"A beginning is the time for taking most delicate care that the balances are correct..." - from "Dune" by Frank Herbert.

1 THE PROBLEM OF "GETTING STARTED"

One of the difficulties with every problem modelling or systems analysis technique is getting started.

As soon as we put pen to paper, or even start to think about a problem, we have begun to make choices; the reverberations of which will be felt throughout the system development life cycle. Choices to be made include the size and scope of the system, the functionality that shall be included and even the identification of the correct problem to model and solve (in other words choice of the right subject centre). "Getting started" is not just how to start but what to start with and where to start from. Just starting can be a major hurdle to begin with.

This appears particularly true where software is involved.

So how can we get started without making too many artificial choices which will affect the system?

In the past, the idea of functional decomposition was considered to be reasonable. However, this method is very subjective and no better than a wet finger in the air when it comes to identifying system behaviour.

Functional decomposition is argued, by Heitmeyer and McLean [PALM84], to be implementation dependant since a more-or-less arbitrary choice among a set of possible decompositions has been made. This may lead to low level functions appearing at unnaturally "high" levels of abstraction, leading to disorientation on the reader's behalf.

Steve Meller and Paul Ward [WARD85] went further to point out that functional decomposition does not guide the developer towards partitioning a system so as to minimise interface complexity.

A basic myth about system decomposition is that there is a natural partitioning for a system. There is not. Partitioning of systems is for our own benefit so that we can appreciate the problem. Any partitioning is a
distortion of the problem being modelled and we can only hope to minimise this distortion.

Functional decomposition does not achieve this goal consistently.

2 ALTERNATIVES TO FUNCTIONAL DECOMPOSITION

The first problem appears to be, what is "The System".

One has to remember that there is no such thing as a system which is not a part of, or sub-set of, another system which is itself a part of, or sub-set of, another system... So the only way to make the term "System" meaningful is to limit the scope or context of it.

A system is an artifact of our observing it to be a system.

In other words any system we care to study will be artificially created by ourselves. We do this by including things of interest to us and omitting those things we find of no interest.

In 1975 Tom De Marco [MARC75] suggested the idea of a System Context Diagram (Figure 1). The Context Diagram is used to limit the domain of study. In other words it limits the scope and therefore the size of the system. The Context Diagram shows sources and sinks of data. "A source or sink is a person or organisation lying outside the context of a system that is a net originator or receiver of system data."

The Context Diagram, therefore, shows the System, the terminators that form part of the system environment and the interactions between the system and its terminators.

![Figure 1. Context diagram for lift system](image)