Core Concepts

Mark A. Bools

2022-09-03

Abstract

Presentation introducing core concepts series.

Concept

From Merriam-Webster online; ‘an abstract or generic idea generalized from particular instances’

I believe there are several core concepts with which IT professionals should be familiar. These concepts are almost universally applicable, whether talking about design, documentation, team structure, methodology, tools, whatever, these concepts will be of benefit.

We will take the following concepts and examine each in detail, showing how they apply to several disciplines within IT.

Concepts

- Cohesion
- Coupling
- Separation of Concerns
- Abstraction
- Scope
- Context
- Contingency
- Entropy

Cohesion

Cohesion and coupling (dealt with in the next section) tend to be presented together when teaching software development. They are, in this context, often inversely correlated. I'm separating them a little because I think each concept has a lot more to offer than the formal presentation in software engineering suggests.

Cohesion

Plain English
When things work well together.

You want to increase cohesion (things working well together is good).
More detailed article.

Coupling

Related to cohesion, coupling is an expression of how difficult things are to separate.

Coupling

Plain English
How difficult things are to separate.

You want to decrease coupling (things being difficult to separate is bad). The more coupled things are the more difficult it is to replace one without effecting the others. Highly coupled entities are troublesome because the effectively lock you in to a particular solution.
More detailed article.

Abstraction

Abstraction allows us to focus on the essential details without worrying about implementation details. It also allows consumers of the abstraction to access the essential features without needing to know about how those essential features are actually implemented (it simply does not matter). Additionally, essential features are typically common to more than one underlying 'concrete' implementation.

Abstraction

Plain English
Extracting and presenting common essential features.

As you might imagine abstraction is an essential feature of reducing coupling. Abstraction layers provide a stabilising 'barrier' between the consumer and provider of concrete implementations.

More detailed article.

Separation of Concerns

A form of abstraction, separation of concerns is related to cohesion. Generally, things that show more cohesion tend to also show a separation of concerns.

Separation of Concerns

Plain English
Do one thing, do it well, and mind your own business.

More detailed article.

Scope

Most things have a range, an 'area of effect'. This is their scope.

Scope

Plain English
How applicable is it?

Scope is often expressed by space and time; over what space is the entity effective and for how long?

More detailed article.

Context

Complementary to scope, context is the environment. Specifically it is the immediate environment that affects the thing under consideration.

Context

Plain English
The environment in which it works.

Everything operates in an environment and this environment tends to effect those operations.

More detailed article.

Contingency

I use this one a lot. There is almost never *the* answer but rather *an* answer.

Contingency

Plain English
"It depends"

More detailed article.

Entropy

The universe is winding down, so is your IT system. The universe overcomes this 'winding down' locally by concentrating energy. We overcome the degradation of our systems by injecting energy in the form of work guided by process and procedure.

Entropy

Plain English
Without effort everything gets more disorganised.

More detailed article.