



# My Key Targets and Pitfalls Checklist


A Ulysses Systems Publication


## Key Targets and Pitfalls


## Comments


-  [Low cost software is future proof](#)


Low cost software is the right target. And you need software that will cost you far less because it lasts forever by being future proof. Future proof software not only cost less. It also eliminates the need to disrupt your staff and working methods during the process of changing software.
-  [Software that costs you less in preoccupation and valuable attention](#)

Low cost software is software that costs you less in preoccupation and valuable attention. You have highly paid people handling software on board and ashore. Naturally, priority number one is to have the software cost you less in wasted time and preoccupation. So, judging whether the software is sufficiently well thought out for your precious staff is a real challenge. It is an important area of concern that cannot be resolved by training. Just like poor engine room layout or poor bridge layout cannot be resolved. It will grind down your productivity forever
-  [The highest cost in marine software is the time you spend doing the transaction using the software.](#)

Time and cost of using software are about **3 x the out-of-pocket cost of the software**. This is for the best of breed software. For software that covers the functionality like a tick box or has not been improved for years, this can be three times higher, so **9 x the out-of-pocket lifecycle cost of the software**. So as to give an idea, the calculations are quite simple. How long do people spend on each application per day? The chief engineers the superintendents and the purchasing specialists? You will find that that cost in labour is about 3 to 9 times higher than the cost of the software spread over the lifecycle, including every out-of-pocket payment you make for the software.
-  [Restaurants with big menus often disappoint!](#)

The marine industry could do with 100% digitization. But let's not confuse ourselves. Because one vendor can build everything for us, it doesn't mean it will all be effective. We must therefore select the best software or burden our staff with mediocrity. After all, restaurants with big menus often disappoint!
-  [Integration is underrated](#)

In every domain, integration of software is the most underrated feature. But especially in marine software, where integration should be a first-class feature.
-  [The best plans are those you can improve](#)

The best plans are those that you can improve because the plan was made with improvements in mind. It follows that the best enterprise software continuously improves, and everything can be replaced in regular upgrades. The goal being, of course, to keep the software future proof.
-  [A single version of the data and code](#)

A single version of the data and code is the result of a well thought out design. Adding data and coding that deviates from a single line of code is like uncoordinated planning or worse still uncoordinated construction.

# My Key Targets and Pitfalls Checklist

A Ulysses Systems Publication

## Configuration beats customization

'Customizing' as opposed to 'configuring' software is like your software vendor changing plans midway between software upgrades. Better then, the vendor who anticipates where you will need improvements and has ready configurations to match differing processes. This way, you configure the software, you don't cement it.

## The marine industry needs its own well thought out co-ordination processes

The highest value that software can provide in the marine domain is awareness and co-ordination of decisions and actions. The software you buy has features that can affect the co-ordination principles of your company. The marine industry needs its own well thought out co-ordination processes. **So, must it be pushed into the way other industries work?**

## Co-ordination needs wisdom and know how

This is why chief executives, army generals and football coaches are well rewarded. But they all need good co-ordination tools that can integrate with all current and future solutions you might choose. So, check for what your vendor has foreseen in way of coordination tools

## User interfaces

Good user interfaces are like good discussions. And the participants in the discussions will know the subject matter well. Given this, conversations require a good grasp of relevance and priority. And good conversationists lay out the discussion with this in mind. Unsurprisingly, there are no magic pills for a layout. But it may seem surprising that the user interface design challenge is not in the technology. The challenge is, actually, a problem as old as people having conversations. So, we must make sure our software vendor understands the concepts behind a good user interface. Above all, it's a lot more than the design of the buttons we see in the UI.

## Good software requires a good platform

Good software is made by using a good platform. Just like making a good car. It needs a good production line to articulate in the right way.

## What is free software like?

Free software is often like free cement. You have to be really careful in case it is really hard to recast or remove.

## Testimonials

We should seek testimonials from actual users of the software that are willing to provide their names in the public domain. These are testimonials that are public statements and can become valuable validators of the real life cycle cost and real usability of the software.