Safety Critical Systems: Challenges and Directions

John C. Knight
Department of Computer Science
University of Virginia
151 Engineer's Way
Charlottesville, VA 22904-4740
+1 434 982 2216

knight@cs.virginia.edu

ABSTRACT

Safety-critical systems are those systems whose failure could result in loss of life, significant property damage, or damage to the environment. There are many well known examples in application areas such as medical devices, aircraft flight control, weapons, and nuclear systems. Many modern information systems are becoming safety-critical in a general sense because financial loss and even loss of life can result from their failure. Future safety-critical systems will be more common and more powerful. From a software perspective, developing safety critical systems in the numbers required and with adequate dependability is going to require significant advances in areas such as specification, architecture, verification, and process. The very visible problems that have arisen in the area of information-system security suggests that security is a major challenge also.

1. INTRODUCTION

A modern heart pacemaker is a computer with specialized peripherals, the U.S. Air Force's F22 fighter relies heavily on a computer network as does a modern car, and many defense facilities are actually distributed computer systems. These and many other systems are examples of so-called *safety-critical systems*, a term whose customary meaning is systems whose failure might endanger human life, lead to substantial economic loss, or cause extensive environmental damage.

Many modern systems depend on computers for their correct operation. Of greatest concern, of course, are safety-critical systems because their consequences of failure can be considerable. There are many applications that have traditionally been considered safety-critical but the scope of the definition has to be expanded as computer systems continue to be introduced into many areas that affect our lives.

The future is likely to increase dramatically the number of computer systems that we consider to be safety-critical. The dropping cost of hardware, the improvement in hardware quality, and other technological developments ensure that new applications will be sought in many domains.

In this paper I summarize the challenges that we face in this important area and discuss some of the opportunities for meeting those challenges. Although the word "system" applies to the structure providing service, our concern is with the computers upon which such systems depend. Throughout the rest of this paper, no distinction is made between a safety-critical system and the computer system upon which it depends.

ICSE'02, May 19-25, 2002, Orlando, Florida, USA.

2. WHAT ARE SAFETY-CRITICAL SYSTEMS?

There are plenty of definitions of the term *safety-critical system* but the intuitive notion actually works quite well. The concern both intuitively and formally is with *the consequences of failure*. If the failure of a system could lead to consequences that are determined to be unacceptable, then the system is safety-critical. In essence, a system is safety-critical when we depend on it for our well being. In this section, the implications of this idea are explored in terms of the classes of systems that should be viewed as safety-critical.

2.1 Traditional Systems

Traditional areas that have been considered the home of safety-critical systems include medical care, commercial aircraft, nuclear power, and weapons. Failure in these areas can quickly lead to human life being put in danger, loss of equipment, and so on.

Computers are used in medicine far more widely than most people realize. The idea of using a microprocessor to control an insulin pump is quite well known. The fact that a pacemaker is largely a computer is less well known. The extensive use of computers in surgical procedures is almost unknown except by specialists. Computerized equipment is making inroads in procedures such as hip replacement, spinal surgery, and ophthalmic surgery. In all three of these cases, computer controlled robotic devices are replacing the surgeons traditional tools, and providing substantial benefits to patients.

The Boeing 777 is described by Boeing as "The Most Technologically Advanced Airplane In The World." Many different technologies have contributed to the aircraft including safety-critical computer systems. There are six primary flat-panel displays and several other smaller displays in the cockpit. The aircraft has several major computerized systems to aid the pilot including flight management and enhanced ground proximity warning. Much of the traditional mechanical and hydraulic equipment is obviated by the use of a fly-by-wire control system. The Boeing 777 primary flight control system uses three separate channels for redundancy. Each channel is implemented with three separate lanes, each of which uses different processors and different compilers. Extensive networking provides the necessary communication between the different subsystems.

2.2 Non-traditional Systems

The scope of the safety-critical system concept is broad, and that breadth has to be taken into account when practitioners and researchers deal with specific systems. A closer examination of the topic reveals that many new types of system have the potential for very high consequences of failure, and these systems should probably be considered safety-critical also. It is obvious that the loss of a commercial aircraft will probably kill people. It is not obvious that loss of a telephone system could kill people. But a protracted loss of 911 service will certainly result in serious injury or death.

Emergency 911 service is an example of a critical infrastructure application. Other examples are transportation control, banking and financial systems, electricity generation and distribution, telecommunications, and the management of water systems. All of these applications are extensively computerized, and computer failure can and does lead to extensive loss of service with consequent disruption of normal activities. In some cases, the disruption can be very serious. Widespread loss of water or electricity supply has obvious implications for health and safety. Similarly, widespread loss of transportation services, such as rail and trucking, would affect food and energy distribution. It is prudent to put the computer systems upon which critical infrastructures depend into the safety-critical category.

2.3 System Design and Manufacturing

It is usually the case that operational systems, i.e., systems such as those discussed in the previous sections, are the only ones thought of as safety-critical. Obviously, failure of such systems can immediately present significant danger.

There are, however, plenty of software systems that are used in the design and manufacture of other systems where the consequences of failure could be considerable. Software that support the development of other software (such as a compiler) is itself safety-critical if the product that it supports is safety-critical.

Equally important are computer systems that support the development of non-computer artifacts. MSC Corporation's NASTRAN system for structural analysis is used heavily in many different industries [5]. It is relied upon to assist in structural design automation, and its analysis is assumed by structural engineers to be correct. Were it in error, the result could be a defective structure. Thus although the end product might be a building or a bridge, the dependence of that end product on a computer system during design make the design computer system safety-critical.

2.4 Information System Security

It has become clear that security attacks against information systems are a large and growing problem. Attacks against both public and private networks can have devastating effects. The Internet is being used increasingly to provide communication service to business, and security attacks against the Internet are a troubling problem for network users.

Although Internet attacks are important, private networks are a bigger concern. Money is moved locally and around the World on private networks owned by financial institutions. Transportation systems are monitored and controlled using mostly private networks. A successful attack against certain private networks could permit funds or valuable information such as credit card numbers to be stolen, transportation to be disrupted, and so on.

The potential for loss is considerable, and, although no physical damage would be involved in security failures, the consequences of failure are such that many systems that only carry information should be regarded as safety-critical.

3. WHAT GOES WRONG WITH SAFETY-CRITICAL SYSTEMS?

There are several well-known examples of safety-critical system failures that have occurred including the Space Shuttle countdown failure on the first launch, the Ariane V launch failure [2], and the losses of the Mars Polar Lander [3] and the Mars Climate Orbiter [6]. Many examples are documented in the text by Neumann [7]. Rather than repeat the details of such failures, several less-well-known system failures are described to illustrate the breadth of the problem.

An example of what can go wrong with systems supporting design occurred with two programs that perform finite-element analysis. These programs are used extensively in engineering design, particularly structural design. In May 1996, the Nuclear Regulatory Commission published NRC Information Notice 96-29 addressed to all holders of operating licenses or construction permits for nuclear power reactors [8]. The Commission had determined that 150 errors had been reported for these programs at that time but the Commission did not know how the programs had been used in safety analysis at nuclear power plants. The recipients of the notice were asked to review the information for applicability. The implications are obvious.

A different type of problem arose with the primary protection system for the Sizewell B nuclear power reactor in the United Kingdom. This system was implemented in software and was required to achieve a reliability of no more than 10^{-4} failures per demand. Once the system design was complete, it used more than 650 microprocessors and 1,200 circuit boards, and the software was over 70,000 lines long [1]. When system tests were carried out, the system only passed 48% of the test cases [4]. The system was reported to have failed the other 52% but, in fact, the real problem was that, for many of the tests, it was not possible to determine whether the test had been passed. This system was designed to shut the reactor down when some sort of problem arose, surely a safety-critical application. Meeting the reliability goal seems unlikely for a system with this many computers, this number of lines of code, and this testing history.

On October 26, 1992, the ambulance service for the city of London, England, switched from a manual dispatch system to a computer aided dispatch system [9]. The changeover was made all at once so that the computerized system was expected to operate for the entire coverage area. The system worked initially but a complex sequence of events led to the system being essentially non-operational as the demand increased during the day. Since ambulance dispatch was severely delayed in many cases, there is good reason to think that deaths or injury resulted from the failure.

There are many lessons that can be learned from incidents such as these, but, unfortunately, the lessons are sometimes missed. The Mars Climate Orbiter crash, for example, occurred because the wrong system of units was used in part of the ground-based software [6]. The first of the recommendations in the report of the mishap investigation board was "that the MPL project verify the consistent use of units throughout the MPL spacecraft design and operation". The MPL is the Mars Polar Lander, a spacecraft that arrived at Mars (and also crashed) after the board issued its report—the goal was to avoid similar problems with the MPL.

Although this is sound advice, it is not the right lesson. The cause of the problem was an incorrect assumption about the use of units that was never checked. Checking units carefully will eliminate repetition of this particular problem, but it will not deal with the general problem of incorrect assumptions. That is a fundamental problem in the way that engineers communicate, and it should be recognized and dealt with as such. It is impossible to eliminate

such problems with any form of simple checklist.

4. FUTURE SYSTEMS

Future developments are likely to extend considerably the number and type of safety-critical systems with which we have to deal. In this section, the impact of technology and the planned areas of enhancement in some example application areas are examined.

4.1 Technology

Hardware development continues at a breathtaking pace with an apparently endless series of improvements in processor speed, memory size, disk capacity and communication bandwidth, and the introduction of relatively new capabilities such as wireless. Despite these advances, several areas of technology have not developed as quickly.

Energy storage is still problematic and limits the use of computers in safety-critical applications. Consider what would happen if the capacity of a AA battery grew at the same rate as the capacity of disk drives. Although unlikely, this is not inconceivable—think of a AA-sized chunk of pure plutonium. We have the technology to remove at least a little of the energy in the plutonium by well-understood fission processes. Having abundant and compact power sources would allow many valuable safety-critical applications to be developed that are presently impractical.

A second area in which technology has not advanced as far as many other areas is communication between high-speed networks and local devices. Imagine the effect of solving the "last mile" problem. We speak of great communications capacity but in reality we do not have it. There is no cost-effective way to get a gigabit/sec. to your home, your car, or your PDA. Ubiquitous, high-bandwidth communication would facilitate a number of new safety-critical applications such as remote medical monitoring and intervention, remote vehicle control, and information-intensive actions by police and military personnel.

As technology advances, scale is going to be a real challenge. More safety-critical applications will be feasible and in greater numbers. Design, development, and deployment of such systems will require significant breakthroughs in both software and systems engineering.

4.2 Applications

Future transportation systems will be far more automated and far more dependent on safety-critical computers than today's systems. The air-traffic-control system is transitioning to use of the Global Positioning System for navigation and precision approaches. Free flight, in which aircraft outside terminal areas are not controlled from the ground is being brought into service, and highly automated control mechanisms are being developed for aircraft control in terminal areas.

The design of commercial aircraft will change radically in the future also. Aircraft that change their shape in flight to optimize their aerodynamic performance are being considered. At the other end of the scale, what amount to personal aircraft that will operate much like a car and require about as much training to fly as one needs to drive a car are being discussed. All of these aircraft will require constant and extensive computer control if they are to operate successfully.

For large commercial aircraft, "synthetic vision" systems are being developed for pilot support in low visibility. The goal is to provide displays that give the pilot all the necessary visual information for all types of flying in weather conditions that would normally limit flight operations.

Automobiles have already been invaded by microprocessors in many safety-critical applications. "Drive-by-wire" systems are not far off, and such systems will remove all mechanical linkages between the driver and the car's various systems. Traffic management systems are gradually being invaded by microprocessors also. Several large cities, Washington DC and surrounding areas for example, have many of their traffic signals controlled by a central computer in order to optimize traffic flow. Traffic sensors and traffic-signal control commands flow around the region via the telephone network. Integration of vehicle computer systems with traffic flow computers is clearly an important approach.

Once the last-mile problem is solved, our overall dependence on information systems will increase dramatically as we continue the transition from moving people to moving information. Telecommuting, for example, would be far more effective if high quality video conferencing were routinely available for private use. As larger and larger fractions of the workforce work in non-traditional environments, the dependability of the underlying information system will become more and more critical.

The same information systems that facilitate telecommuting will provide enhanced remote services such as sophisticated telemedicine in real time. Rather than requiring hospitalization, in many cases it will be possible to permit people to stay at home but be monitored remotely and treated by automated in-home equipment. An entire living environment then becomes a safety-critical computer system.

5. CHALLENGES

We are rapidly moving to a situation in which computers are "embedded" in society as well as in traditional control systems thereby blurring somewhat what we mean by an embedded system. The result is that serious consequences of failure arise for all the traditional safety-critical application but also for entirely new application domains. In addition, entirely new failure modes are evolving such as denial-of-service attacks against networked information systems. Damage occurs not just through physical effects but also through removal of service or damage to information.

In one way or another, many people in the software business are working on safety-critical systems technology. Many more systems than one might expect have to be viewed as safety-critical, and the number is increasing all the time. So what are the major challenges that we face?

In some cases, what amount to completely new technologies are required. The number of interacting safety-critical systems present in a single application will force the sharing of resources between systems. This will eliminate a major architectural element that gives confidence in correct operation—physical separation. Knowing that the failure of one system cannot affect another greatly facilitates current analysis techniques. This will be lost as multiple functions are hosted on a single platform to simplify construction and to reduce power and weight requirements. Techniques that provide high levels of assurance of non-interference will be required.

Breakdowns in the interplay between software engineering and systems engineering remains a significant cause of failures. It is essential that comprehensive approaches to total system modeling be developed so that properties of entire systems can be analyzed. Such approaches must accommodate software properly and provide high fidelity models of critical software characteristics. They must also deal with the issue of assured non-interference.

Defective software specifications are implicated in many serious failures, and it is clear that we have difficulty stating exactly what software is required to do. There are many aspects of specification that are not supported by any current technique, and, even where specification techniques do exist, there remains a lack of integration to permit whole specification analysis.

Verification by testing is impossible for systems that have to operate in what has been called the ultra-dependable range. Yet, in practice, there are few choices. Formal verification and model checking are desirable technologies but are limited in their applicability. High performance, rapid, comprehensive approaches to verification will be essential if we are to have confidence in the wide variety of safety-critical systems that we expect.

Development time and effort for safety-critical systems are so extreme with present technology that building the systems that will be demanded in the future will not be possible in many cases. Any new software technology in this field must address both the cost and time issues. The challenge here is daunting because a reduction of a few percent is not going to make much of an impact. Something like an order of magnitude is required.

Security is becoming an increasingly important topic in the field of safety-critical systems, and it must be addressed comprehensively if safety-critical systems are to be operated successfully. The challenge here lies very much in the field of software engineering rather than security technology. The vast majority of security problems that arise in networked information systems arise because software defects make the systems vulnerable to attack. The common problem of buffer-overrun attacks is well understood but such attacks continue because systems continue to be deployed with vulnerabilities.

Finally, it is important to raise awareness of the current limitations of software engineering. Engineering of safety-critical systems is a complex task involving many technical fields. Software is a key component of any safety-critical system yet far too few engineers in other disciplines understand what software can and cannot do. We can begin the process in the universities by introducing more material on safety-critical systems into our courses.

6. ACKNOWLEDGEMENT

This work was funded in part by NASA under contract NAG-1-2290.

7. REFERENCES

[1] Day, J.W., The Reliability of the Sizewell 'B' Primary Protection System, Reactor protection Group, Nuclear Electric (January, 1990).

[2] European Space Agency, Ariane 501 Inquiry Board Report (July 1996) http://ravel.esrin.esa.it/docs/esa-x-1819eng.pdf

[3] Jet Propulsion Laboratory, Report on the Loss of the Mars Polar Lander and Deep Space 2 Missions, JPL D-18709 (March 2000).

[4] Marshall, P., NII Found Problems in 50%-Plus of Sizewell-B PPS Computer Tests, Nucleonics Week, 34, 43 (October 28, 1993).

[5] MC Software, Inc. http://www.mscsoftware.com/products/index.cfm?S=85

[6] National Aeronautics and Space Administration, Mars Climate Orbiter Mishap Investigation Report, Washington, DC (November 1999) ftp://ftp.hq.nasa.gov/pub/pao/reports/2000/MCO_MIB_Report.pdf

[7] Neumann, P. *Computer Related Risks*, Addison Wesley (1995)

[8] Nuclear Regulatory Commission, Information Notice 96-29, Washington, DC (May 1996).

[9] The Communications Directorate, South West Thames Regional Health Authority, Report of the Inquiry into the