Unit 1

Introduction to Information Systems

(Seminar)

Seminar 1 Preparation - En Route Automation Modernization

In this unit, I prepared considerations regarding the En Route Automation Modernization (ERAM) system failure. A brief outline of the failure is, the system went into a looped-restart cycle due to a failure to accommodate flight lack of path data from a U-2 spy aircraft that did not provide altitude data. The failure resulted in numerous delayed and grounded flights on the West Coast of America. Fortunately, there were no accidents due to this failure; however, after that, Flight Aviation Authority (FAA) required altitude data for every flight plan with increased system memory to prevent future failures.

The resolution was that a flight controller manually entered the expected altitude for a U-2 spy aircraft, ~60 000 feet. However, the system took the data and considered all altitude values between 0 and infinity, which generated multiple error messages and caused the repeated restarts.

Analysis of Failure

Considering the failure, a few points stood out as unbelievable, given the cost (over $2.4bn USD) of the system: (1) the price paid for software systems does not equate to increased reliability, (2) were this scenario considered during testing, likely, the failure would not have manifested, (3) error handling capability of the information system.

(1) **Cost**.

Too often, as consumers, we believe the more expensive an item is, the better “quality” it contains. However, in the world of software development, this cannot be further from the truth. Anything that involves human development is guaranteed to exhibit failure. This is because humans, by nature, are corruptible and therefore, our processes though abstract, yet manifested in code, exhibit the same capacity for corruption and failure.

It is unreasonable to expect system engineers to develop systems that never fail, given faulty inputs or lack thereof. For this reason, I often consider whether or not the reduction of information system failures is an underlying reason for the rise of interest in artificial intelligence (AI)? Regardless, no matter the cost to develop information systems, humans are key actors in the consumption and development of the systems. Moreover, I would argue that most of an information system’s costs go to the human effort required to test usage scenarios or construct the system.

(2) **Testing**.

Every time an information system fails, especially within the public domain, it is easy to raise questions about whether the system was tested or not. And rightly so. Because it is impossible to test every single process pathway in an information system, these pathways become the points of failure. Furthermore, they are merely points of failure because their testing plans exclude them from consideration. Again, to me, this ties in with the “human” nature in every business collective; people deal with test scenarios, test data and test cases. Until machines can consider every single pathway, software information systems will continue to exhibit failures.

The only reasonable contingency plan is, therefore, to communicate well.

(3) **Error handling**.

It was quite a curious idea to consider a system that went into a continual recycle loop when it encountered missing data. Curious indeed that somewhere along with the development and design phase, system engineers must have discussed how the system must behave in the absence of data. The evidence is their outcome was to “have you turned it off and on again?”. Suppose this approach was viable in general information systems. In that case, we would have to turn off and on all our software-driven devices such as servers, laptops, TVs, microwaves, remotes, phones regularly to be sure their process pathways work as expected. However, contrary to the ERAM failure, we do not do this. Such reality shows that an inherent requirement of any information systems must accommodate erroneous, incorrect or incomplete data packets.