The art of simplicity

"for the architect’s inspiration"

Edwin Zwart

Today’s systems are utterly complex. Architecture is the answer to provide order in this chaos. An architecture is your compass for change, with the purpose of planning investments, development and system change in a managed way. Architects should strive for simplicity in their designs, such that they can be understood by the relevant audiences. As such, simplicity is an extremely relevant notion to information architecture, which already has a strong, if implicit, influence on the field.

This article attempts to give an explanation of simplicity and to inspire you as an architect to think this notion through. This then is my view on how to achieve designs that fall in the category of "simplicity", that is, designs of systems that can be understood by the relevant audience(s), and additional audiences, if needed. Much of my interpretation is derived from the philosophies or religions on Taoism that were composed by Lao Tse or Lao Tzu.

Simplicity defined

Single; not complex; not infolded or entangled; uncombined; not compounded; not blended with something else; not complicated; as: a simple substance; a simple idea; a simple sound; a simple machine; a simple problem; simple tasks.

Simplicity is a more qualitative word connected to simple. It is a property, condition, or quality which things can be judged to have. It usually relates to the burden which a thing puts on someone trying to explain or understand it. Something which is easy to understand or explain is simple, in contrast to something complicated. In some uses, simplicity can be used to imply beauty, purity or clarity. Simplicity may also be used in a negative connotation to denote a deficit or insufficiency of nuance or complexity of a thing, relative to what is supposed to be required.

From the short phrases above speaks an element of singularity in simplicity: striving for simplicity in a system means that no elements are unnecessary or unnecessarily large, convoluted or complicated. There is nothing in there that does not directly serve a purpose. As such, the idea of simplicity encompasses efficiency. But the simple solution is not just superior for the sake of efficiency alone. Less is more, and there is value in an increased understandability of the system, when it is simpler.

In addition, there is also an aesthetic element to this principle. There is an elegance and beauty in simplicity, as we often experience the most straightforward solution to a problem, or the simplest system, as the most beautiful one. In modern art, artists seek to find beauty in simplification through abstraction, and modern art movements do try to achieve the simplest representation of a particular feeling, impression or stimulus, through mediums such as painting or poetry. Leonardo da Vinci has been quoted to feel that “Simplicity is the ultimate sophistication.” J.S. Bach strived for simplicity in his music, specifically the pieces in “The Well-Tempered Clavier”, through repetition and iteration of (variations of) the same musical
structures. Another famous composer (Frederic Chopin) once said: “Simplicity is the final achievement. After one has played a vast quantity of notes and more notes, it is simplicity that emerges as the crowning reward of art.”

The concept of simplicity has also been related to truth in the field of epistemology, and as such can be applied to science in general. According to Occam's razor, all other things being equal, the simplest theory is the most likely to be true. This is a widely held belief in the scientific community at large. Albert Einstein has summarized this aptly, by saying that: “Everything should be made as simple as possible, but not simpler.”

This may give the impression that simplicity is easy to achieve, as all one needs to do is remove all parts of the system that are not strictly necessary. Unfortunately, the contrary is true: simplicity is very hard to achieve, since it is the core structure of the system as whole and its sub-systems that determines its potential for simplicity.

**Designing simple systems**

Now let me apply this more specifically to the design of an (architecture) system.

Every design starts by defining the system at hand, and we should give it a name that is easily understood by those concerned. This will take some time of problem and goal analysis, until we understand what needs to be solved and be designed in terms of a solution (or alternative solutions). Once we have agreed upon a name for the system, we should develop a short definition, that describes the system as agreed upon. The description will have some themes that each describe the essential parts of the system in just a few words. And these essential themes are the triggers to decompose the system into parts, parts with much cohesion and little dependence on each other. This is much like what we call today (and especially in the 80s) the object oriented approach for developing software systems. But this approach can (and should) be applied to all systems, if we want to achieve "simplicity". In this approach, the focus on essential parts of the system is an important factor in achieving simplicity, as this means that from the start only necessary elements are added in such a way as to keep the structure as straightforward as possible.

Once we have an idea of what the essential parts of the system are, which we call the sub-systems, this process for achieving simplicity is repeated for each sub-system. During this process, iteration will step in, meaning that the system will be redefined as we have become familiar with its parts; this leads to an improved system and its parts. Also, it is important to define the relationships between the parts at every level; these can be viewed as sub-systems as well.

Ideally, the decomposition of the system, in case of software systems for example, should be limited to say, 5 levels of abstraction and it will give you the architecture of the system. The number of abstractions per sub-system need not be the same as abstractions are not a goal in itself, and the complexity per sub-system will be different creating a differing necessity for abstraction levels per subsystem.

We should take into account that the decomposition covers the aspect areas of processes, applications and technology. Also, we may need to describe several points of view as related to the different types of audiences (for example: the client, architect and designer). Something that may seem simple to one person may be much more complex to a different audience.

Simplicity is already implicitly supported by and strived for in the methods that we use for Architecture. These methods reflect years of experience, and are tried and tested. However, on each occasion, we need to customize our approach. And methods have a lifecycle. They should be updated with our learning from new experiences, and adapted to changing situations and circumstances. In this case, it is important to keep striving for simplicity, as it is tempting to just adapt by adding new parts to the system that are devised to address a particular change. In order to achieve a simpler design, it is important to also keep looking at the structure of the system as a whole, and wonder if more simplicity could be achieved through changes to its core. Simplicity is also supported by the tools we use. Qualified, professional people are needed who have an overview of the system as a whole as well as its constituent parts, so that
simplicity can be strived for on all levels throughout the lifecycle of the system. Simplicity never ends, and should be a continuous focus of the designer of any system.

Edwin Zwart
edwinzwart@hotmail.com