simpler possibilities such as a computer technician with a grudge or a cyberattack. However, terminating the employment of nurses (as happened to many nurses in this case) is cheap and appears to be a definitive solution, enabling the hospital to move on. See section 3.3 for more detail.

2.1 The initial position

Figure 1 provides a high-level overview of the Princess of Wales case, and figure 2 expands figure 1 to make clear the central role of the hospital IT systems. I distributed this second diagram in court to help argue that the hospital IT system was a potentially significant player in the discrepancies that, at that point, had been entirely attributed to the nurses.

Figure 3 starts to show the real complexity of the hospital system, as cross-examination started to explore the evidence. After a few days, I was able to present figure 4 to help explain some of what to me were problems that could undermine the prosecution case (summarized by the judge [4]).

Notice how the diagrams are starting to get complicated, and keeping them up to date, correct, and just neatly drawn is starting to get hard.

2.2 Representing the case study with a REED

Based on figure 4, we now explore how a clearer and more formal REED, figure 5, could be used in the case. This new diagram was drawn with a tool to help draw it and to help keep its narrative discussion consistent with the drawing. The complete REED used for the case study can be explored interactively at <https://www.harold.thimbleby.net/reeds/pow-reed.html>

The REED shown in figure 5 is called version 2 because it has already benefitted from information uncovered during the previous weeks of cross examinations. The red warning labels in figure 4 are now separated from the diagram and put into the narrative evidence, as illustrated in the extracts in section 2.3 below.

2.3 Example narratives for a REED

A REED has narrative evidence covering all nodes, and optionally some or all arrows. In the case study, outline narrative evidence was written for this article, but in an actual case the narrative

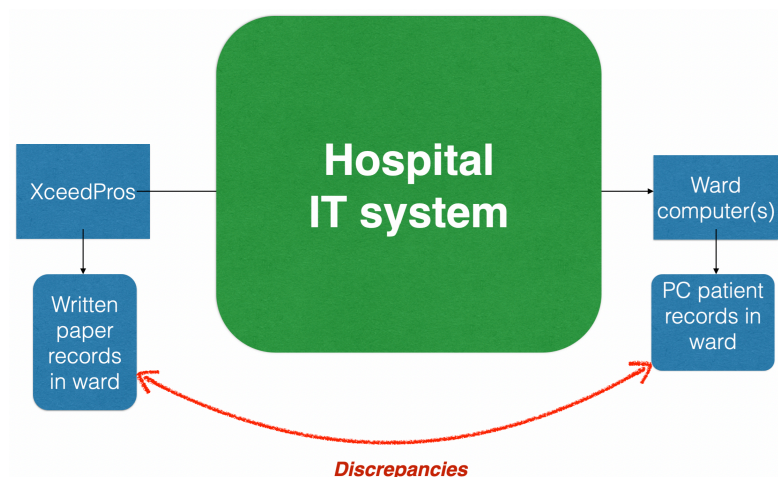


Figure 2. This is an original diagram distributed in court. Everybody agreed on this outline of the case, and most thought that the defence case was hopeless because of the (apparently) substantial computer evidence.

The diagram above indicates how nurses use XceedPro glucometers to obtain blood glucose readings from patients; the nurses record on paper what they have done. In addition, the XceedPros copy their blood test results to ward computers, but the computer evidence showed no blood glucose readings had been taken on multiple occasions. Clearly, it was alleged, the nurses fabricated their handwritten notes to show they were caring for patients when in fact they were neglecting them.

evidence would generally be more substantial.

Two small extracts from the outline narrative illustrate how the police compromised the evidence, moreover in a way that contradicts the Common Law presumption that computer evidence is reliable.

First, here is an example arrow narrative, extracted automatically from the full narrative:

Arrow E

v2-4.2 Abbott PrecisionWeb database

→ v2-5.1 Police forensic database system

The Police exported a CSV file from the PrecisionWeb database of every record derived from all XceedPros in the hospital (see arrow D) over the relevant period of time. **CSV is a non-forensic and very unreliable data format.**

In addition, the arrows labelled ‘?’ in figure 5 indicate evidence flows we *should* know about, but the relevant information was not made available despite requests (it probably does not exist).

Second, here is an example node narrative:

Node v2-1.1 Wrong XceedPros seized by Police

— (Group: XceedPro evidence)

The PrecisionWeb data records XceedPro serial numbers, and shows that XceedPros routinely move around the hospital.

The PrecisionWeb data confirms that the Police seized all three XceedPros that happened to be on Ward 2 on the date when they visited it. However, the seizure did not include any XceedPros that had been used by the defendants.

Seizing the wrong XceedPros will give the false impression that the XceedPro data confirms the Prosecution’s contention that the nurses’s fabricated meter readings. The seized XceedPros were checked as reliable by Abbott (node v2–2.1, “Abbott labs”) — that is, Abbott unsurprisingly confirms their XceedPros work as they expect, not that they checked the *right* XceedPros.

Reference [16] explains how the Police came to seize the wrong XceedPros.

→ v2-2.1 Abbott labs

→ v2-3.1 Police summary of wrong XceedPro evidence

2.4 Raising and resolving evidential gaps and problems

There are two reasons why a REED might need updating: thinking through a diagram makes it clear that there are more things that need representing, and evidence disclosed or uncovered through cross-examination means more evidence needs representing in the diagram.

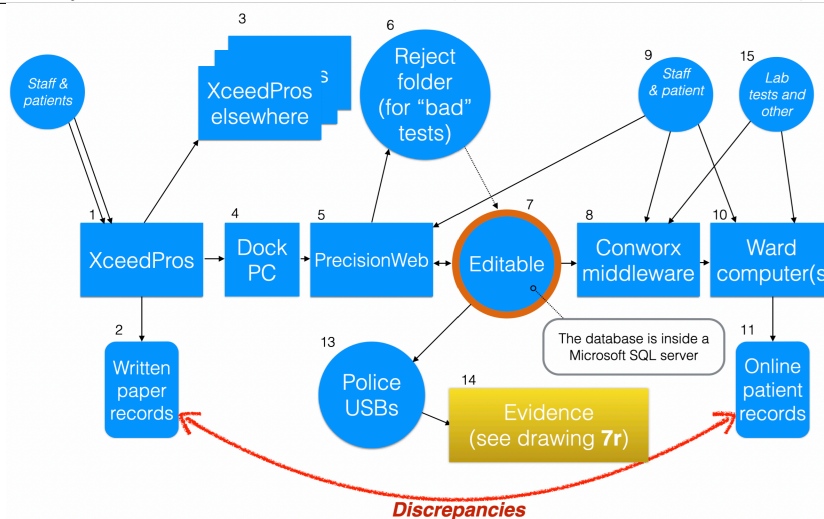


Figure 3. As shown in this original diagram distributed in court as it became clear that the hospital IT systems were far more complex than the prosecution had portrayed them. For example, the PrecisionWeb database used a data format the main hospital systems did not recognize, so the middleware Conworx was required to change formats. Conworx had hospital technicians managing it, and they would have been able to interfere with its data had they wished.

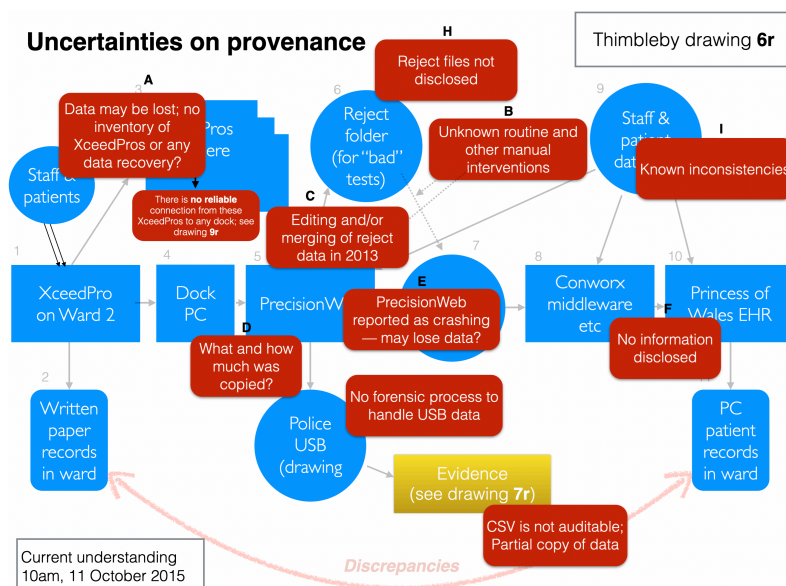


Figure 4. Another original diagram distributed in court. As the court examined witnesses, it became clear that *the reliability of the police evidence was compromised* — all of the red boxes overlaid on this figure outline serious problems. In particular, the so-called forensic methods used by the police were unreliable.

The narrative evidence for any REED *must* include a discussion of anything significant that may be missing from the diagram.

Dashed arrows are used in REEDs to show lines of evidence that had not yet been presented to the court. Such lines will be documented in the ‘missing’ section of the narrative.

Figure 5 shows node v2-1.4 for hospital computer operators, which begs the questions *who* are these people, and *who else* has access to the hospital IT systems?

Almost all IT systems have end user licence agreements (EULAs) that allow the manufacturer or developer to deny any warranty for reliability. EULAs are typically very long and tedious; it is suggested that expert witnesses share relevant EULAs and extract any relevant clauses from them for the REED.

For example, the Abbott PrecisionWeb database operator manual specifically says PrecisionWeb cannot be used for clinical purposes, which implies it is not reliable enough for evidential purposes.⁴ (PrecisionWeb seems designed primarily to help monitor and maintain XceedPros

⁴More precisely, the PrecisionWeb warranty shows that PrecisionWeb is not reliable enough for the manufacturers to warrant; whether it is reliable enough to provide probative evidence for use in court for a particular case is another matter. If PrecisionWeb evidence is used in court (as it was in the case study) there needs to be specific and substantial evidence from Abbott to show whether PrecisionWeb is reliable enough — despite the manufacturers saying it is

Princess of Wales Hospital blood glucometer case REED v2, 26 January 2025

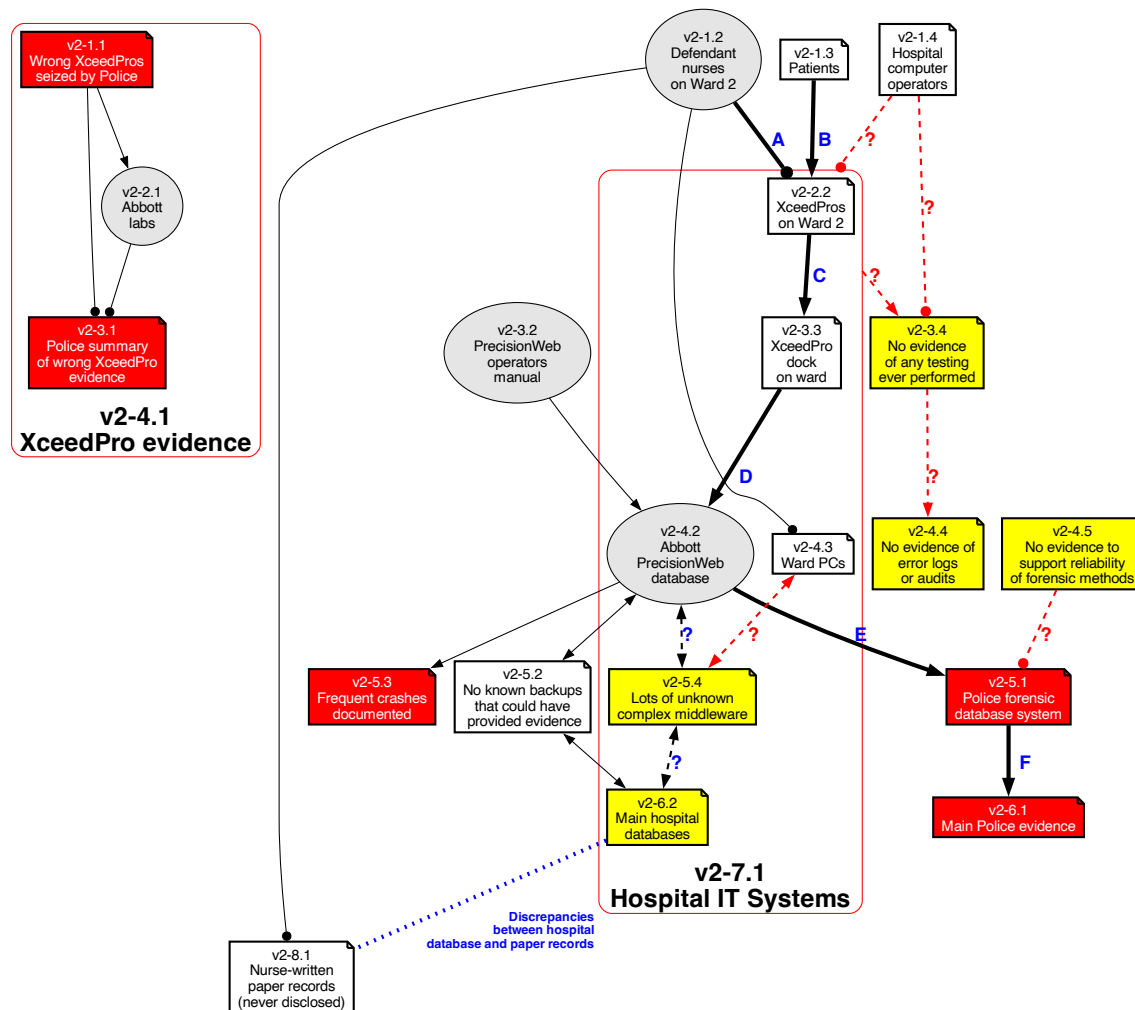


Figure 5. Redrawing key parts of the original figure 4 as a REED, including some further information that had come to light during the early stages of the case. Note that a REED also includes narrative evidence, which will be cross-referenced back to the nodes and arrows in the diagram — examples of narratives are provided in the article.

Nodes can be highlighted to raise points for attention: in this REED some nodes are highlighted in red (indicating problematic evidence), and in other colors to flag other issues as needed. For a legend for all the colors used here, see figure 6. For accessibility reasons or if the printer used does not show colors, the REED tool can optionally label all nodes with their color names.

	red used 5 times	Specific, relevant computer problems as already admitted in evidence. Use of non-forensic tools like Excel (which, for instance, allows rows of data to be deleted <i>without leaving any record of changing the data</i>) used to process the evidence. In short, any evidence highlighted in red is unreliable.
	white used 7 times	No information available (yet). This may or may not be considered a problem after relevant evidence is provided.
	yellow used 5 times	Problems that have not yet been resolved. Concerns need to be addressed in cross-examination.

Figure 6. Color key for the REED shown in figure 5.

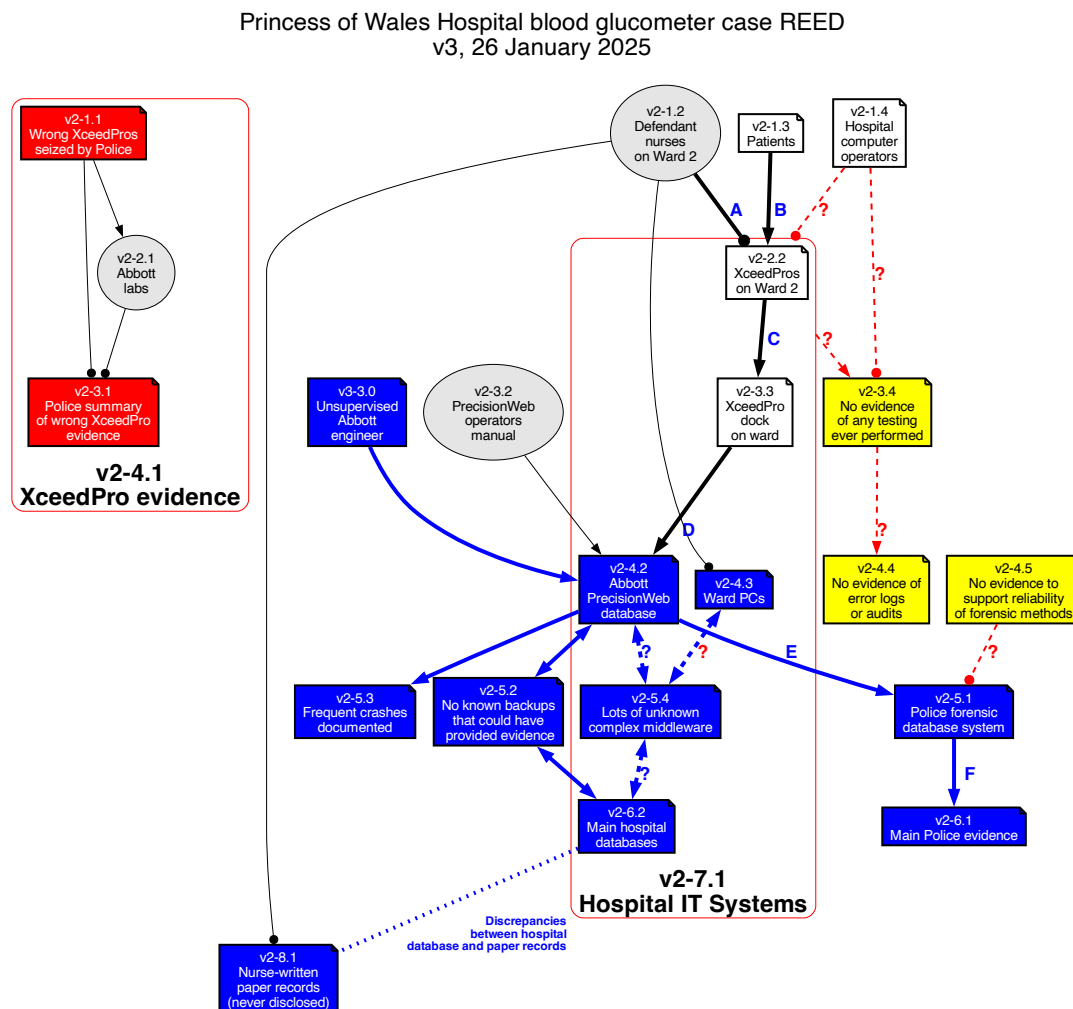


Figure 7. REED version 3 has introduced a new node (v3-3.0, on row 3) to represent the evidence of the Abbott engineer's interference. The REED author highlighted the new evidence and told the REED tool to automatically highlight everything affected, so the scope of impact could be appreciated easily. It appears, then, that the main police evidence (v2-6.1) is compromised, and the case's central discrepancy (indicated by the dotted line between v2-6.2 ... v2-8.1) has been discredited.

The color legend for this figure is provided in section 2.6.

— for instance to identify low battery conditions that need rectifying.)

Figure 7 makes it clear that there was no evidence available on hospital IT system management or how paper notes were managed, and what (if any) precautions (e.g., to detect and recover from cyberattack or hardware failure) there were in place. I only noticed this as a problem to raise explicitly when I drew figure 7, which is the point of using a REED diagram — and then the REED tool automatically required the point to be documented.

2.5 Updating a REED with new evidence

Nick Reece, a PrecisionWeb support Specialist employed by Abbott Diabetes Care, was called by the prosecution to give evidence. It transpired that the Abbott engineer had had unsupervised access to the hospital PrecisionWeb database. The engineer revealed how his ‘tidying up’ of the database had destroyed the computer evidence that would have confirmed the professional behaviour of the nurses.

The engineer’s evidence introduced facts that would have been used to update the REED to a new version, as shown in figure 7. With a new node added, the tool automatically requires additional new narrative text to explain it. Hence the following text was added to the narrative:

Node v3-3.0 Unsupervised Abbott engineer

Nick Reece, a PrecisionWeb support Specialist employed by Abbott Diabetes Care, had been asked by the hospital to help when the Princess of Wales PrecisionWeb database faced problems in 2013. Mr. Reece then worked unsupervised on the PrecisionWeb database, apparently taking no notes and certainly not following any forensic process.

Reece gave evidence that he had failed to take notes on exactly what he did when editing, deleting, and modifying data in the PrecisionWeb database. He admitted he had deleted critical data, which would have created the impression that nurses had been negligent not performing blood glucose tests.

This deletion of computer evidence explains the discredited pancy in patient records that the prosecution claims had been wholly and entirely caused by the nurses’ criminal negligence.

→  v2-4.2 Abbott PrecisionWeb database

Like other node narratives, the narrative example above has an arrow → showing the evidence node (or nodes when more than one) this evidence may affect, exactly following the connections shown in figure 7. The REED tool can also generate HTML narratives, and then the arrow links (both forwards and backwards) become dynamic hyperlinks, so a reader can easily navigate around the structured narrative by clicking on the links.

Although illustrated in this article as figure 7 to show how it might have been drawn to support further cross-examination, no new version of the REED was actually needed in court as the prosecution now conceded they consequently had no evidence. The case was dismissed.

2.6 Using highlighting





Highlighting is used in REEDs as an easy and clear way of flagging nodes and arrows of particular interest.

REEDs do not impose any fixed interpretation for colours, but allows colours to be defined to suit the case.⁵ Colours can be used to highlight points of evidence where the experts disagree, and perhaps which should be addressed in cross examination.

In the REEDs used for the case study, the narrative text has the following highlighting details and colour meaning key, as provided by the author, automatically added to the narrative:

unreliable in other respects.

⁵The REED tool has an option to add the highlight colour as an explicit textual explanation on each highlighted node for accessibility reasons or in case printers do not have colour.

Highlighting key		
	blue used 10 times	Version 3 uses blue to highlight the impact of critical new evidence introduced by the Abbott engineer. Only one node was explicitly highlighted blue, but the REED tool is used to automatically cascade the blue highlight to all affected evidence. (blue used explicitly once before cascading)
	red used 2 times	Specific, relevant computer problems as already admitted in evidence. Use of non-forensic tools like Excel (which, for instance, allows rows of data to be deleted <i>without leaving any record of changing the data</i>) used to process the evidence. In short, any evidence highlighted in red is unreliable. (red used explicitly 5 times before cascading)
	white used 4 times	No information available (yet). This may or may not be considered a problem after relevant evidence is provided. (white used explicitly 7 times before cascading)
	yellow used 3 times	Problems that have not yet been resolved. Concerns need to be addressed in cross-examination. (yellow used explicitly 5 times before cascading)

This of course is basically the highlighting legend shown in figure 6 but now updated for version 3 of the REED, for instance with details of the new blue highlighting.

Normally, highlighting affects only specific nodes or arrows, but the tool has a useful feature to ‘cascade’ highlighting across the diagram. In the case study, the REED authors only needed to explicitly highlight the one new piece of problematic evidence (the unsupervised Abbott engineer and his actions), and the REED tool then automatically propagated the highlighting throughout the diagram (see figure 7).

It is notable that node v2-8.1 has now been automatically highlighted because of highlighting the engineer’s evidence. Potentially, the nurse-written paper records, which were selected by the Prosecution, are no longer the right records.

2.7 Unresolvable disagreements

It is possible that parties do not agree on the interpretation of some evidence. REEDs therefore allow the narrative evidence to present conflicting statements.

When conflicting notes are made for any node or arrow, the REED tool reports the duplicate note(s) and requires different authors for the notes. The notes are then highlighted, with details of authorship etc, and the diagram has markers drawn on the nodes or arrows to highlight the existence of conflicts.

In addition, if parties disagree on the status of a node, it can be highlighted with a colour designated to highlight the disagreement.

3 Learning points from the case study

The use of REEDs and the details of the case study raise numerous learning points which are summarized in the following sections:

3.1 On reliability

The worked example in this article shows that an apparently reliable hospital system used successfully for years failed, regardless of how long it had previously appeared to be reliable. In fact, the hospital provided no evidence that their computer systems had ever been reliable — there was no auditing or other record that the hospital had ever tried to establish how reliable their systems were. The hospital, the police, the prosecution team including the prosecution experts just assumed the electronic evidence was reliable at face value.

The naïve argument that because computers, blood glucometers, etc, have seemed reliable in the past, or reliable in lots of other places, never means their evidence is reliable for the *particular* case under consideration.

The same style of misleading argument about computer reliability was also made repeatedly in the Post Office Horizon scandal [19]: the computer system Horizon worked well for millions of transactions up and down the country so the prosecution would like the court to draw

the conclusion that the defendant's Horizon accounts are reliable and therefore prove that the defendant is guilty.

Of course, the very fact of litigation in any case implies *something* has gone wrong, and therefore a blasé dismissal that ignores the specific details of the case is unlikely to be correct.

3.2 For the police: improve digital forensics literacy

The Princess of Wales case study shows how the police's approach to forensic IT investigations was naïve, and in the event counter-productive.

Forensic IT evidence should include features (at a minimum, checksums and digital signatures) so that it can be independently checked for completeness and integrity.

3.3 For the Princess of Wales Hospital

An NHS inquiry followed the court case, *Review of the Blood Glucometry Investigations* [...] *establishing lessons learned* [8], and then various committee meetings. Curiously, the hospital determined that as they had upgraded their wired networks to wifi the problems must have been fixed; moreover the Clinical Director claimed there was nothing for the hospital itself to worry about because three nurses had pleaded guilty and been convicted.

The Princess of Wales hospital has been widely reported as saying [21]:

"We can give assurances that regular checks since [the court case] have revealed no further problems.

Issues around the care of patients in the Princess of Wales Hospital, including the blood glucometry test issues, led to the commissioning of the 'Trusted to Care' Andrews Report.⁶

The follow-up review was published last month and gave reassurances that the care of frail older people is now much better.

We remain determined to do more, to continually improve, and continue to have a commitment to dealing with issues as they arise in an open and transparent way. Like everyone else we have just heard this news about the trial and we will now need to take time to reflect."

On the contrary, the problem exposed by the case study was *not* anything to do with frail older people, but serious problems with the management of computer systems. It is inevitable, then, that the "regular checks" mentioned as reassurance will continue to miss the underlying problems that have not been acknowledged. The regular checks are unlikely to help if the hospital does not address why it failed to preserve, *and did not realize it was not preserving*, critical evidence it used in a court case. Furthermore, as some nurses pleaded guilty before the case, their cases should be reviewed as the grounds on which they were convicted and struck off are now known to be unreliable.

The expert witness reports to the court contain justified statements like this from a joint report:

"The database content exhibits serious management issues that have and had not been addressed." [18]

The NHS should have carefully read such expert witness reports, and (given that the case collapsed and therefore did not explore all issues) asked the experts for more insights. For instance, the hospital had not enabled basic functions the PrecisionWeb database supported to help check the integrity of data [4]. The recommendations of the internal review [8] to involve me on a number of points were never followed up.

Section 1.3 used the effectiveness of the Theory of Change to help argue the benefits of REEDs. Figure 8 shows how a Theory of Change can be used to summarize this article's insights about the hospital case.

A collection of relevant documents and commentary are provided in reference [18]. I wonder if the hospital had had anything as clear as a REED whether their response would have been different.

⁶Cited in this article as reference [8].

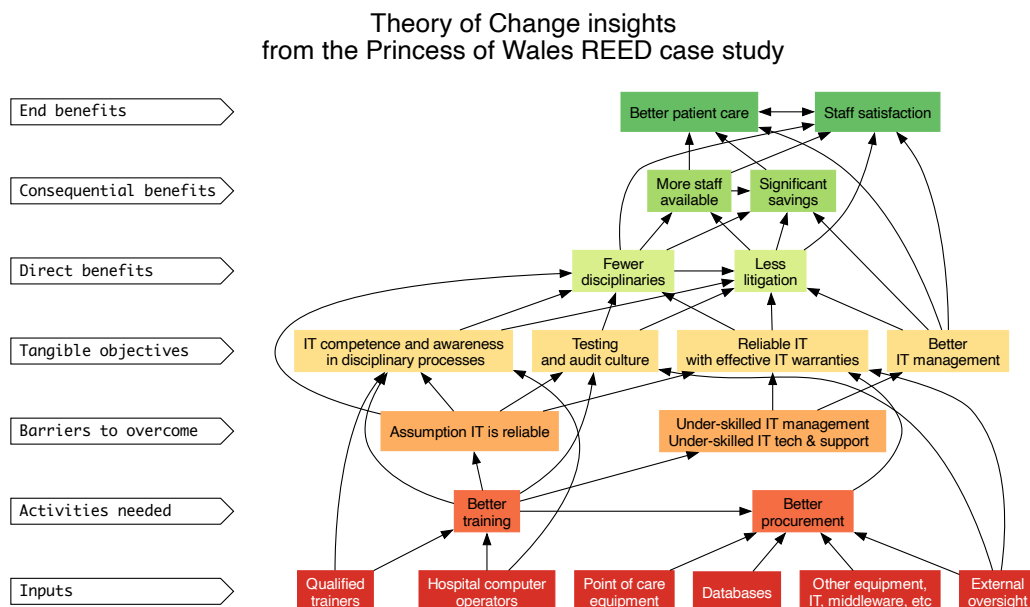


Figure 8. This diagram represents insights drawn from this article derived from the REED analyses of relevance to the NHS healthcare practice, and represented as a concise Theory of Change (omitting many obvious details like resourcing and the high-level planning).

The diagram above was prepared using the same tool as used to prepare REED figures 5 and 7. Although the tool allows the nodes to have cross-referenced explanations, for brevity the narratives are suppressed here, despite being an important component of a complete Theory of Change.

An interesting use of the REED diagram / Theory of Change notes, supported by the tool automatically requiring all nodes to have notes, would be to help systematically document the NHS's response to the learning opportunities.

3.4 For the NHS and other organizations more generally

This paper raises many insights going beyond the specifics of the case study, raising serious concerns about widespread miscarriages of justice. There are general lessons about the lack of computer expertise available in the NHS (and likely in other organizations), which both led to the situation causing the problems, and led to the NHS mistakenly embarking on disciplinary and litigious actions as a result of its computer problems.

When a court case is supported by computer evidence, it is very likely that known and unknown problems go beyond the defendants and clinical or other specialities involved in the specific case.

Organizations need to vet and supervise everyone with access to IT systems, including visiting computer engineers and police.

A court case stops when it reaches a decision. The implication is that the organizations should not just read expert witness reports after a case has concluded, but be aware that the reports will have been written before all evidence was available. A case will complete or may be dismissed or collapse for very specific reasons, so wider lessons will be very hard to learn without expert support. Organizations should specifically ask experts to help explain issues that were *not* raised or resolved in court.

For example, the NHS may be aware of the Abbott engineer's corruption of data and that this constituted a problem for the case — although my reading of hospital committee minutes suggests they have no idea. They may not be aware that the lack of supervision of an engineer with unrestricted access to patient and staff data was problematic; the lack of recording what the engineer did was problematic; the reason why an engineer was called in was itself problematic; the reason why the database had deteriorated to such an extent an engineer was requested was problematic; the fact that a badly designed database had been procured in the first place was problematic. Unless the NHS recognizes these implicit problems it will not be able to avoid them in future.

IT systems should be continually monitored for cybersecurity breaches and other losses (and corruptions) of data.

Copies should be made of any evidence taken — in the Princess of Wales case, the evidence taken was in CSV format, which has no protection against accidental or deliberate editing, and there was no way to check the police evidence was the same as the original PrecisionWeb data.

Regular testing should be undertaken to check that IT systems are working as expected, and logs should be treated as controlled documents.

Expert witnesses need access to IT systems, their documentation, test procedures and logs, as well as their operators and managers (the Princess of Wales took the position that a criminal case was underway and no access was permitted).

3.5 For Abbott, manufacturers, and regulators

As this paper showed, disciplinaries, suspensions and prosecutions followed from the unfortunate actions of Mr. Reece, the engineer who deleted data and caused the impression that nurses had been negligent. Arguably, the database and glucometers should have been designed so that loss of data without any record would not have been possible. It appears that Abbott, the manufacturers, performed no investigation into the problems of the design of their systems or into the actions of their engineers. It follows that new devices and systems will not benefit from any learning from the case, and the weaknesses may be perpetuated in future designs.⁷ Similarly, the relevant UK regulator, the MHRA (Medicines & Healthcare products Regulatory Agency) showed no interest in the case, despite the fact that the systemic issues the case exposed about hospital devices and computer systems are endemic [17].

4 The future of REEDs

4.1 Why current REEDs are simple

REEDs link textual, narrative, evidence with an evidence pathway diagram. As currently implemented, a REED is defined using a very simple text file, as described in Appendix C. The definition file can be written in any order and split up without restriction; it can be edited with any text editor. This makes REEDs very simple to use and easy to learn, as there are no superfluous features to learn or which might have compatibility issues with other software.

On the other hand, most comparable systems provide many features and integrate editing, notetaking, diagramming, networking, and collaboration features — this makes them more powerful, but also makes for steeper learning curves, as well as providing lots of integrated features that have nothing to do with narrative evidence. Often, unlike current REEDs, the underlying data or database is proprietary not easily sharable. In other words, to get more features, these systems tend to lock in their users making it harder to share data and collaborate across and between organisations.

The point of making REEDs so simple is to help focus on how the ideas facilitate understanding complex evidence, investigating and critiquing it. More features could be a distraction.

4.2 Making REEDs easier to use

The idea of connecting diagrams to text, as REEDs as described in this paper, is not new. There are many notetaking and productivity tools such as Coda [3], Logseq [11], Notion [13], and Roam [14] that work a bit like REEDs. While these tools provide some of the benefits of Reliability of Electronic Evidence Diagrams, they focus more on:

Flexibility Rather than providing features specifically aimed at one task well (such as supporting evidence), notetaking tools focus on general purpose features that can do anything and therefore are weak on checking.

Ease of use These notetaking tools are integrated with user interfaces for multi-user notetaking. In contrast, REEDs are based on ordinary text files, and allow users to work with any editors.

Markdown These notetaking tools support stripped-down versions of HTML ('markdown') for formatting. In contrast, REEDs can use standard HTML or \LaTeX .

Communities Most notetaking tools are supported by large communities of developers, and therefore fixing bugs and adding new features is easier.

⁷I wrote to Abbott 28 April 2025 inviting comments on this article, and specifically on the statements in section 3.5, but have not (yet) had any reply. If Abbott have any reviews, I would welcome seeing them (even, if need be, under an NDA).

Wikis Many notetaking tools are based on wikis [1], including Coda, Notion and Roam (with Logseq supporting wiki features with a plugin).

Easy to use multi-user collaboration and interactive ways of editing documents are useful features the current paper has not addressed. Future work will therefore likely to combine the best from easy to use productivity tools and REEDs.

4.3 REEDs could be treated like car MOT certificates

In the UK it is illegal to drive a car without a MOT certificate (if it requires one). If a car is involved in an accident, insurance and courts take the absence of an MOT as contributory negligence and may increase penalties, it invalidates insurance, and it very likely reduces compensation (if any is due) to the owner and occupants of cars without a MOT.

Using the MOT as an analogy with evidence used in court proceedings depending on electronic evidence: if a court is not presented with a competent REED (which will support cross examination) then the presumption should be that the computer evidence is incomplete and unjustified — and likely misleading.

The Road Traffic Act 1956 introduced MOT certificates, which at first only covered brakes, lights, and steering. MOTs were optional until 1961. MOT certificates are now a legal requirement, and cover many points including tyres, emissions, seat belts, chassis, bodywork, wipers, driver's view of the road, and wiring, etc. Since 2018 drivers can face fines if their cars are not roadworthy regardless of possessing a valid in-date MOT certificate; put in other words, drivers can face fines if they would not be able to obtain a valid MOT given the current state of their car.

Like the history of car MOTs, it is proposed that REEDs should be introduced in a simple form, then increasingly tightened and improved with experience of their use.

Unlike MOTs, which are summary certificates that a car meets the legal requirements to be roadworthy, REEDs also enable the sources and management of evidence to be critiqued, and help parties achieve a common understanding of technical issues and weaknesses. However, if the MOT analogy is pursued further, REEDs are more like MOT *advisories* — they raise specific problems and concerns that need addressing. Indeed, an interesting analogy with MOT certificates is that in the same way that drivers can use their MOT advisories to help plan maintaining their vehicle, REEDs can be used routinely by organizations to manage and gain insights into the state of their IT systems.

New cars do not need MOT certificates because they are assumed to be well-designed, well engineered, and road worthy (in a process called homologation, where the design and manufacturing process themselves are certified). Likewise IT developers and vendors might like to provide REEDs (or a variation of REEDs) as evidence their products are reliable and easy to understand and hence help procurement processes. In fact, cars have a separate voluntary system of safety, performance and environmental ratings, which has contributed to a dramatic increase in car safety and reduced environmental impact. These voluntary rating systems have also helped consumers become better informed, which in turn has pressurized the market, and led to improvements in manufacturing. A similar approach might be used for computer systems.

Car design and development is a mature, well-regulated industry. In contrast, computer systems are neither mature nor well-regulated, so the MOT analogy has limitations. Outside of safety critical industries (such as nuclear power) there has been very little interest in, and even resistance to, assuring computer systems are dependable. Even for medical devices, the regulation is years behind any effective oversight of the growing complexities of medical computer systems, particularly — but not just — AI [17].

5 Conclusions

This article introduced REEDs, Reliability of Electronic Evidence Diagrams, as a practical way to explore and improve the details and reliability of computer evidence. A large NHS criminal case was used as a case study, showing that the technique scales up to real complex cases, and more generally appears to be of benefit to any investigations by facilitating better use of electronic evidence.

A REED consists of one or more evidence diagrams and associated narrative evidence. The REED diagrams in this paper were drawn using a prototype tool that draws the diagram and

manages the narrative using a simple text file format.⁸ The tool has three key advantages: it checks for errors (such as missing narrative sections); being purely textual it is very easy to email and edit and involve others to review diagrams and narratives; and because the diagram itself is drawn automatically, the tool means that no special drawing skills are required to use it.

It is arguable that had REEDs been used routinely by the Princess of Wales Hospital to support more critical operational IT management, then the IT problems causing this the would have been uncovered and addressed earlier, or the hospital's investigations would have uncovered and understood their IT problems and thus not progressed to suspensions and prosecutions.

Importantly, with the clarity of REEDs, the court case may have been resolved in a way that the clinical staff and leadership in the hospital understood their computer management problems so it would have taken appropriate corrective action, which, in the event, it did not. Section 3.3, below, shows that despite the findings of the court, the hospital still failed to make effective use of the insights, arguably because it couldn't properly understand the court's findings without first having the context of *critically* understanding its own computer system operations, which it did not have to start with.

The use of REEDs could be researched to establish how parties can best work using them, and to identify and evaluate possible ways to improve them (see Appendix A for alternative approaches). Standards could also be developed, as courts would find it helpful to be able to check REEDs are adequately prepared to conform to relevant standards.

Improving the quality of computer evidence and improving the ability to critically examine computer evidence will, in the longer run, help improve the quality of computer systems and computer system management.

6 Acknowledgements

Many thanks to Kirsty Brimelow KC, Leo Freitas, Peter Ladkin, Stephen Mason, and Prue Thimbleby.

⁸Figures 2, 3, and 4 were drawn as freehand diagrams in 2014 using Apple Keynote, a program similar to Microsoft PowerPoint. All other diagrams in this article were drawn with the prototype REED tool.

7 References

- [1] T B Chatfield, *The Complete Guide to Wikis How to Set Up, Use, and Benefit from Wikis for Teachers, Business Professionals, Families, and Friends*, Atlantic Publishing Group, 2009.
- [2] J Christie, "The Post Office Horizon IT scandal and the presumption of the dependability of computer evidence," *Digital Evidence and Electronic Signature Law Review*, 17:49–70, 2020. <https://journals.sas.ac.uk/deeslr/article/view/5226/5073>
- [3] Coda: *Your all-in-one collaborative workspace*, 2025. <https://coda.io>
- [4] HHJ Crowther QC, "CASE RULING: ENGLAND & WALES Case citation: R v Cahill; R v Pugh. Ruling 14 October 2014, Crown Court at Cardiff." Case numbers: T20141094 and T20141061, *Digital Evidence and Electronic Signature Law Review*, 14:67–71, 2017. <https://doi.org/10.14296/deeslr.v14i0.2541>
- [5] Department for Environment Food & Rural Affairs, Defra Theory of Change (ToC) Tool, 2021. <https://randd.defra.gov.uk/ProjectDetails?ProjectId=20910>
- [6] A Gawande, *The Checklist Manifesto: How to Get Things Right*, Profile, 2011.
- [7] C Haddon-Cave, *The Nimrod Review — An independent review into the broader issues surrounding the loss of the RAF Nimrod MR2 Aircraft XV230 in Afghanistan in 2006*, House of Commons, UK Government, 2009. <https://www.gov.uk/government/publications/the-nimrod-review>
- [8] A Hopkins, *Commissioned Review, June to September 2016. Review of the Blood Glucometry Investigations in Abertawe Bro Morgannwg University Health Board. Establishing lessons learned*, 2016. Archived at <https://www.harold.thimbleby.net/glucometry-records/Hopkins-report.pdf>
- [9] PB Ladkin, S Mason & H Thimbleby, "Misunderstanding Digital Computer Technology in Court: A Commentary on a Case Involving the Post Office Horizon System," *Digital Evidence and Electronic Signature Law Review*, in press.
- [10] D Lawton, R Stacey & G Dodd, *eDiscovery in digital forensic investigations*, Centre for Applied Science and Technology (CAST) Publication 32/14, Home Office, 2014.
- [11] Logseq, *Connect your notes, increase understanding*, 2025. <https://logseq.com>
- [12] S Mason & D Seng, eds, *Electronic Evidence and Electronic Signatures*, 5th ed (see in particular, Chapter 5, The presumption that computers are 'reliable'). Institute of Advanced Legal Studies for the SAS Humanities Digital Library, School of Advanced Study, University of London, 2021. Open source <https://uolpress.co.uk/book/electronic-evidence-and-electronic-signatures>
- [13] Notion: *Notion for Notes & Docs*, 2025. <https://www.notion.com/notes>
- [14] Roam: *A note-taking tool for networked thought*, 2025. <https://roamresearch.com>
- [15] I Sommerville, *Software Engineering*, Pearson Education, 10th Revised edition, 2017.
- [16] H Thimbleby, "Misunderstanding IT: Hospital cybersecurity problems in court," *Digital Evidence and Electronic Signature Law Review*, 15:11–32, 2018. <https://journals.sas.ac.uk/deeslr/article/view/4891/4841>
- [17] H Thimbleby, *Fix IT: See and solve the problems of digital healthcare*, Oxford University Press, 2021.
- [18] H Thimbleby, *Archived Princess of Wales glucometer documents and notes*, 2024. <https://www.harold.thimbleby.net/glucometry-records>
- [19] N Wallis, *The Great Post Office Scandal: The extraordinary story behind the major ITV drama: The fight to expose a multimillion pound IT disaster which put innocent people in jail*, Bath Publishing Limited, 2021.
- [20] H Woolf, *Final Report to the Lord Chancellor on the Civil Justice System in England and Wales (Access to Justice)*, Stationery Office Books, 1996.

- [21] B Wright, The Press Association, "Nurses accused of neglecting patients at Princess of Wales cleared as case collapses," *Wales Online*, 2015. <https://www.walesonline.co.uk/news/wales-news/nurses-accused-neglecting-patients-princess-10258426>

— ONLINE APPENDICES —

All 7 pages of appendix material suggested to be online subsidiary material.

A Alternative notations for REEDs

Further research is called to assess, develop, and find better notations and approaches to REEDs. Alternative candidates, which may be found more helpful in specific cases, include:

- There are many easy-to-use drawing tools (including Apple’s Keynote and Microsoft’s PowerPoint) that can be adopted to help draw REEDs and write the narrative evidence. However such tools provide no checks and no automatic cross-referencing.
- There are many tools used in eDiscovery, some of which have powerful graphical facilities [10].
- Safety assurance diagrams [23], such as Adelard’s ASCE (the Assurance and Safety Case Environment) [22].
- Theory of Change diagrams [5] (see sections 1.3 and 3.3).
- Unified modeling language, UML [26].
- Why-Because Analysis, WBA, and Why–Because Graphs, WBG [27].
- ... and integrated techniques for rigorously developing software and documentation together, such as DRisQ’s System Kapture [25], and the National Security Agency’s/Praxis’s Tokeneer [24].

All the above notations come with powerful tools.

B Semantics of REEDs

A REED has a title, set of authors, date, and version, and has as contents an annotated annotated directed acyclic graph (DAG).

Currently, a REED is represented using a simple textual representation (see section C below), which for convenience may be split into any number of files, which, for instance, may be managed by different authors in different places.

The key part of a REED are the annotated DAGs with at least one node. Each node in a DAG has a unique label referring to a narrative description, a summary of the evidence and its evidential issues and problems (if any). Arrows can be optionally labelled and similarly provided with a narrative description. Nodes and arrows can be labelled with a highlight color, where each color has an associated narrative text.

The REED need not be a connected graph, but if so each connected subgraph is a REED in its own right. The REED tool reports if the REED DAG is not connected, and lists its components (independent REED diagrams not connected to each other with arrows, regardless of the directions of the arrows; i.e., the DAG’s weakly connected subgraphs).

In practice the graph is presented as an annotated arrow /node diagram, with the summary of evidence associated with each node presented in a conventional textual document, with section numbers cross-referencing the node labels. Error messages and other additional information is also inserted in the document at appropriate places.

An arrow $u \rightarrow v$, which can also be written $v \leftarrow u$ and spoken as “ v depends on u ,” denotes

$$\text{problems}(u) \implies \text{problems}(v)$$

Put in words: if there are or are anticipated to be problems with the evidence represented by node u then there are or are likely to be problems with the evidence represented by node v . The probity of v relies on the probity of u .

Each node is required to have a narrative description which explains the known or speculated problems (if any). Additionally REED nodes can optionally be highlighted with colors. Each color is required to have a narrative description to define or explain the highlighting.

The REED tool provides a number of features. For instance, a practical feature is the tool allows users to use node abbreviations that are easier to write and more convenient than the full node labels themselves: each label only needs to be written once in full; however, the full labels are used in all output generated by the tool, including in all diagrams it generates.

Since a REED is a directed acyclic graph the tool checks that there are no cycles (such as $u \rightarrow v \rightarrow u$). Another check is that a node can only have more than one narrative description if each description is labelled with a different set of authors.

Since the \rightarrow arrow is transitive (so, for example, $u \rightarrow v$ and $v \rightarrow w$ implies $u \rightarrow w$) the REED tool optionally allows color highlights to ‘cascade,’ which automatically makes the coloring apply to every node following the path of transitive arrows. (It is an error if two or more such paths share nodes.)

In general, a series of evidence dependencies may get quite complex, so the REED tool allows the standard notation $v \rightarrow^* w$ to check that there is a connected path of arrows from u to w .

Thus, a REED author may write Smith \rightarrow^* Database before finishing writing the narrative for Database in order to check that the Database node really depends, perhaps in a complicated way, on what Smith did or put in evidence. The REED tool will then report an error if there is no path Smith $\rightarrow \dots \rightarrow$ Database.⁹ This is a particularly useful check if a REED document is being written by several collaborating authors, as (for instance) one author may accidentally delete or redirect an arrow that another author is implicitly relying on.

The REED tool will generate diagrams and narrative files even for faulty REED descriptions. This allows teams to develop REEDs, find out what is problematic with them (whether as reported by the tool or by co-authors’ criticisms), and iteratively improve them. The tool also generates digital signatures, so that (if they wish) authors can ensure a REED is not further modified after they have edited it.

C A tool for REEDs

C.1 Key benefits of a tool

The prototype tool creates a diagram (as a PDF file) and generates a typeset report of the evidence cross-referenced back to the diagram. No drawing skills are required as the tool uses a simple text file to define the REED. The text file can be emailed directly between parties and edited with familiar text editing tools. The text file can be split up and written in any order, which facilitates collaboration between many authors. The reports and checking features the tool provides further encourage and support constructive collaboration.

An important benefit of using a computer tool is that new ideas or features for improving REEDs or the processes to support their use can be implemented in the tool, and hence correct use can be audited and checked, for instance to check for problems such as omitting narrative for a node, or someone else providing narrative for a node that has not yet been drawn in the diagram. The tool approach therefore encourages parties to discuss computer evidence without worrying that they may lose track of details.

The tool finds a good balance between being straightforward and easy to use, with providing substantial power to help produce high quality REEDs. The tool provides many helpful diagnostics, such as warning that node narratives have not been provided or that highlight colours have been used but their meaning has not been defined.

As well as the REED diagrams like figures 5 and 7, the tool can of course easily draw other types of diagram. Figure 1 and the Theory of Change diagram (figure 8), and all the diagrams in this Appendix, were also drawn using the prototype tool.

C.2 REED identification

Some information must be added to identify each REED diagram.

The title, version and date can be added. Examples are reproduced in the REED diagrams shown in figures 5 and 7.

An abstract and a list of REED authors can also be added.

⁹If the tool determines there is no evidential pathway Smith \rightarrow^* Database it will also check Database \rightarrow^* Smith, in case the arrow had been accidentally written the wrong way round.

C.3 Drawing diagrams

A simple textual language is used to specify the diagram and to annotate it.

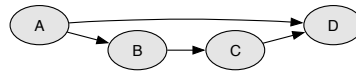
A node is represented by an identifier, consisting of letters, digits and underline. Identifiers like 18 or Post_Office_Computer2 are valid examples.

An arrow is represented by `->`, `<-` or `<->`. So, for example the (perhaps artificially concise) statement

```
B -> C -> D <- A -> B
```

defines a diagram where B has an arrow to C, C has an arrow to D, and A has arrows to both B and D.

The diagram so defined will look like this, before any numbering, shapes, styles, colours or highlighting have been added to the nodes:¹⁰

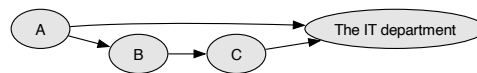


Node identifiers are supposed to be clear and simple, but completely arbitrary identifiers can be placed between quote marks, such as "A long identifier consisting of several words and arbitrary symbols like £s and \$s."

In general, there is a balance between convenient, easy-to-use node names and longer descriptive names to use in a diagram. Any node can therefore be given a longer name using the command `is`, as in

```
D is "The IT department"
```


With this addition, the previous diagram will now update to look like this, below, with D displayed in full as "The IT department":



In this example, the simple D alone can continue to be used to conveniently define arrows and keep track of the structure of the evidence, but the longer explanatory name *The IT Department* will be used in all diagrams and narratives that everyone sees.

The advantage of this approach is that the full name can be changed at any time in just one place, without needing to revise any of the rest of the diagram — the full name will update everywhere it is needed, but (in this case) D can continue to be used everywhere to help specify arrows and narrative notes without needing any further changes.

C.4 Highlighting nodes

Nodes can be visually flagged, so they are highlighted in the diagram *and* so that a flag like  will automatically appear against each highlighted node in the narrative, as illustrated in section 2.5 in the body of the article.

For example the definition,

```
highlight D
```

highlights the node D.

Highlights default to the colour red, but they can be explicitly coloured in other ways to help authors more easily manage their collaboration by drawing attention to areas that are of specific concern or need further work, etc.

```
highlight D is blue
```

Colour schemes like de Bono's Six Thinking Hats can be used; de Bono's scheme keeps the number of colours manageable and uses pre-defined colours to stimulate thinking and conversation — which is the point of REEDs [28]. However, since highlight colours can be used for any purpose, it is important to define what colours are intended to mean. Hence colours, like nodes, can be given longer identifiers:

¹⁰The tool allows diagrams to be drawn in any of four standard directions, left to right, right to left, top to bottom and bottom to top. Unlike the examples in the body of the article, the examples in the appendices are drawn left to right to save space.

```
highlight blue is
```

```
"Some serious problem in the evidence that needs addressing."
```

All such colour definitions are summarized in a key legend at the start of the narrative REED evidence.

If a colour has been given a meaning (e.g., that something needs addressing, as in the definition of the blue highlight above) but this colour has not been used anywhere in the REED, this will be noted in the narrative to avoid any confusion (or to indicate that possibly the wrong colour was defined). Furthermore, when REED files are combined, the tool checks that colours are given consistent meanings.

Highlights can be set to 'cascade' marking every node and arrow not otherwise coloured that they are connected to in the appropriate arrow direction. For example,

```
highlight blue cascade
```

```
is "Critical new evidence introduced for version 3."
```

was used for figure 7, and made any blue highlighting cascade along arrows: the engineer's node was the only one explicitly highlighted blue, but the blue automatically cascaded from the engineer's node along arrows across the REED to every affected node.

Cascading can be limited to a particular node, as in:

```
highlight D cascade is red
```

Here, the cascading will start from node D alone, but the cascading will not affect any other red highlighting.

For accessibility reasons highlight colours can be set to be spelled out in words, and font colours to be changed automatically to contrast with the highlighting colour (e.g., automatically changing to white text for any black highlight).

C.5 Strings

As described above, node names and arrow names cannot contain spaces. Strings allow names and narratives to be arbitrary text.

There are two types of string: *quoted strings*, and *herestrings*.

Quoted strings are written between " marks. Quoted strings follow backslash conventions. Thus between quote marks, writing \" means a ", when otherwise a single " would finish the string; writing \n means a newline; and, in particular, writing \\ represents a single backslash.

Herestrings are a popular way to handle long strings, without relying on backslash conventions.

The herestring notation provided by the tool is that anything between <<< text and a line, apart from spaces, containing the same text text (whatever text is) is the string, regardless of any backslashes and quote marks inside it.

Herestrings are useful for L^AT_EX text, which contains lots of \ characters, which would otherwise have to be doubled-up and written in quoted "-strings as \\, which soon gets unreadable.

C.6 Notes

The tool will automatically generate unique node references (typically numeric references), regardless of the names used in the specification, so that the diagram and narrative can always be easily cross-referenced. The evidence narrative is split up into sections, each associated with a node by writing

```
note node-or-arrow narrative-text ...
```

However, it is usually clearer to combine both note and is functions together, as in this example

```
note 17 is "The IT Department" <<< ENDNOTE
```

```
This note will be a full description of the IT
Department's impact on the evidence. The text can
span several lines or paragraphs as necessary.
```

```
We are using 'herestrings' (as described above in the
section on strings) so this string ends immediately
before the next line because it only says ENDNOTE:-
```

```
ENDNOTE
```

Writing all the evidence out like this for each node one after another in the REED specification is intended to look and read like section names and section texts in a conventional text document.

Arrows, too, can have notes and be named in full using is:

```
note Nurse -> Notes is Written
  "Nurses write notes that are signed off and dated."
```

will ensure the arrow from the node Nurse to the node Notes will be annotated “Written” and written up in the narrative as “Nurses write notes that are signed off and dated.”

```
note 17 is "The IT Department" author "Harold Thimbleby"
<<< ****
  Prof Thimbleby thinks ...
  ****
note 17 is "The IT Department" author "Phillip Starling"
<<< ****
  Mr Starling thinks ...
  ****
```

Here, matching **** markers were used to highlight the start and end of the note’s herestring text. It does not matter if a note has its name defined more than once, as here, so long as the definitions are the same.

C.6.1 Cross-referencing nodes and evidence

Within the narrative text, nodes can be cross-referenced by writing their identifier inside double angle brackets, as in `[[[NodeID]]]`, which notation will be replaced by the node’s reference number and full name. For example, writing

See further details discussed under node `[[[Abbott_engineer]]]`.

will be replaced by

See further details discussed under node v3-3.0, Unsupervised Abbott engineer.

— assuming v3-3.0 is the REED’s reference to the node with ID ‘Abbott_engineer’ in the diagram (node references like “v3-3.0” give the REED version, dash, then count nodes drawn from the left, dot, then count down in rows from the top of the diagram).

In HTML, this will be a hyperlink, so if you click on it, your reading position will jump to the narrative notes for node v3-3.0, or whatever the node link refers to.

In \LaTeX the notation `[[[NodeID]]]` just generates literal text, exactly as illustrated above. However, since arbitrary \LaTeX documents may need to cross reference nodes (this paper being a case in point), the tool also generates an aux file which can be input into \LaTeX , and then writing `\ref{node-id}` will provide a cross-reference from the ID to the node’s reference (such as v3-3.0), and `\ref{node-id-is}` will provide the full name (such as Unsupervised Abbott engineer) — in other words, this feature introduces a set of cross references to the narrative evidence into a \LaTeX document that work in the normal way. Hence with this feature, \LaTeX documents can be written and reliably refer to the narrative evidence without worrying about keeping track of node versions, reference numbers, and name changes as the REED itself is edited. (\LaTeX provides error messages if node names are undefined or misspelled.) This cross-referencing feature was used several times in this section, but it has had more substantial uses throughout this paper.

C.6.2 Combining HTML and \LaTeX notes

The tool can convert a single REED into both HTML files (creating interactive web documents) and \LaTeX files (creating high quality typeset documents).

Narrative text can be written in either HTML or \LaTeX , and the tool translates to the target file format, whether to HTML or \LaTeX . For example, if the author writes the HTML `&#`; it will be unchanged in an HTML file output, but will be translated to `\&` if output in a \LaTeX file.

HTML and \LaTeX can both be used in complicated ways, potentially going beyond the capabilities of the tool to translate accurately. To solve this problem:

- Any text written after `<html>` is considered to be exact HTML and will not be used in any form for the \LaTeX output;
- Any text after `<latex>` is considered to be exact \LaTeX and is only used as \LaTeX and will never be used in HTML output;
- Any text after `<both>` is considered to be basic text that is simple enough to be treated (with minor automatic edits) as either HTML or \LaTeX as required.
- Text is treated as if it starts with `<both>`.

This example makes the idea clearer:

```
Anything written here appears in both LaTeX and HTML files,
with direct translations like ---
can be used directly in LaTeX but is automatically translated
to &mdash; for HTML, equally &mdash; can be used directly in HTML
but is automatically translated to --- for LaTeX files.
<html> anything written here only appears in HTML files
<latex> anything written here only appears in LaTeX files
<both> anything written here appears in both LaTeX or HTML files.
```

Complex HTML or \LaTeX , say like HTML's `<img...>`, is reported as an error if the tool does not know how to process it. Instead, images would be written something like this, clearly separating the different HTML and \LaTeX code:

```
<html> 
<latex> \includegraphics[width=2in]{photo.jpg}
<both>
```

The approach allows the author to specify exact meanings of their text in either HTML or \LaTeX , as required.

The following table illustrates some specific translations that are made in each direction:

Print	HTML	\LaTeX	Print	HTML	\LaTeX
&	&	\&	%	%	\%
'	‘	'	'	’	'
"	“	"	"	”	"
—	—	--	—	—	---
		\@	\$	\$	\\$
<i>nonbreaking space</i>	 	~	£	£	\pounds
<i>paragraph</i>	<p/>	blank line	#	#	\#

The special case `[[[id]]]` (explained in section C.6.1) is translated into HTML as `...` and translated into \LaTeX as `\ref{id}`.

The tool reports anything `<unrecognized>`, `&unrecognized;` or `\unrecognized` so you can decide how to translate it manually.

C.7 Using versions to override properties

Often REEDs are needed in several versions, as for example this article used two REED versions as illustrated in figure 5 (version 2), and in figure 7 (version 3). However, generally, most of a new version REED will be the same as the old version: in this article, the only significant change from version 2 to version 3 was adding a single node for the Abbott engineer plus the narrative notes for that new node — everything else remained exactly the same.

Normally the tool will report an error if there is any attempt to redefine any REED property such as the title, date, or version number. For example, the following would be reported as an error because a REED cannot be both version 1 and version 2:

```
version v1
...
version v2
...
```


Often, however, it is useful to have one version be a small variation of another, like version 3 in this article was version 2 with only a single node added — the rest of version 3 was the same as version 2. In this sort of case, to make things more flexible, the tool allows redefinitions to override old definitions, as follows:

```
version "v2"  — A normal definition must be unique
override version "v3"  — An overriding definition,
                        here, all following material defines for v3
```

With this feature, it is possible to combine several REEDs making incremental changes, such as adding a few new nodes, but leaving the rest of the narrative evidence unchanged.

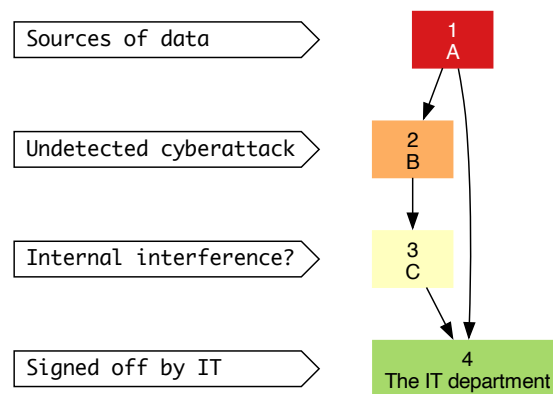
The tool provides a useful feature: it can be instructed to select particular versions, hence allowing a single file to specify any of a collection of REED versions as specifically requested.

C.8 Summary of further features

Nodes and arrows can be grouped, shaped, and coloured and styled freely, and the specification can provide the title, version, authors, date, and an abstract, which are then automatically passed over to the diagrams and narrative documents the tool generates.

Nodes can be optionally numbered, either in a single sequence (e.g., 1, 2, 3, ...) or using row/column numbering (e.g., 1.1, 1.2, 1.3, 2.1, 2.2, ...), as illustrated in figure 5. Nodes can also have arbitrary references instead of automatic numbering.

The tool has many features that are not discussed here, for instance to automatically colour and lay out diagrams. Here is the example diagram from above, drawn and laid out using the tool's standard Theory of Change style for default colouring and layout, plus a simple number sequence generated for nodes and simple explanatory texts added to each row of the diagram:



C.9 Command-line features of the prototype REED tool

The prototype tool reads an ASCII text description of a REED (possibly combining multiple files and versions), and outputs several files:

- A PDF file representing the graph diagram.
- An HTML file representing the narrative document, including the graph diagram.
- A \LaTeX file representing the narrative document, including the graph diagram.
- The REED represented in various formats to support collaboration or analysis: GraphViz, JSON, Mathematica, and XML.
- Digital signatures (currently MD5) for the documents.

The tool is currently written in C and runs on a Unix command line. In the long run, it is envisaged that the tool would be reimplemented as a conventional graphical application so that it looks more like, say, PowerPoint with handouts, rather than as a textual language that for some may have a steep learning curve; however, the current approach will be no problem for expert witnesses with a minimal IT background.

The features described in this Appendix are subject to change; indeed a streamlined version is being written in Rust which has minor differences.

D Additional appendix references

- [22] Adelard, “ASCE – the Assurance and Safety Case Environment,” 2025.
<https://www.adelard.com/asce>
- [23] The Assurance Case Working Group (ACWG), *Goal Structuring Notation Community Standard*, Version 2, **SCSC-141B**, 2018. <https://scsc.uk/r141B:1?t=1>
- [24] M Cristiá & G Rossi, “An Automatically Verified Prototype of the Tokeneer ID Station Specification,” <https://arxiv.org/abs/2009.00999>, 2020.
- [25] D-RisQ, *Kapture*, 2025. <https://www.drisq.com/kapture-more-about-information>
- [26] ISO/IEC 19501:2005 – Information technology – Open Distributed Processing – Unified Modeling Language (UML), 2005. <https://www.iso.org/standard/32620.html>
- [27] PB Ladkin & K Loer, “Analysing Aviation Accidents Using WB-Analysis — an Application of Multimodal Reasoning,” Spring Symposion. pp169–174, AAAI Tech Report SS-98-04, Association for the Advancement of Artificial Intelligence, 1998.
https://www.researchgate.net/publication/2549709_Analysing_Aviation_Accidents_Using_WB-Analysis_-_an_Application_Of_Multimodal_Reasoning
- [28] H Thimbleby, “Designing Interfaces for Problem Solving, with Application to Hypertext and Creative Writing,” *AI & Society*, 8:29–44, 1994. DOI 10.1007/BF02065176