



# Why do we need DFSS ?

We need design methodologies like DFSS because we live in a complex world; new products and processes have many opportunities to ‘go wrong’.

If they go wrong, then DMAICT is needed to fix them – or a complete re-design becomes necessary.

This is too late and too costly.

DFSS provides a systematic **proactive** approach to design which is backed up with **quantitative** predictive design tools which lead to **quantitative confidence** in the performance of the new process or product prior to implementation.

## **For examples of new system (product/process) failures look around you . . .**

**Are you happy with the service you receive from your bank or goods suppliers?**

**Do you think that they have good customer-oriented systems in place?**

**Are you happy with your washing machine or cooker?**

**Do you find them easy to use, with good performance and reliability?**

**Do you find yourself frustrated by things which 'ought to work' but are clearly 'not fit for purpose' or things that you need 3 hands to use?**

A number of high profile new IT system failures in the public sector have been well-documented by the media – millions of pounds spent on new computer systems which subsequently do not fulfil customer requirements.

There is no reason to believe that new process/system implementations are any better in the private sector either.

New cars have been re-called by manufacturers for safety concerns.

There are lots of other examples of things which don't work to the satisfaction of their users when implemented from new.

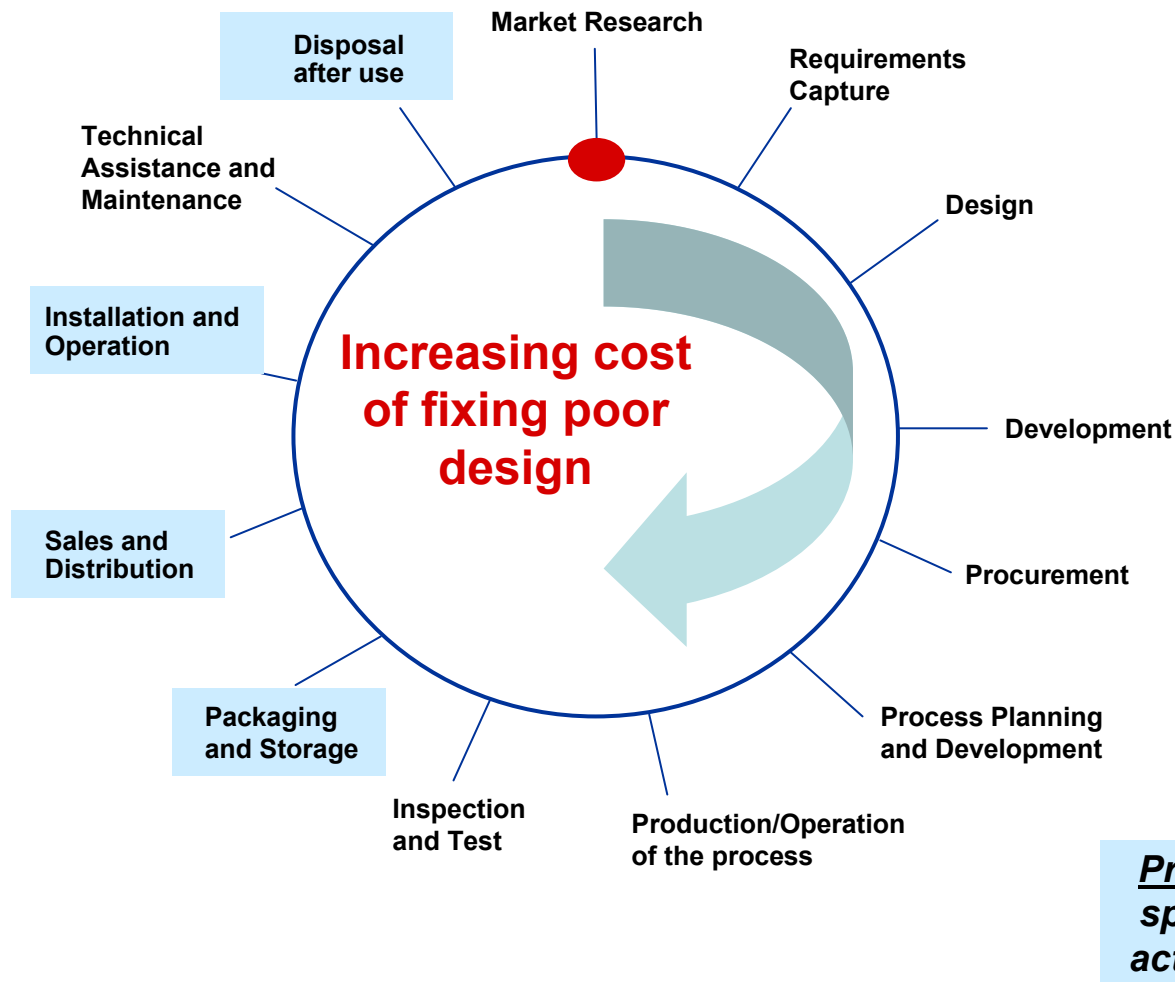
## Why the difficulty?

New process/system design requires:

- a lot of thought and careful planning **up front**.
- careful and comprehensive attention to customer needs.
- that customer needs are 'flowed down' (integrated) into the design and that this has been **verified** to be the case **prior** to implementation.

**This is what DFSS helps you to do !**

# Getting it right – designing for the product/process life cycle



- Errors if left undetected will be built upon in subsequent activities – potentially making them harder to find and more costly to fix.
- DFSS focuses on doing more work up front but involving every function with a stake in the life cycle to ensure that the design captures and delivers all requirements.