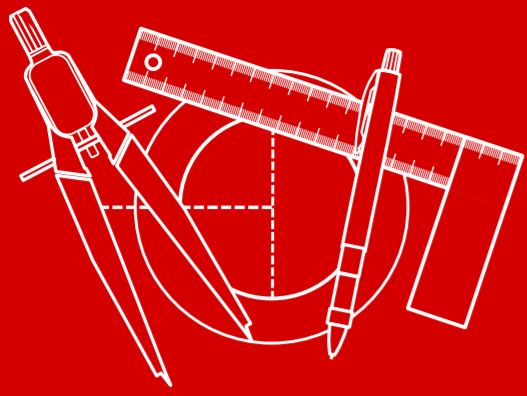
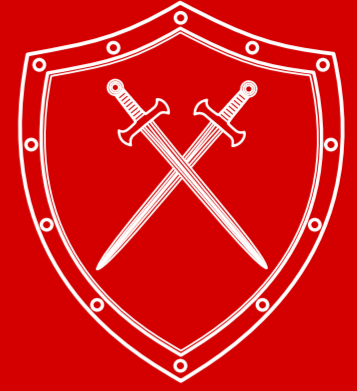


Many attacks happen on the application layer where infrastructure security solutions are not effective. Application Security is important, however, still rare in application development. Let's change that!

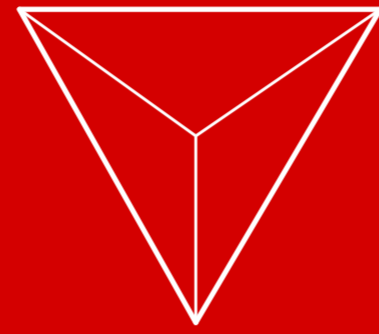
## Three useful notions for proactive Application Security!



# 「Shift Left」



# 「Security by Design」



# 「Continuous Hardening」

A security story so that 「bottom-up」 meets 「top-down」

### Shift Left

Mindset for increasing efficiency and controlling cost during the whole application lifecycle.

## 「Shift Left!」

Find and fix defects early for reducing cost and risk  
Handle it like Product Quality or Safety!

### Top-Down Support

Efficient Application Security is not feasible without strong support from top management!

### Application Lifecycle



Time when finding a defect: .....

Damage and cost for fixing it: .....

### Rule of Thumb

The later it gets, the more expensive it is to change an application (getting significantly costlier!)

### Top Management Involvement Required

Keep an eye on the realities in your organization regarding, 「Shift Left」, 「Security by Design」 and 「Continuous Hardening」!

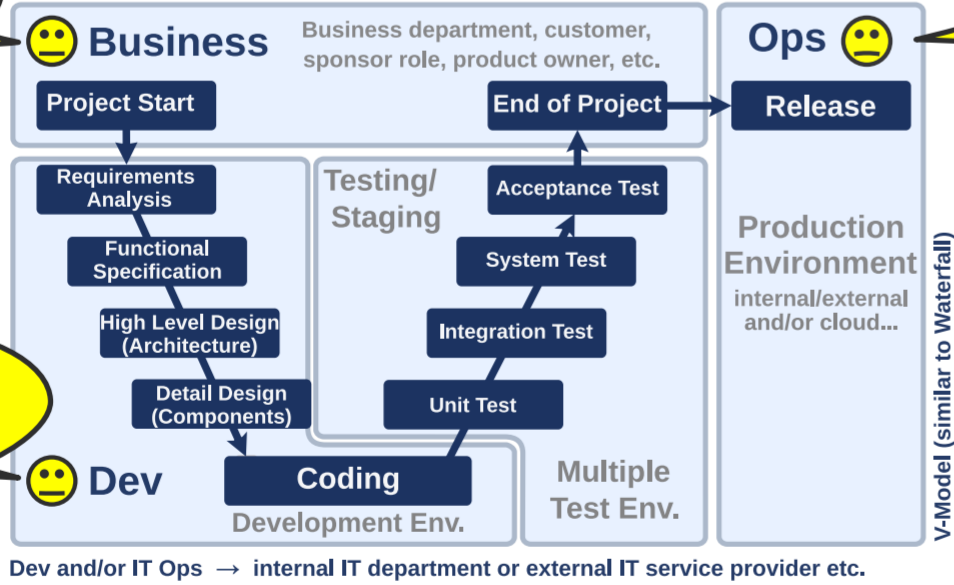
Years ..... Months ..... Delivery Time

The Universe of Development Methods

Delivery Time → ..... Weeks ..... Days ..... Hours

## Traditional Dev. Methods

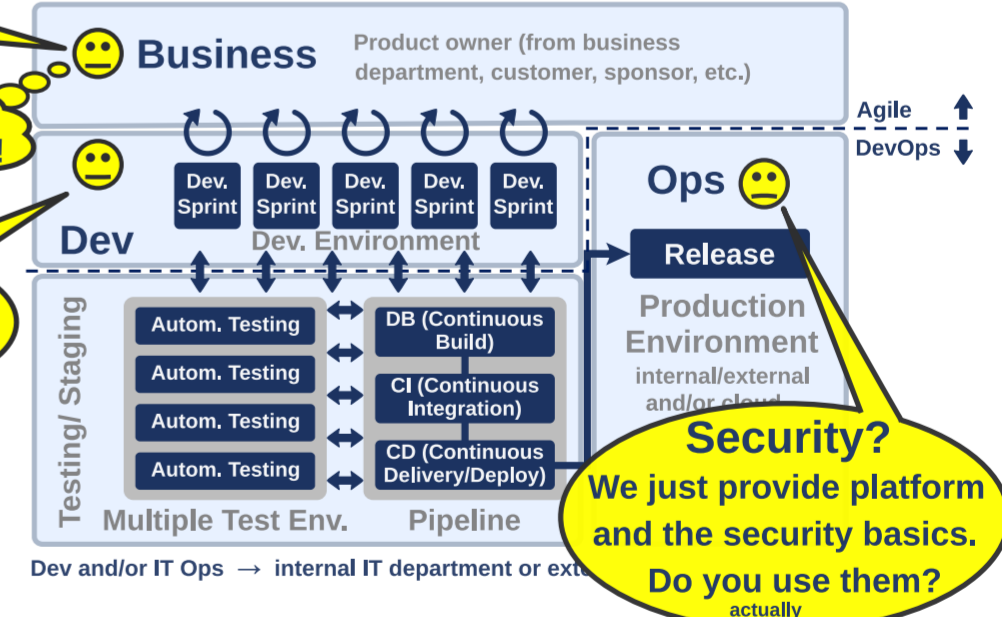
Waterfall approach at one extreme end in the universe of development methods.



Security? We just provide platform and the security basics. Do you use them? actually

## Agile Dev. & DevOps

Agile Dev. & DevOps combo at the other extreme end in the method universe.



Security? Leave it to IT...  
Wow! It's really fast now!  
Security? Leave it to IT Ops...

## Tragedy in the IT Trenches

Still an often seen reality!

## Add 「Security by Design」 & 「Continuous Hardening」!

## Better Approach

1. Business requires a proactive approach and provides budget.
2. Development project creates a security design. Suggested starting point: Zero Trust.
3. Development and other teams work together for efficiently

- a) implementing Continuous Hardening.
- b) maintaining the intended security level afterwards in order to stay consistent over time.

