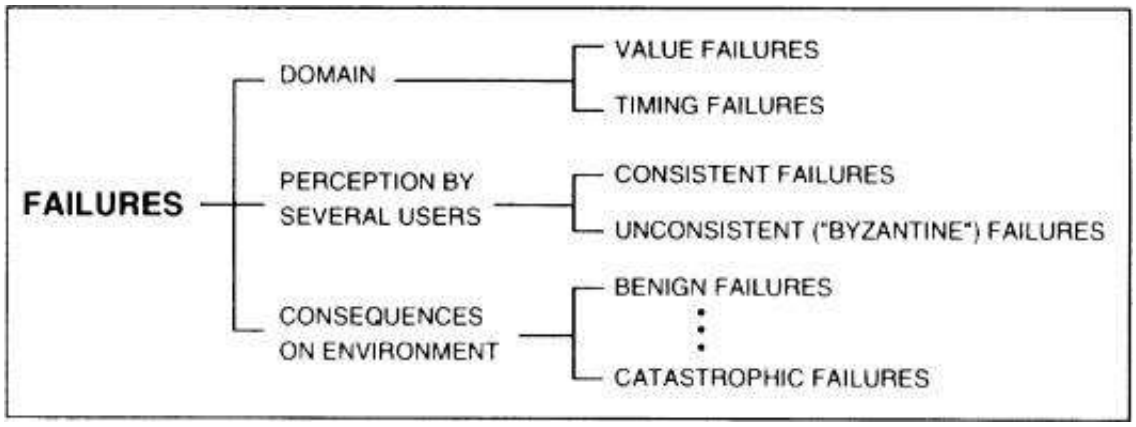
Anomalous inputs can be due to failed hardware, timing problems, harsh/unexpected environmental conditions, multiple changes in conditions and inputs that are beyond what the hardware is able to deal with. Bad user input may also be a source for such exception conditions.

A system may not, and generally does not, always fail in the same way. The ways a system can fail are its **failure modes**, which may be characterized according to three viewpoints:



1. **Domain:** leading to distinguish **Value failures** and **Timing failures.** A class of failures relating to both value and timing are the **halting failures**: system activity, if any, is no longer perceptible to the users. According to how the system interacts with its user, this absence of activity may take the form of:
   a. **Frozen outputs**: a constant value service is delivered; the constant value delivered may vary according to the application (e.g. last correct value),
   b. **A silence:** no message sent in a distributed system.

2. **Perception by users:** when a system has several users:
   a. **Consistent failures** (all system users have the same perception of the failures);

b. **Inconsistent failures**: the system users may have different perceptions of a given failure.

3. **Consequences on the environment**: Grading the *consequences of the failures* upon the system environment enables the **failure severities** to be defined. Two extreme levels can be defined according to the relation between the benefit provided by the service delivered in the absence of failure and the consequences of failures:
   a. **Benign failures**, where the consequences are of the same order of magnitude as the benefit provided by service delivery in the absence of failure;
   b. **Catastrophic failures**, where the consequences are much greater than the benefit provided by service delivery in the absence of failure.

7. Faults and Fault Classes

Faults and their sources can be classified according to **five main viewpoints** that are:

1- their *phenomenological* cause leads to distinguish:
- **Physical faults**
- **Human-made faults**

2- their *nature* of faults leads to distinguish:
- **Accidental faults**
- **Intentional faults**

3- their *phase of creation or of occurrence* with respect to the system's life leads to distinguish:
- **Development faults**
- **Operational faults**

4- their situation with respect to the *system boundaries* lead one to distinguish
   - **Internal faults**, which are those parts of the state of a system which, when invoked by the computation activity, will produce an error,
   - **External faults**, which result from interference or from interaction with its physical or human environment.

5- their *temporal persistence* leads one to distinguish:
   - **Permanent faults**, whose presence is not related to conditions
   - **Temporary faults**, whose presence is related to conditions whether external (environment) or internal (computation activity) and are thus present for a limited amount of time.

The notion of **temporary fault** deserves the following comments:

- *Temporary External Faults* originating from the physical environment are often termed **Transient Faults**;

- *Temporary Internal Faults* are often termed **Intermittent Faults**; these faults result from the presence of rarely occurring combinations of conditions.