

Project SL. No. (3): Computerized Information Systems (*CIS*) constitute the technology for organization of information decision for complex systems. Discuss errors in *CIS*.

(Note: This project can be further developed with particular reference to small and large engineering systems, though the investigation will also be applicable to soft systems.)

Framework:

In today's world, with digital data flow across the enterprise becoming a technological reality, *IS* is a computerized information system, which plays an important part in monitoring, controlling and managing complex systems.

Thus, the computerized (main-frames, minis or micros included) information systems today are the nerve centers of almost all small scale and large scale systems. For example, they constitute a flight control system in an advanced aircraft, a safety control system of a nuclear reactor or a life support system in an intensive care unit, as also controls for large engineering and social systems such as power networks, air traffic control systems, military command and control systems, industrial process complexes, banks, etc.

And it is these computerized information systems that contain errors that are made but not corrected. These errors could be due to various reasons/faults and result in failures as follows:

- (a) Malfunctions and defects in instrumentation or computer hardware/software leading to significant loss of revenue, injury or loss of human life, destruction of expensive equipment, loss of valuable management information, pollution of environment with toxic gases, etc.,
- (b) An erroneous signal from a computer to a pressure valve causing a vessel to burst,
- (c) A malfunction of flight control computer in an aircraft damaging the engine,
- (d) A software error in a rail network computer derailing a train,
- (e) Failure of a safety control system causing unnecessary shutdown,
- (f) Failures of sensors and actuators causing instabilities in the feedback system leading to plant shutdown or equipment breakdown,
- (g) Computer providing wrong information to operator which in turn leads to hazardous situations,
- (h) Computerized information systems are composed of three major systems: The dynamic process, the computer and the instrumentation system which interfaces the plant (physical system) with the computer. The operator (i.e. indirect or direct user of the *CIS*) also plays a major role in performance of the *CIS*. Each of these three sub-systems as well as the *CIS* user (direct as well as indirect) are complex dynamic systems and comprise several components. The computer has various hardware units such as I/O units, CPU, memory, etc. and complex applications as well as systems software. The instrumentation system has sensors, signal conditioners, ADCs, DACs and actuators such as valves, motors, and switches.

Failures in any one of these hardware/software components (and also of the direct as well as indirect user) can introduce error(s) in *CIS*.

- (i) Specifically, in the dynamic plant, several failures resulting in structural changes can occur: failure of interconnections resulting in isolation or disconnection of one or more subsystems, power supply failures, etc.

- (j) In the instrumentation subsystems, failures can occur in sensors, instruments, ADC's, DAC's, motors, valves, etc. impairing its normal functioning and causing errors in *CIS*.
- (k) Failures in digital computers can cause serious information processing errors. One example of such failure is that of a single failure lasting a billionth of a second in one of the hundreds and thousands of digital components that switch billions of times a day. Experience has shown that almost all malfunctions are distributed among three types: illegal operations, hardware failures and software failures.

There are fundamental differences between hardware failure and software failure; namely, (i) Software failures are generally design faults, whereas hardware faults are predominantly component failures, (ii) Checking and correction on software can be done independently on a copy of the software module resident in the computer system whereas in case of hardware faults the failed unit is necessary for conducting diagnostic tests, (iii) Software faults cause wrong output signals of the *CIS*, i. e. the control computer, which may in turn cause damage to the plant and more appropriately to the entire supply chain delivering incorrect product but the software itself is not affected. This is in contrast with hardware failures.

- (l) Examples of illegal (incorrect) operations are: (i) the overflow of arithmetic registers, (ii) an attempt to divide a number by zero, (iii) execution of an instruction with an incorrect format or undefined operation code, etc. These errors are generally detected by hardware or programs that monitor user programs.
- (m) Hardware faults include: (i) unstable power supplies, (ii) failure of cooling system, (iii) faults in disk memory, I/O channels, CPU registers, (iv) failure of any peripherals, etc.
- (n) Coming to software failures, they could be due to faults such as: (i) design errors in the original programs, (ii) errors introduced while modifying software, (iii) errors introduced due to transient hardware errors which corrupt the stored program, etc.
- (o) The question of failures in construction and in use of computers has been of importance right since the early days of computer. The first generation of computers was characterized by relatively low architectural complexity (as measured in terms of the number of gates and gate connectivity), small program size, high hardware cost and extremely high component failure rates. For example, components such as vacuum tubes or relays had typical failure rates as high as one in a hundred thousand cycles.

Then there was also the issue of communication in noisy (hence error inducing) channels. In the 1960's several applications began to emerge in which error free operation was essential. Spacecraft computers and computer controlled telephone switching systems are representative examples of such applications. In 1970's, under NASA sponsorship, applications for space shuttle were required with still more stringent requirements on error avoidance.

In the 1980's, the applications in which extremely accurate performance is required, grew in number and variety. Thus one began encountering applications such as (a) computer controlled railway signaling wherein, if computer fails and disrupts the signal system, human life will be endangered, and (b) industrial control applications, such as control of nuclear reactors or plants producing hazardous chemicals, wherein computer operation error may have catastrophic consequences.

And now coming to the 1990's, in terms of computerized information systems one has applications such as e-commerce whose best known public face, amongst others, is buying books and trading stocks on the net. Applications as this are understandably very sensitive to any errors in computerized information processing as, due to the very nature of the technology, the error consequences can have their adverse impact across the supply chain.

- (p) Finally, there are some unique aspects that lend a special character to the problem of computer failures.
- (i) Firstly, faults in computer systems can be classified according to the criterion of duration, extent, value and the source of the fault. Fault may be labeled as “physical” or “human-made.”
 - (ii) The set of physical faults may be partitioned to the source. The source may be internal (eg. , a random breakdown of a semiconductor junction) or originate externally (eg. , a burst of electromagnetic radiation).
 - (iii) Further, computers contain some conventional electronic and mechanical components that are subject to failures.

To explain, major components of a computer are logic components. The number of logic components in a computer is very large, of the order of millions. Applications require that these logic components demonstrate very stringent failure rates at least as low as 10^{-6} to 10^{-7} per hr so as to render any useful service at all (Note: If a component or device fails, on average, once in every 1000,000 hours of operation, it is said to have failure rate of 10^{-6} per hr.). Thus, life testing, in terms of the number of components tested and the duration of the test becomes prohibitively expensive and time consuming. *As a result, the level of experience available for new technologies constantly emerging to provide faster and cheaper components, does not permit accurate estimation of component failure rates.* Computers are also characterized by a vast number of discrete system states. Even excluding the memory sub-system, a computer may contain 100 to 1000 binary storage elements allowing 2^{100} to 2^{1000} system states.

- (iv) Human-made faults are also separable in to two major classes. The first contains design faults that are due to mistakes during specification, design, implementation and modification of the system. The second contains interaction faults that are due to mistakes by human operators in assessing the system via control or maintenance panels during operation.
- (v) On the basis of duration, faults may be classified as permanent, transient or intermittent. Permanent faults are caused by irreversible changes in components. Examples are: shorted transistors, open connections and “bridging” between leads. Transient faults do not manifest themselves longer than for a specified period of time and are often caused by external interference. Intermittent fault occurs when a combination of rare events occur.
- (vi) In classification according to value, one may have deterministic faults (eg. , logic variable “stuck-at-zero”) or indeterminate faults, where variable alternates between possible values but not in accordance with the original design specification.
- (vii) The extent of a fault is determined by the number of logic variables Simultaneously affected by then fault in a different universe. Accordingly,

fault may be local or distributed giving consequent error propagation. Computer malfunctions are also varied in character. A system may fail due to single or multiple failures and the failures may or may not be independent.

0-0-0-0-0