Dependability and Trust in Organisational and Domestic Computer Systems

Ian Sommerville, Guy Dewsbury, Karen Clarke, Mark Rouncefield

Computing Department, Lancaster University.

Our economy and national infrastructures are dependent on a range of socio-technical systems and, by and large, these systems can be trusted to provide a dependable service. For example, electricity and telecommunication systems are generally reliable, the bank ATM network can usually deliver cash to authorised customers and automated stock control systems have meant that large stores and supermarkets rarely run out of specific products.

In essence, at least in Western societies, the vast majority of people trust the services that are provided through the physical and economic infrastructure. This trust is engendered because, these services almost always meet the expectations of their external users. In order to meet these expectations, complex socio-technical systems have to be put in place by the service providers and these now, universally, rely on computer-based information systems. These information systems are essential elements of the socio-technical systems so both the organisations running these systems and the system users depend on them.

The information systems that support the socio-technical systems that run the national and business infrastructure have two important characteristics:

1. They are situated in organisations (banks, telephone companies, electricity generators) that have a history of service provision and that have well-established processes for managing the delivery of these services. External users of organisational systems trust these organisations to use their best endeavours to ensure that their computer systems deliver correct information. Furthermore, it can be assumed that the people in these organisations follow the defined operational processes when it is appropriate to do so and react in a contingent way when they are faced with exceptional situations not covered by these processes.
2. They are essential for the effective provision of organisational services and the people within the organisation who are involved in the process do not have the authority to decide whether or not the automated systems should be used. It can be assumed that the operators have received some training in the use of the software and also that, whatever

> the flaws in the software system, they do not have the discretion to simply discard that system and replace it with an alternative.

Organisational systems are designed for a specific purpose, support known and defined processes and their use is controlled by the organisation. In this context, when we consider the issue of what is meant by a 'trusted' computer system, we argue that a technical view of trust is appropriate. A system is trusted if it correctly provides the services that it has been designed to deliver and is available for service when required. Because both the operators and the computer system are within the organisation then issues such as the provenance of the system are disregarded in assessing its trustworthiness. Furthermore, as far as external users of the system are concerned, their access is mediated by a human operator so there is no direct trust relationship between the external user and the computer system.

Therefore, for systems that have a clear role in organisational socio-technical processes, the primary trust relationship is between the operator and the computer system and the dominant factor in that trust is the *dependability* of the system. We discuss the notion of dependability in the following section but, essentially, you can think of it as an amalgam of other system properties such as system availability, security, reliability, etc.

More broadly, however, when we consider socio-technical systems that are not entirely situated within an organisation then trust is, of course, far more than a technical issue. It reflects the user's confidence that the system will do what they want (whether or not this has been specified by the system designers) and that it will not cause damage that results in losses of time, information, money, etc. to the user.

The degree of trust that an external user has in a system depends on factors such as previous experience with comparable systems, the provider's reputation, the existence of external sanctions on the system provider if they fail to deliver services and the price paid. It also reflects the degree of risk taken by the user in that people are more willing to trust a system where the exposure to loss is relatively low and legal factors such as the existence of regulators and compensation bodies.

We see examples of this when organisational systems are Internet-enabled for external users. People have few problems trusting information-giving systems such as timetables and catalogues (low risk) but are more wary of systems where there is potential financial loss. Many people are still reluctant to use Internet banking, even although the technological safeguards are, if anything, stronger than in traditional banking systems. It is noticeable that many new entrants to banking enabled by the Internet have not been successful. Rather, users have preferred known banks because of their reputation. Here trust is clearly engendered by a known brand rather than any technical characteristics of the bank's information systems.

In this chapter, we will not be concerned with these broader issues of trust but, rather, will focus on trust from a technical perspective. However, we will argue that, for systems where the use of defined operational processes cannot be guaranteed or where users can choose whether or not to use the system, there is a need to extend the technical view of dependability to cover broader issues of fitness for purpose and adaptability as well as more traditional properties such as system reliability and availability.

The remainder of the chapter therefore includes four principal sections. Firstly, we discuss the currently accepted technical model of system dependability as applied to organisational systems. We then go on to critique this model and propose a broader model of system dependability that incorporates this model but which extends it to be applicable to domestic and discretionary systems - workplace systems where users have a choice whether or not to make use of them. Finally, we propose ways in which this model may be used in the design process for domestic and discretionary systems,

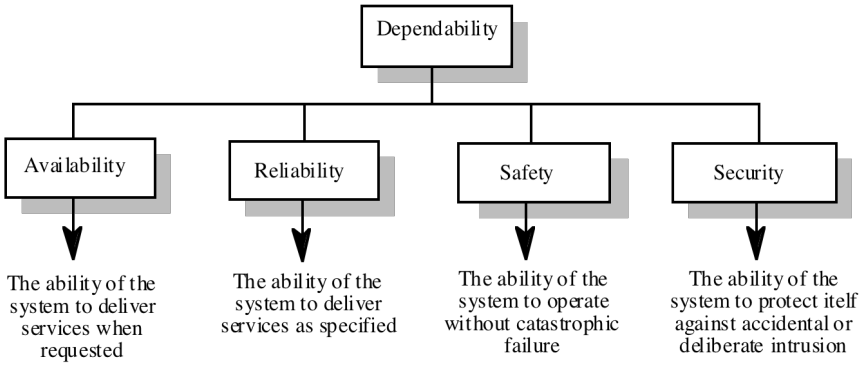
1. DEPENDABILITY – A TECHNICAL PERSPECTIVE

> ***Dependability*** *is defined as that property of a computer system such that reliance can justifiably be placed on the service it delivers. The service delivered by a system is its behaviour as it is perceptible by its user(s); a user is another system (human or physical) which interacts with the former[1].*

The dependability of a computer system is a property of the system that equates to its trustworthiness. Trustworthiness essentially means the degree of user confidence that the system will operate as they expect and that the system will not 'fail' in normal use. A trustworthy system has the potential to be trusted by a user although other factors such as previous experience and the provenance of the system influence whether or not users actually trust the system. As discussed in the introduction, we believe that dependability is by far the dominant factor in influencing whether or not organisational systems are trusted by their users.

Dependability is not a simple, measurable system property but, rather, is a complex property that reflects the fact that simpler properties are inextricably intertwined; it rarely makes sense to consider them in isolation. Figure 1 [2] shows the principal properties that contribute to system dependability:

1. *Availability* The availability of a system is the probability that it will be up and running and able to deliver useful services at any given time.
2. *Reliability* The reliability of a system is the probability, over a given period of time, that the system will correctly deliver services as expected by the user.



1.1 Figure 1: Dependability attributes

3. *Safety* The safety of a system is a judgement of how likely it is that the system will cause damage to people or its environment.
4. *Security* The security of a system is a judgement of how likely it is that the system can resist accidental or deliberate intrusions.

These properties themselves can be decomposed into simpler system properties. For example, security includes integrity (ensuring that the systems program and data are not damaged) and confidentiality (ensuring that information can only be accessed by people who are authorised). Reliability includes correctness (ensuring the system services are as specified), precision (ensuring information is delivered at an appropriate level of detail) and timeliness (ensuring that information is delivered at the time when it is required).

The principal dependability properties of availability, security, reliability and safety are clearly inter-related. For example, the safe operation of a system usually depends on availability (is the system up and running) and reliability (is the system delivering services as specified). A system may become unavailable because security failings allow external denial of service attacks. If a system that has been demonstrated to be safe is infected with a virus then the system itself has been corrupted; safe operation can no longer be assumed.

As well as these 4 principal dimensions of dependability, other system properties are also sometimes considered under the heading of dependability. These include:

1. *Repairability* System failures are inevitable but the disruption caused by failure can be minimised if the system can be repaired as quickly as possible. If a system is to be repairable, it must be possible to diagnose

the problem, access the component that has failed and make changes to fix that component.

2. *Maintainability* As systems are used, new requirements emerge and it is important to maintain the usefulness of a system by changing it to accommodate these new requirements. Maintainable software is software that can be adapted economically to cope with new requirements and where there is a low probability that making changes will introduce new errors into the system.

3. *Survivability* A very important attribute for Internet-based systems is survivability which is closely related to security and availability [3]. Survivability is the ability of a system to continue to deliver service whilst it is under attack and, potentially, while part of the system is disabled.

4. *Error tolerance* This property could be considered as part of usability and reflects the extent to which the system has been designed so that user input error are avoided and tolerated. When user errors occur, the system should, as far as possible, detect these errors and either fix them automatically or request the user to re-input their data

The type of system and its context of use determines which of these dependability properties are most important. For a system controlling a car engine (say), safety and reliability considerations are significant but security is less important because there is no external access to this system. For an e-commerce system, availability and security are usually the most important properties.

Laprie, a leading researcher in system dependability, proposes that, for critical systems used in organisations, the key dependability properties are availability, reliability, safety, confidentiality, integrity and maintainability [1]. He relates these to the system behaviour as seen by an external system user:

> *"the readiness for usage leads to availability, the continuity of service leads to reliability, the non-occurrence of catastrophic consequences on the environment leads to safety, the non-occurrence of unauthorized disclosure of information leads to confidentiality, the non-occurrence of improper alterations of information leads to integrity, the ability to undergo repairs and evolutions leads to maintainability."*

These dependability attributes are one component of Laprie's dependability tree where, as well as dependability attributes, he identifies the means to achieve dependability and the impairments to dependability. This dependability tree is shown in Figure 2.

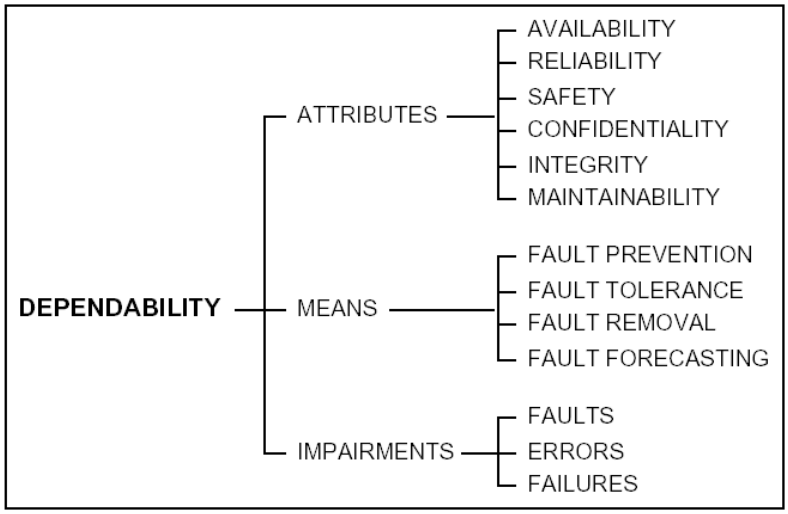


Figure 2: Laprie's dependability tree

Randell [4] expands on Laprie's notions of means and impairments in a discussion of faults, errors and failures. Essentially, these terms can be defined as:

1. *Faults.* A fault is deemed to be the cause of an error in a system. For example, if a variable in a program has been wrongly set up (say as 1 rather than 0) then this is a fault. Faults, however, need not manifest themselves every time that a program executes – indeed, they may never manifest themselves as, in many programs, sections of code are included to cope with situations that never arise.
2. *Errors.* An error is defined to be an unexpected or unwanted system state. That is, using the above example, when the faulty statement is executed then a part of the system state has a value of 1 rather than the expected value of 0. The fault is the latent condition; the error is its manifestation when the system is in operation.
3. *Failures.* A failure is an external manifestation of an error when some system service behaves in an unexpected way. For example, if the service is to add numbers input by the user but the initial value of the sum is 1 rather than 0 then the final result will be incorrect.

Laprie and Randell have focused on the dependability of critical control and protection systems in their work. Consequently, their views on dependability are influenced the nature of these systems. Their definitions of impairments to dependability embed a number of assumptions:

- That system failures (defined by Laprie as a deviation from fulfilling the system function) can be recognised when they occur. In a control system, this might be because sensors indicate that a

controlled variable is changing in an unexpected way or, sometimes, systems simply terminate execution unexpectedly.

- That errors (defined by Laprie as 'that part of the system state which is liable to lead to a subsequent failure') can be detected by an external observer who has access to information about the system. For example, system logs may show that a program variable has an unexpected value of –10 rather than +10.
- That errors arise inevitably from faults (the hypothesised cause of an error). For example, a system fault may be the omission of code to check that an operator input is not negative. Faults in programs are assumed to arise because there has been a failure in the development system. For example, the software testing process may never have checked the system's response to incorrect operator inputs.

Randell proposes that this technical fault-error-failure model of dependability can be applied to the development system for software as well as to the software itself. This leads to a conceptually attractive failure-fault dependency in different systems as shown in Figure 3. Failures in one system inevitably lead to faults in another system that may then manifest themselves as failures.

However, while the failure-fault cascade is certainly valid within computer systems where a failure in a sub-system can lead to a fault in an encompassing system, we are unconvinced that it applies equally to socio-technical systems, such as systems used for software development. The scheme shown in Figure 3 is conceptually attractive but we believe that the reasons why faults are introduced into software systems are more complex than the model implies. We return to this discussion in the following section.

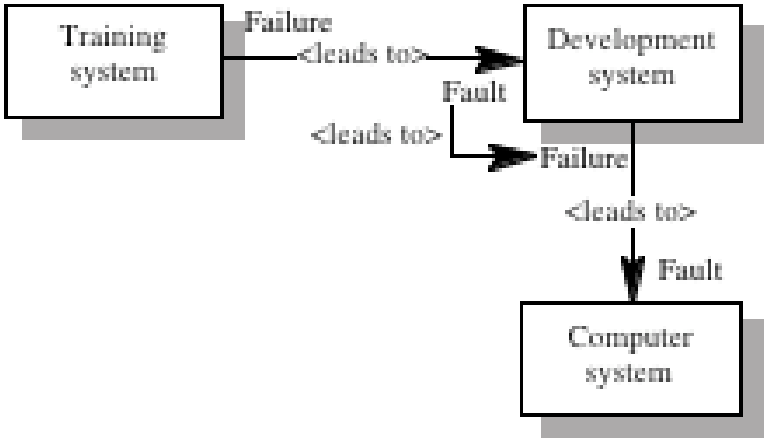


Figure 3 System dependencies

Finally, from Laprie's dependability tree, the means are the ways in which the developers of a computer system can achieve dependability. These are:

1. Fault prevention – ensuring that faults are not introduced into a system.
2. Fault tolerance – designing the system in such a way that it can continue in operation in spite of the occurrence of faults.
3. Fault removal – reducing the number or the seriousness of faults before the system is deployed.
4. Fault forecasting – estimating the number, incidence and consequences of faults.

Fault prevention can be achieved through the use of development techniques and tools that identify potential faults at an early stage in the development process or, more simply, by excluding approaches to development that are known to be likely to lead to faults. For example, modern programming languages such as Java do not allow the use of pointers – a programming construct that is notoriously error-prone. Consequently, a large class of faults resulting from mis-oriented pointers simply cannot occur.

Fault tolerance can be achieved in programs by making use of diversity and redundancy. An approach that is used in some critical systems (such as the flight control system in some models of the Airbus aircraft) is based on multi-version programming where several versions of critical systems components are developed by different teams [5-7]. There is an assumption made that the teams are unlikely to make the same mistakes. A checking mechanism is embedded in the system and if a component appears to be producing results that differ from other functionally identical components then it is switched out of the system.

In practice, shared cultural and educational backgrounds as well as problems with clarity of specification means that the practical benefits from this approach are less than theoretically predicted [8]. However, there is no doubt that it does lead to a significant increase in software system reliability.

Fault removal is essentially a development strategy where the goal is to identify faults that have been introduced into the system and then change the system to remove these faults. Different techniques are used to achieve this from very comprehensive system testing through to mathematical proof that a program meets its specification. For most large, complex computer-based systems, fault detection and removal is the most time-consuming and expensive part of the development process.

Fault forecasting does not, in itself, help achieve dependability but helps us make judgements of whether or not the system is sufficiently dependable. Examples of fault forecasting techniques include fault seeding and system reliability modelling[9] [10]. It is essentially impossible to achieve a system that is completely fault-free and pragmatic considerations mean that systems are usually delivered with known (and unknown) faults. Fault forecasting allows the organisations developing and using the system to make judgements about when the risks of failure resulting faults that have not been identified or repaired are acceptable.

2. DEPENDABILITY - A HUMAN PERSPECTIVE

In technical models of dependability, such as the Laprie/Randell model, humans are considered to be system elements that can be treated in the same way as other software or hardware elements. In his paper, Laprie recognises the importance of human operators but discusses them in terms of ‘interaction faults’ resulting from ‘human errors’. Failures on the part of humans in the operational system lead to these interaction faults which result in unexpected computer system state and hence computer system failures. Similarly, as suggested by Figure 3, failures in the development system as a result of human errors lead to the introduction of faults in the operational system.

Human ‘errors’ and the relationships between these errors and system failures have been extensively discussed by authors such as Reason [11]and Rasmussen [12]. Rasmussen discusses different types of human errors such as skill-based, rule-based and knowledge-based errors and Reason, in his ‘Swiss Cheese Model’ relates human error to system failure. He suggests that human errors lead to system failures when they bypass the checks and protection built into a system. Researchers in human psychology argue that so-called ‘human errors’ [13, 14] arise because the systems designers did not consider the practicalities of system operation in their design. Although

we do not discuss dependability from the perspective of human error here, this body of work suggests that failures resulting from human errors have complex causes and should not be considered in the same way as failures deriving from faults in hardware or software components.

If we consider broader socio-technical systems and apply the technical dependability model to the people in these systems, it is our contention that the fault-error-failure model breaks down. Recall that failures are unexpected behaviour, errors are undesirable system states and faults are the causes of an error. The basic problem arises because, for people, the notions of fault, error and failure are inapplicable:

Failure recognition People are not automata and they use their intelligence to discover many different ways of doing the same thing. An action that might be interpreted as a failure for one person (such as an air traffic controller placing aircraft on a collision course) might be part of a dependable operational process for another (the ATC may have a reliable method of ensuring that they will move one of the aircraft before any danger ensues[1]). Clearly the failure is recognised when the near miss occurs but how much earlier could it have been recognised? Was the failure placing the aircraft on a collision course or failing to subsequently separate the aircraft?

Error identification. How can we tell if an unwanted state has resulted in the failure? The notion of explicit state is one that is particular to computer systems and is difficult to apply outside these systems. For example, we cannot monitor our brains to identify the erroneous state that has arisen nor can we keep records of how a set of thought processes led to some action being taken.

Fault recognition. What was the fault that resulted in the human error? Was it a training fault or something more fundamental. People are not deterministic and their emotional and physical state profoundly affects their behaviour. The notion that failures in the development process lead to faults in the 'system' clearly doesn't apply to people. The development process for people from conception (fusing of genetic histories) through nurture to education and training is so extended and complex that identifying the 'fault' that resulted in a consequent failure is impossible.

For some classes of highly automated system, where operational processes and tightly defined and operators are highly trained, then the benefits of adopting a consistent view of dependability that encompasses both people and computers may outweigh the disadvantages of treating the human operators in a simplistic way. However, within organisations, there are many systems that are discretionary whose use is not constrained by

[1] In studies of air traffic controllers, we actually observed this control strategy.

organisational processes and where users do not face sanctions if these are not used. For those systems, the notion of what is mean by a human 'error' or 'failure' is more difficult. If a user does not read a system user guide and hence makes an input error is that a human failure? Or, is this a system failure because the designers have made invalid assumptions about the reality of system use?

Of course, the ultimate discretionary systems are those that we have in our homes. For those systems, there are no organisational constraints – we are free to do what we wish with systems and to discard them if we are unhappy with them. We believe that the simple technical model of dependability as discussed in the previous section does not apply to domestic and discretionary systems. Our work on extending this model domestic systems and the lessons learned for system dependability and trust is the topic of the remainder of this chapter.

3. DOMESTIC SYSTEMS DEPENDABILITY

For domestic systems, the users of the system are central to the design and central to the consideration of dependability. In the home, people do not follow defined operational processes, system users may vary widely and within the same home there may be both techno-phobes and techno-philes. The dependability of home systems is played out daily through the routines and situated actions of the people in the home. Therefore, we contend that the requirements of dependability in the home setting are derived from different roots from traditional dependability models of software design. To achieve dependability, we must take an approach that integrates the user and environment with the technology rather than considering dependability as a property of the technology alone.

CRITERIA	HOME CONTEXT	ORGANISATIONAL CONTEXT
USAGE	Ad Hoc Uncontrolled	Systematically Controlled
STANDARDISATION	Legislative and Product Specific	Standardised with Organisational Environment
PROCESSES	Uncontrolled and Ad Hoc	Controlled and Systematic
OPERATORS	Untrained and Unskilled	Training Available
OPERATIONS	Unrestricted and Ad Hoc	Restricted and Systematised
ACTIONS AND ACTIVITIES	Undefined and Uncontrolled	Predefined and Limited
SAFETY	Suggested but Difficult to Enforce	Controlled through Systems

Table 1: Home and Organisational Differences

In contrast to organisations where technologies and processes are limited, within the home people can choose whether or not to use technology, how to use it and where they wish to use it. People do not read instruction manuals, are not trained in the use of domestic technologies and the use of these technologies often depends on their previous technology experience. For example, on early video recorders the process of setting up a timed recording was difficult and error-prone. Although this has been much improved on modern machines, a large number of people simply do not use pre-recording because their previous experience was that it was beyond their capabilities.

In organisations, activities tend to be set in regular procedures, such that work begins at prescribed times. The organisational system has regular processes through which activities must follow. Dependable operation may rely on this timing. For example, in a hospital, a surgeon in a hospital can usually assume that appropriate pre-operative procedures have been carried out. A significant difference between the organisational system and the home system is that processes and timing are far more flexible and adaptive in the home. Home routines are often unplanned and lacking rigid structure, although foreseen events, such as children's music lessons, may be planned and approximately situated into a daily/weekly/monthly schedule.

Table 1 outlines some of the differences between technology use in organisations and the home environments. Clearly, this is a generalisation and the criteria are not applicable to *all* organisations or *all* homes. However, it essentially summarises what we see as the key differences between these settings, namely the uncontrolled nature of the home.

The overall dependability of an organisational socio-technical system that includes a computer-based system is derived from the dependability of the computer system and how it is used. The controlled nature of the organisational environment means that usage of a computer-based system can be controlled and mandated.

In the home, however, the dependability of the socio-technical system, that is the user plus the technology, depends primarily on how (if at all) the user *chooses* to use that technology. For example, if an elderly person is offered a communication aid that they cannot fit into a pocket of their normal clothing, they may choose not to carry that aid. Therefore, the availability of the communication aid system is limited because the user can't always carry it around. The communication aid itself may be dependable, but the overall *system* of helping with communication is not.

The dependability of systems extends beyond the hardware and software into the social and lived experience of the home dweller. As Lupton and Seymour [15] suggest, technology becomes part of the self-concept for the user and therefore it is essential that dependability does not just mean that a system behaves according to the expectations of its designers. Systems therefore have to be designed so that they are *acceptable* to users and so that they can use them for their intended purpose. We should not underestimate the difficulty of this design problem in domestic settings.

3.1 A dependability model for domestic systems

It is our contention that the techno-centric model of dependability that is exemplified by Laprie's dependability tree needs to be developed and extended for it to be applicable to domestic computer-based systems. We have proposed an augmented model that is based on a number of field studies of people in their homes [16]. Our work, in fact, has focused on a specific type of domestic system namely assistive technology systems for the elderly. However, we believe that it has more general applicability to any type of computer-based system that is used in the home to deliver what the people in that home consider to be important services.

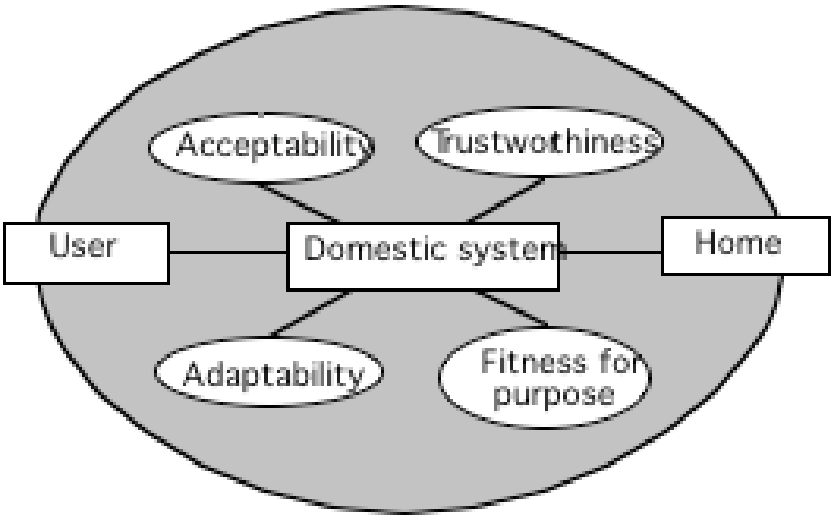


Figure 4: Dependability attributes of a situated AT system

Fundamentally, techno-centric dependability models exclude the user and the user's environment from considerations of what dependability means. These models assume that the system will actually be used as intended by its designers. The technical model of dependability can consider a system that meets its specification to be dependable, even if it is practically useless and never used. We generally reject this view (not just for domestic but for all systems) and believe that we should not just be concerned with dependability *in* use but also dependability *of* use. By this, we mean that it is not enough for a system to be dependable in that it meets its specification and operates according to that specification. The system must also be accepted by its users and used for its designed purpose. Dependability, therefore, is not just a technological consideration but a holistic notion that applies to the technology and its practical use.

For domestic systems, we need to consider the dependability of the socio-technical system as a whole including the user, the home environment and the installed technology. We propose that the dependability characteristics of domestic systems should be considered under 4 headings as shown in Figure 4:

1. *Trustworthiness* The trustworthiness of a system reflects whether or not the system will behave as intended by its designers and as expected by its users. We consider this attribute to be the equivalent of 'dependability' in Laprie's model. That is, it includes the traditional dependability attributes of availability, reliability, etc. However, we suggest below that these may need to be re-interpreted to take into account the specific characteristics of domestic systems.

2. *Acceptability* The acceptability of a system reflects whether or not that system fits in with the user's everyday life and environment. We argue that a system that cannot be integrated with normal activities will not be accepted and so will not be used. Therefore, it is essential that system characteristics that affect its acceptability such as the system learnability and aesthetics are considered in the design process.
3. *Fitness for purpose* Fitness for purpose is taken for granted in most of the dependability literature but, socio-technical system failures regularly arise because a computer-based system does not meet user requirements so that users have had adapt their operational processes to accommodate the system's inadequacies [17, 18]. When the use of a system is discretionary, then it must be fit for the purpose intended by its users; otherwise they simply will not use it.
4. *Adaptability* Within the home both the environment and the user's of the systems change. People's knowledge and capabilities change over time. This is particularly true for elderly people whose vision, hearing and memory tend to decline as they age. Therefore, if system dependability of not to degrade, then it must be able to evolve over time, generally without interventions from the system's designers.

Now let us examine each of these characteristics in more detail to assess what they might mean for domestic, computer-based systems.

3.1.1 Trustworthiness

In the context of domestic systems, we consider the trustworthiness of a system to correspond to the technical notion of dependability as defined by Laprie. That is, the trustworthiness reflects the systems availability, reliability, safety, confidentiality, integrity and maintainability. However, the nature of home systems as assemblies of relatively cheap, off-the-shelf components, the fact that home users are not systematically trained in the use of these systems and the nature of the home itself means that these dependability characteristics have to be re-interpreted for domestic systems:

Availability and reliability

As far as availability and reliability are concerned, we need to consider two classes of domestic system namely critical and non-critical systems. Critical systems are those that supply a critical services such as some assistive technology systems that help elderly or disabled people or control systems for power, external security, etc. Non-critical systems are systems such as entertainment systems where failure is inconvenient but does not pose any real threat to people in the home.

For critical systems, availability and reliability are critical attributes. An elderly or disabled person's quality of life may be dependent on their assistive technologies and failure of these systems has severe implications

for them. For non-critical systems, availability and reliability are perhaps more critical for the system vendor rather than the system user. Failures of these systems can mean that buyers will reject that company's products in future.

However, domestic technology system designers are faced with a challenging problem when trying to build systems by with high-levels of availability and reliability. Systems are mostly composed of off-the-shelf devices where the system designer has no control over the engineering of these devices. For example, consider a situation where a system is to be installed to allow a disabled person to see visitors, communicate with them by voice and to automatically unlock the door if they are to be allowed in. A domestic television is to be used as the display device. This system may involve integrating a set-top box on the television with an external video camera, a voice system and an electronically controlled door lock. These are provided by different vendors and the failure of any one of these components can result in overall system failure.

Cost is often the dominant factor in manufacturing domestic systems so lower quality standards may be applied to systems components and external interfaces may not be provided. Typically, hardly any information may be available about device reliability so designers must trust manufacturer specifications, which, in our experience, are often optimistic.

Safety

Clearly safety is a very important factor in domestic systems and home technology must pass rigorous standards for electrical safety. However, few products can dictate how they should or should not be used in a domestic setting. A large number of domestic accidents result from inappropriate use of equipment. For example, accidents have occurred because people try to use a hairdryer while they are taking a bath, because they try to clean equipment while it is switched on, etc. The critical factor in home safety is rarely the equipment itself but how it is actually used.

Given that most domestic systems are low power systems that conform to electrical safety standards, we consider that the risks of injury associated with failures in computer-based home systems are relatively low. This does not mean, of course, that we should install unsafe systems – however, it does suggest that it is not worth incurring very high costs in activities such as detailed product safety analysis. Rather, it may be more productive to think about 'design for misuse' and try to design these systems so that potentially unsafe ways of using them are made as difficult as possible.

Confidentiality and integrity

If a system is to be dependable, a user must be able to trust that system to keep personal information confidential and to ensure that the information is

not lost or corrupted. This is equally true for organisational or domestic systems. While the need for integrity goes without saying, the issue of confidentiality is much more difficult in situations where elderly or disabled people depend on monitoring technology that alerts relatives and carers when a problem arises. These users often value their privacy and wish to maintain the confidentiality of their personal information. On the other hand, this may compromise the safety of the overall system as it may limit the speed and type of response in the event of a problem. The level of confidentiality in a system therefore cannot be fixed but has to be programmable and responsive to an analysis of the events being processed by the system.

Maintainability

Maintainability is the ability of a system to undergo evolution with the corollary that the system should be designed so that evolution is not likely to introduce new faults into the system. We distinguish here between maintainability as the process of making unanticipated engineering changes to the system and adaptability, which is the process of changing a system to configure it for its environment of use. It is now the case that the low-cost of much domestic equipment means that that replacement rather than maintenance is the norm so software and hardware changes and upgrades are unlikely. Therefore, we consider maintainability under the adaptability attributes that we discuss later.

3.1.2 Acceptability

Acceptability reflects whether or not a domestic system fits in with the user's abilities, personal preferencies, environment and routines of everyday life. The notion of acceptability was initially conveyed through an advocate of Universal Design (UD), an approach to design that advocates that designers should design for all ages and skills. Sandhu [19] presents a diagrammatic representation of system acceptability within a Universal Design context (Figure 5). Systems that are not acceptable to users will simply be discarded even in situations where their functionality is clearly of some value.

Sandhu's diagram illustrates that for systems to meet his Universal Design criteria there are a considerable number of attributes and properties that the system and designer must address that are comparable to those derived by software engineers considering dependability. The model that Sandhu proposes situates the user and the product within the same contextual model so reflects our views on the central significance of the user when considering system dependability.

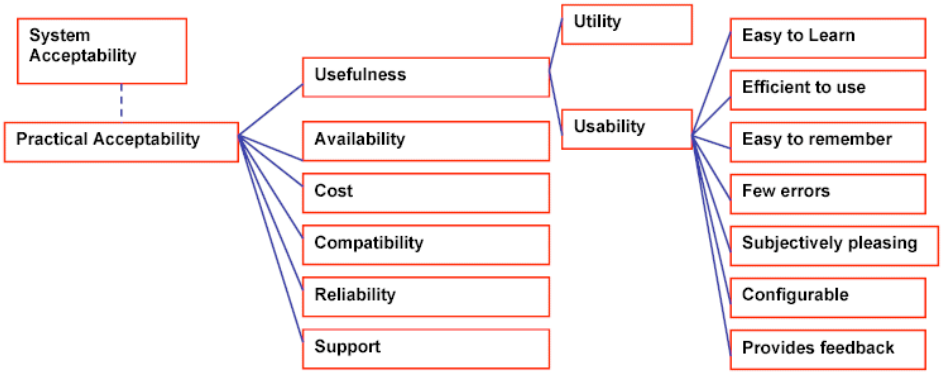


Figure 5. Sandhu's system acceptability model

Our view of acceptability takes a simplified view of Sandhu's model as we consider some of his acceptability characteristics such as reliability, availability and configurability under other headings such as trustworthiness. Essentially, we consider that a system will only be acceptable if the user feels that the benefits that accrue from the system justify the costs and effort of buying, installing, learning to use and using the system. We therefore consider the principal acceptability characteristics to be:

- *Usability* It must be possible to use the system on a regular basis without error and without having to re-learn how to benefit from the system. This suggests that user interfaces should be intuitive and should not be based on modes or complex sequences of actions.
- *Learnability* It should be possible to learn to use the system relatively easily with no steep learning curve before any benefits can be gained from it. Again, this highlights the needs for intuitive interfaces that reflect the most common ways in which the system might be used.
- *Cost* The system should also be within the budget of the person allowing for maintenance and repair costs in the future.
- *Compatibility* The system must be compatible both physically and electronically with other systems that are installed in the home. Systems should, essentially, be 'plug and play' and users should not have to understand the details of interfaces to make different products work together.
- *Efficiency* The effort and time saved by using the system must significantly exceed the effort involved in making use of it.
- *Responsiveness* The system must respond in a timely fashion to user requests and provide feedback on its operation to the user.

- *Aesthetics* If a system is to be actively used in the home, it should be aesthetically pleasing, blending in with the décor of the existing home and the user's taste.

Of course, these factors are not just relevant to domestic systems but apply in many cases, to organisational settings. The difference, however, is that in organisational settings resources may be available to pay for ways to cope with the deficiencies in the technology. Problems of acceptability may be addressed through training and the adaptation of operational processes. However, we strongly believe that, in this area, the design of organisational as well as domestic systems would benefit if system designers paid more attention to the acceptability of these systems in their intended environment.

3.1.3 Fitness for purpose

The fitness for purpose of a domestic system reflects the extent to which that system meets the real needs of its users. This is particularly important for systems, such as assistive technologies for the elderly or disabled. These are not mass-produced consumer commodity systems but are systems that are designed and tailored specifically for an individual set of disabilities. If the systems do not address the specific problems faced by the user, they are essentially useless.

Fitness for purpose is related to but distinct from acceptability. A domestic technology system may be acceptable to a user but if it is not carefully tailored to their specific needs then the compromises that have to be made in using the system may lead to system failures. For example, a voice-activated system may be installed to help elderly users set off an alarm in the event of accident or illness. This system may work reliably so long as the user's voice is strong enough but if it does not take into account the fact that the elderly person's voice may be weakened in the event of an accident then it is not fit for its intended purpose.

Of course, this is not just an issue for domestic system but a more general dependability concern. For organisational systems, dealing with this concern is seen as a specification issue i.e. failure to meet real needs is equated to a specification failure. Given that the level of specification that is used for organisational systems is totally impractical for domestic systems for the elderly or disabled, the issue of fitness for purpose cannot be addressed in this way. Rather, the system has to be designed to evolve during installation and use to take into account the routines of the user's life and the particular characteristics of that user and their home.

3.1.4 Adaptability

Homes and the people living in these homes change with time [20]. Spaces are reconfigured to cope with changing demands and tastes, new people come to live in the home, children grow up and the capabilities of elderly

adults may decline as they grow older. Consequently, the requirements of users in the home for domestic systems are constantly changing. If systems cannot be adapted *in situ* to meet new requirements they will become less and less used and, hence, less dependable.

We can identify three types of modification that may be made to domestic systems:

- Addition of new equipment. This can be in addition to existing equipment or can replace obsolete devices. Given the relatively low costs of domestic equipment, this will often be the most cost-effective way to modify a system.
- System configuration or re-configuration by its users. In this case, the user (or someone with technical knowledge) adapts the system using built-in capabilities for adaptation. For example, if a person's eyesight degenerates, then the default font size on a screen that they regularly read may be increased.
- Configuration or re-configuration of a system by its supplier. In this case, the supplier or installer of the system may visit the home to make the system modifications. Alternatively, if the system can be connected to a network, then remote upgrades of the software may be possible. This is already commonplace for mobile phones and digital TV set-top boxes.

Of course, it is well known that dependability problems in computer systems regularly arise because of errors made during system maintenance. These occur in spite of extensive quality control and testing mechanisms that are in place. There are no such mechanisms in the home so clearly the potential for undependability after modification is significant. This fact, along with the need to support system change leads to the following adaptability attributes:

1. *Configurability* This attribute reflects the ability of users or equipment installers to adapt the system to cope with a range of human capabilities such as variable hearing, eyesight, balance, etc.
3. *Openness* This attribute is concerned with the system's ability to be extended with new equipment, perhaps from different manufacturers.
4. *Visibility* This attribute reflects the extent to which the operation of the system can be made visible to users and installers of that system. This is particularly important when problems arise as it increases the chances that these problems can be diagnosed without expert assistance.
5. *User repairability* This attribute reflects the extent to which system users can repair faults in the system without specialist tools or knowledge. This is important for domestic systems as it means that users

or helpers can fix problems without the need for an external service call. Thus the system can be brought back into operation quickly and the overall availability of the system is increased.

4. DEPENDABILITY, TRUST AND DISCRETIONARY SYSTEMS DESIGN

We have argued that, for domestic systems, we need to extend the notion of technical dependability as developed for organisational systems to embrace broader notions of acceptability, fitness for purpose and adaptability. The question now is: how can this broader dependability model be used to help system designers create better systems? That is, how do we design systems that, within a socio-technical context, are more likely to be trusted by their users?

Although the focus of our work has been domestic systems, we are convinced that the domestic dependability model is equally applicable to 'discretionary systems' in organisations. Professional users in organisations, such as doctors or engineers, who choose to use systems to support their work are often unwilling to change their ways of working to accommodate these systems. As in the home, they have rhythms and routines of daily work and they expect their computer systems to fit in with these. They become extremely frustrated if they have to change how they work because of the computer system and, in such circumstances, will simply discard the system. Therefore, we argue that the dependability model for domestic systems may also be applied to discretionary workplace systems.

We believe, that for discretionary systems design, there are a number of ways in which the dependability model may be used:

1. As a way of focusing communications with potential system users.
2. As a way of organising and presenting observational studies.
3. As a checklist for designers of discretionary systems.
4. As a means of assessing existing technology and classifying problems and deficiencies in that technology.

The problem of discovering user requirements for a system is recognised as the most difficult issue in computer systems engineering [21]. The essential difficulty is that system users really don't really know what they want from a system. This problem is particularly acute for discretionary systems where there is no strictly defined process that users must follow to do their work. Even in situations where the users have a fairly clear idea of what they would like, they are poor at articulating the practical constraints on the operation of the system. The advantage of using the model that we propose here to structure communications with potential users is that it integrates

functional characteristics (fitness for purpose) with non-functional characteristics (trustworthiness and acceptability). Furthermore, it highlights the importance of evolution and change (adaptability) so allowing the discussion to consider not just the immediate user requirements but how these requirements might change.

A related use of the model is to help organise and present field studies in the home or workplace. Field studies (ethnographies) collect a large volume of data about the rhythms and routines of everyday life and work including data on the use (or the lack of use) of technology. We are not suggesting that the model itself drives the ethnography. Rather, it becomes useful once studies have been completed as it allows the ethnographer to organise his or her data in such a way that it can be communicated to the potential system users or to system designers.

Both of these uses of the model are appropriate in situations where a system is being developed for use in a specific setting with a clearly identified set of users. This may be a discretionary system for professionals (e.g. problem reporting system for anaesthetists) or a specially constructed system to support a disabled person in their home. However, many domestic systems, in particular, are developed and marketed as generic products that are intended for use in a wide variety of different situations.

The danger here is that designers of these generic products focus on the product technology and the functionality that it delivers without paying sufficient attention to how it will actually be used. We see this in all sorts from products from mobile phones to video recorders and in the invention of a range of devices for 'the home of the future' such as smart fridges and heating systems. Such systems often include unnecessary and unwanted functionality that serves to confuse normal operation of the device. The dependability model that we propose, with its focus on the user and use of the technology, provides a checklist to designers that helps them consider how the technology will be used. From the model, we can derive questions such as:

- How will the user learn to use the system?
- Can they get some benefit from the system without reading an instruction manual?
- Is there a need to interface this system with other systems in the home or workplace?
- How will the system provide feedback on its operation to users?
- What user-level configurability will be supported in the system?
- How will users (with different ability and experience) access this configurability?

- How can users find out what the system is doing?

Finally, an immediate use of the model is as a way of assessing existing systems and classifying the problems that arise with these systems. It can be used in this way with one-off systems such as systems intended to help a disabled person, with workplace systems used by a group of professionals or with generic products. In the latter case, the model can be the basis of a user survey to elicit information about what users like and don't like about a system.

To illustrate how the model might be used, consider a situation where a system is to be installed in a housing complex for elderly people that is intended to help them communicate informally and share information. It makes them aware of who is available and interested in talking, provides a messaging facility and access to an electronic noticeboard that maintains information that is potentially of interest to all residents.

It is not possible to provide a complete analysis of this system here but the snapshot below shows how the classification in the model can highlight issues that have to be considered by system designers.

Attribute	**Issue**	**Proposal**
Confidentiality and integrity	Some users are concerned that it will be possible to 'eavesdrop' on private communications.	This is true but addressing the problem adds to the complexity of the system. Inform users that the system is not intended for private conversations.
Maintainability	Inevitably, there will be system software failures and the software will have to be restarted. Repairs and updates will be required. However, at least some of the users will not be able to do any installations themselves.	Provide a remote diagnostic and maintenance facility so that updates are possible without user intervention. Provide a (large) restart button on the device.
Learnability	Many users have minor problems with short-term memory. Hence, learning how to use the system can take some time.	Provide all users with a quick reference card. Arrange 'buddies' so that people help each other to learn to use the system.
Compatibility	Each flat in the complex has an alarm system that can be used to call for help if an emergency arises. Ideally, it should be possible to activate this from this system.	Requires further analysis to see if system protocols are compatible.
Aesthetics	Users are short of space and mostly have traditional decoration in their homes.	Use a tablet-PC and thus avoid the need for electronic boxes. All communications should be wireless.
Configurability	Some users suffer from arthritic fingers and have difficulty pointing at small targets.	The user should be able to increase or decrease the size of all buttons and menus in the system.

5. CONCLUSIONS

This paper has discussed traditional notions of system dependability that have arisen from research into computer-based control and protection systems. These have highlighted the importance of system characteristics such as reliability, availability and safety. Building on this work, we have proposed broader notion of dependability for systems, such as domestic systems, where users choose whether or not to use these systems. We argue that dependability is not just a technical system attribute but also includes those factors that influence the user's choice of whether or not to use a system. If a system is not used, it is not meeting its designer's intentions and hence, we argue, it is not dependable.

We believe that the domestic dependability model is an important contribution to broadening the notion of system dependability and has real practical value in the analysis and design of domestic and discretionary systems. We are currently gaining experience in the use of the model in the design of a communications system for elderly people and anticipate that this will allow us to extend the approach. Extensions may include a discussion of impairments – what stops a system being used – and design guidelines that provide more detailed advice for system designers.

6. ACKNOWLEDGEMENTS

We would like to thank Age Concern, Barrow; MHA Penrith, Dundee Social Work and Aberdeen Social Work departments as well as a number of elderly system users and potential users who helped us understand their needs for domestic systems. The research was sponsored by the EPSRC under the DIRC Inter-disciplinary Research Collaboration on system dependendability.

7. REFERENCES

1. Laprie, J.-C. *Dependable Computing: Concepts, Limits, Challenges.* in *25th IEEE Symposium on Fault-Tolerant Computing*. 1995. Pasadena: IEEE Press.
2. Sommerville, I., *Software Engineering, 6th edition.* 2001, Harlow, UK: Addison-Wesley.
3. Ellison, R.J., et al., *Survivable Network System Analysis: A Case Study*. IEEE Software, 1999. **16**(4): p. 70-7.
4. Randell, B., *Facing Up To Faults.* Computer J., 2000. **45**(2): p. 95-106.
5. Avizienis, A., *The N-Version Approach to Fault-Tolerant Software*. IEEE Trans. on Software Eng., 1985. **SE-11**(12): p. 1491-501.

6. Avizienis, A.A., *A Methodology of N-Version Programming*, in *Software Fault Tolerance*, M.R. Lyu, Editor. 1995, John Wiley & Sons: Chichester. p. 23-46.
7. Randell, B. and J. Xu, *The Evolution of the Recovery Block Concept*, in *Software Fault Tolerance*, M.R. Lyu, Editor. 1995, John Wiley & Sons: Chichester. p. 1-22.
8. Brilliant, S.S., J.C. Knight, and N.G. Leveson, *Analysis of Faults in an N-Version Software Experiment*. IEEE Trans. On Software Engineering, 1990. **16**(2): p. 238-47.
9. Littlewood, B., *Software Reliability Growth Models*, in *Software Reliability Handbook,*, P. Rook, Editor. 1990, Elsevier: Amsterdam. p. 401—412.
10. Musa, J.D., *Software Reliability Engineering: More Reliable Software, Faster Development and Testing*. 1998, New York: McGraw-Hill.
11. Reason, J., *Human Error*. 1990, Cambridge, UK: Cambridge University Press.
12. Rasmussen, J., *The definition of human error and a taxonomy for technical system design*, in *New Technology and Human Error*. *Chichester: Wiley*. , J. Rasmussen, K. Duncan, and J. Leplat, Editors. 1987, John Wiley and Sons: Chichester.
13. Norman, D.A., *The Psychology of Everyday Things*. 1988, New York: Basic Books.
14. Norman, D.A., *Human Error and the Design of Computer Systems*. Comm. ACM, 1990. **33**(1): p. 4-7.
15. Lupton, D. and W. Seymour, *Technology, selfhood and physical disability*. Social Science & Medicine, 2000. **50**(1851-62).
16. Sommerville, I., et al. *A Dependability Model for Domestic Systems*. in *SAFECOMP 2003*. 2003. Edinburgh, Scotland: Springer.
17. Edwards, K. and R. Grinter. *At Home with Ubiquitous Computing: Seven Challenges*. in *Proc. Ubicomp 2001*. 2001. Atlanta, Georgia.: Springer.
18. Miller, C., K. Haigh, and W. Dewing. *First, Cause No Harm: Issues in Building Safe, Reliable and Trustworthy Elder Care Systems*. in *Proc. AAAI-02 Workshop on Automation as Caregiver*. 2002. Edmonton, Canada.
19. Sandhu, J., *Multi-Dimensional Evaluation as a tool in Teaching Universal Design*, in *Universal Design; 17 Ways of Teaching and Thinking*, J. Christopherson, Editor. 2002: Husbanken, Norway.
20. Dewsbury, G., et al. *Designing Dependable Digital Domestic Environments*. in *Proc. HOIT 2003: The Networked Home and the Home of the Future*. 2003. Irvine, California.

21. Kotonya, G. and I. Sommerville, *Requirements Engineering: Processes and Techniques*. 1998, Chichester, UK: John Wiley and Sons.