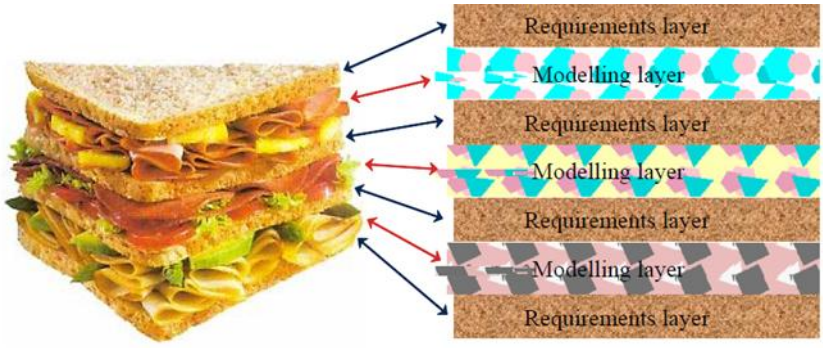
The Requirements Sandwich

The idea of the "requirements sandwich" was introduced in a paper by Dick and Chard at INCOSE IS 2004 (so a long time ago) called *"The Systems Engineering Sandwich: combining requirements, models and design"*. Its key figure is reproduced below:



I have referred to this "concept" many times, most notably in IS 2018 paper "*Requirements Management or Assurance Management – Keep the Data Integrated*" co-authored with Iain Cardow and Andy Pickard. It did validate an idea of mine that had helped get me the role of Head of Systems Engineering in Defence Aerospace in Rolls-Royce in 2006 – namely that the "left hand side of the V" was not a straight line but a zig-zag as the requirements for the next level should be grounded in an traceable from the design of the layer above. I will discuss separately the full details of information management, and the order it can be developed in – suffice it to say everything does not have to be created top-down, but eventually it looks if it was / could have been.

But this model, and discussion of it with Jeremy Dick, yielded what (at least for me) seemed like a profound insight into the nature of Systems Engineering – by questioning the detail of the illustration (I know it is only an analogy and you shouldn't push a metaphor too far, but I did and it was useful.

In common usage when describing a sandwich it is described as a "something" sandwich (a ham sandwich, a cheese sandwich etc.), with the "something" being the filling, and the bread is not mentioned. So I thought that this meant in this model the "requirements" were the filling and the figure shows it as the bread. So I initially speculated that the analogy was miss named. Starting with requirements certainly made sense in the process view of Systems Engineer - start with defining requirements (always understand problem before solving it is a starting point for standard SE explanation) and work top down through architecture, decomposition of requirements through the layers of the system. Ending with a requirements layer (the bottom slice of bread) for me implied the final engineering drawing defining how the part for to be made (I see outcome of design engineering as always being description and instruction for the actual system). So this use of requirements as the "bread" made me think model misnamed.

However, I do believe that the name is correct and the figure needs redrawing. In terms of engineering a solution thinking that you start and end with requirements makes some sense (in terms of the engineering activities). BUT – the solution system is always in a real environment, so

1) The top level of the sandwich is the environment into which the system with go, and so the requirements are derived from that environment and what impact / change the system of interest is desired to have on that. So the requirement for the top level is the first filling.
2) From the first "filling" of requirements system architecture and analysis defines the elements of the next layer. This is next layer of "bread" holding the sandwich together – ad defining the next layer of requirements "filling".
3) This repeats until the bottom layer
4) At the bottom, the analogy can become strained, but you can argue that from the requirements for the lowest level of elements (the last filling) the actual (real or physical) parts come that are integrated together and form the system that is deployed.

To me this helps me see that the solution system that is created by a systems engineering process, at any level, is always grounded in the real environment – and it is from that environment requirements are derived (and must be traced to) and ultimately it is a created actual thing that goes into that environment. So I "expand" the idea of a modelling layer into a solution layer that includes the modelling, architecture and analysis of the system at that layer.

I need to draw this requirements sandwich in my terminology view point.