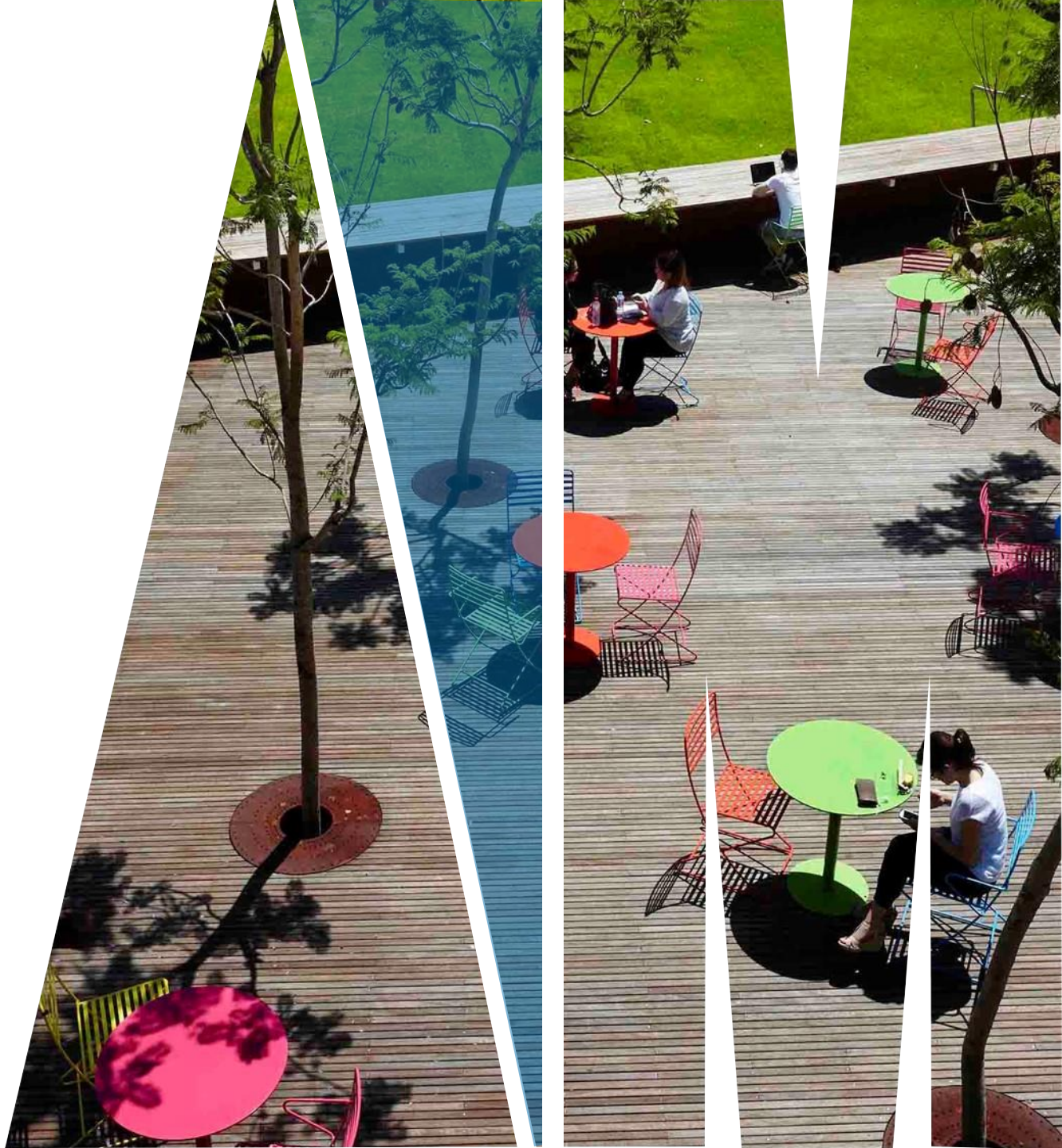
MONASH University

FIT2099 Object-Oriented Design and Implementation

Technical debt



MONASH University

EXAMPLE OF unCONSCIOUS DESIGN

Source: The Daily WTF
http://thedailywtf.com/articles/dictionary-definition-of-a-loop

```
private static double FindInterestRate(int operationYear,
                                       Dictionary<int, double> yearToInterestRates) {
    //where 0 is the first year
    if (operationYear < 0)
        return 0;
    else {
        for (int i = 1; i < yearToInterestRates.Count; i++) {
            if (operationYear < yearToInterestRates.ElementAt(i).Key - 1)
                return yearToInterestRates.ElementAt(i - 1).Value;
        }
        return yearToInterestRates.Last().Value;
    }
```

It's treating a Dictionary as a list or array rather than a Dictionary. This method is unnecessarily complex most likely due to some **earlier design commitments**.

WHAT IS TECHNICAL DEBT?

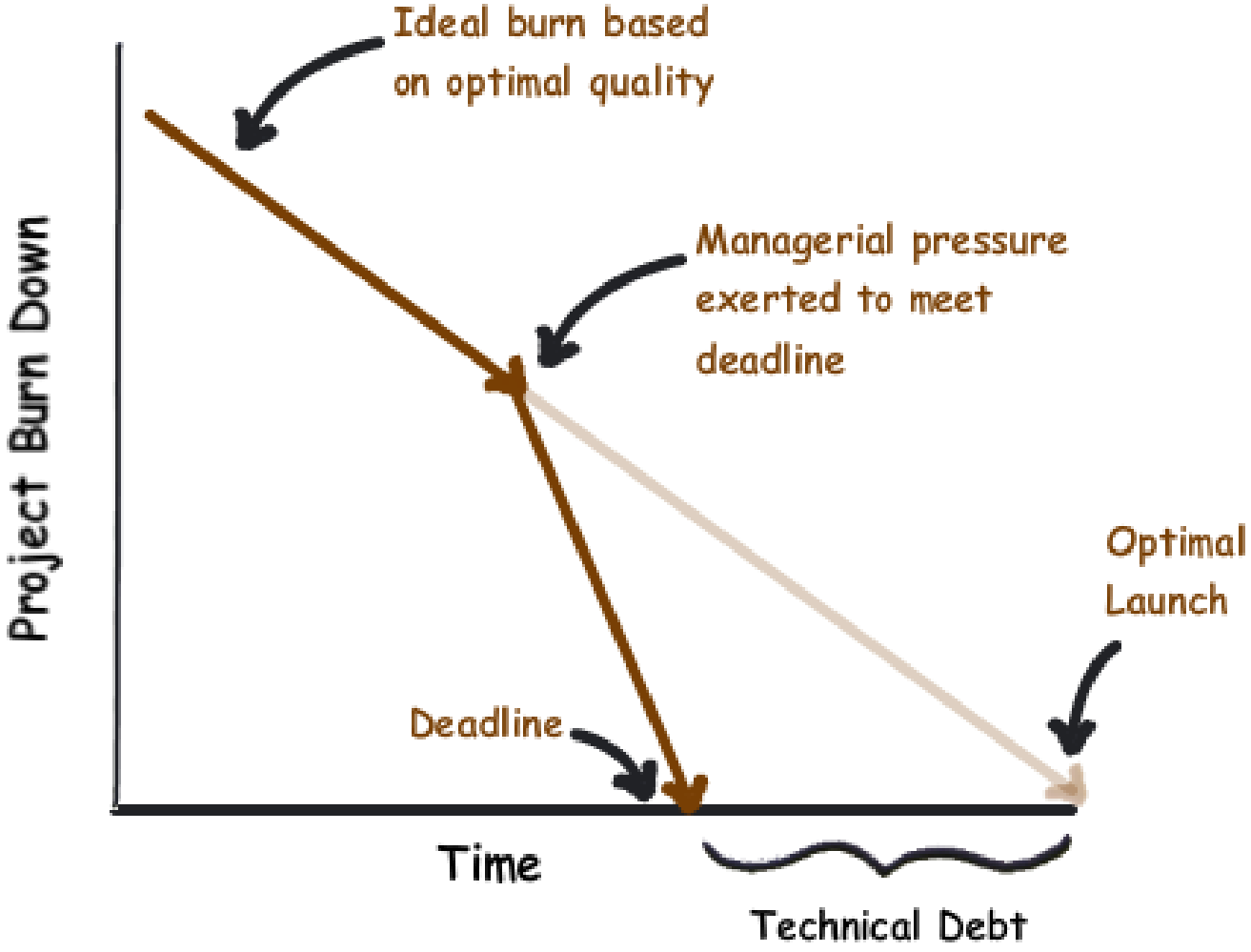
"You have a piece of functionality that you need to add to your system.

You see two ways to do it, ***one is quick to do but is messy*** *- you are sure that it will make further changes harder in the future.*

The other results in a cleaner design*, but will take longer to put in place.*

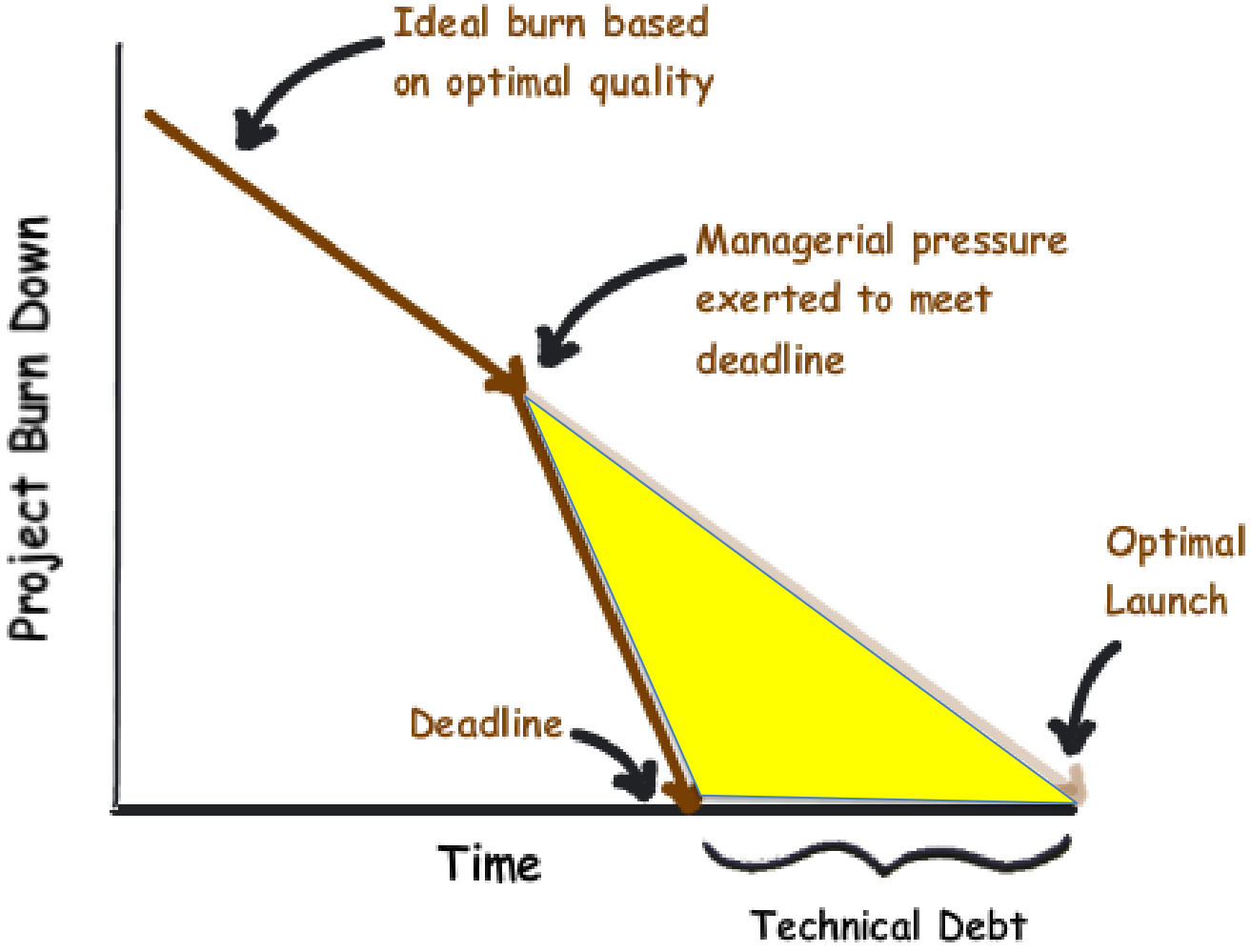
– Martin Fowler

WHAT IS TECHNICAL DEBT?



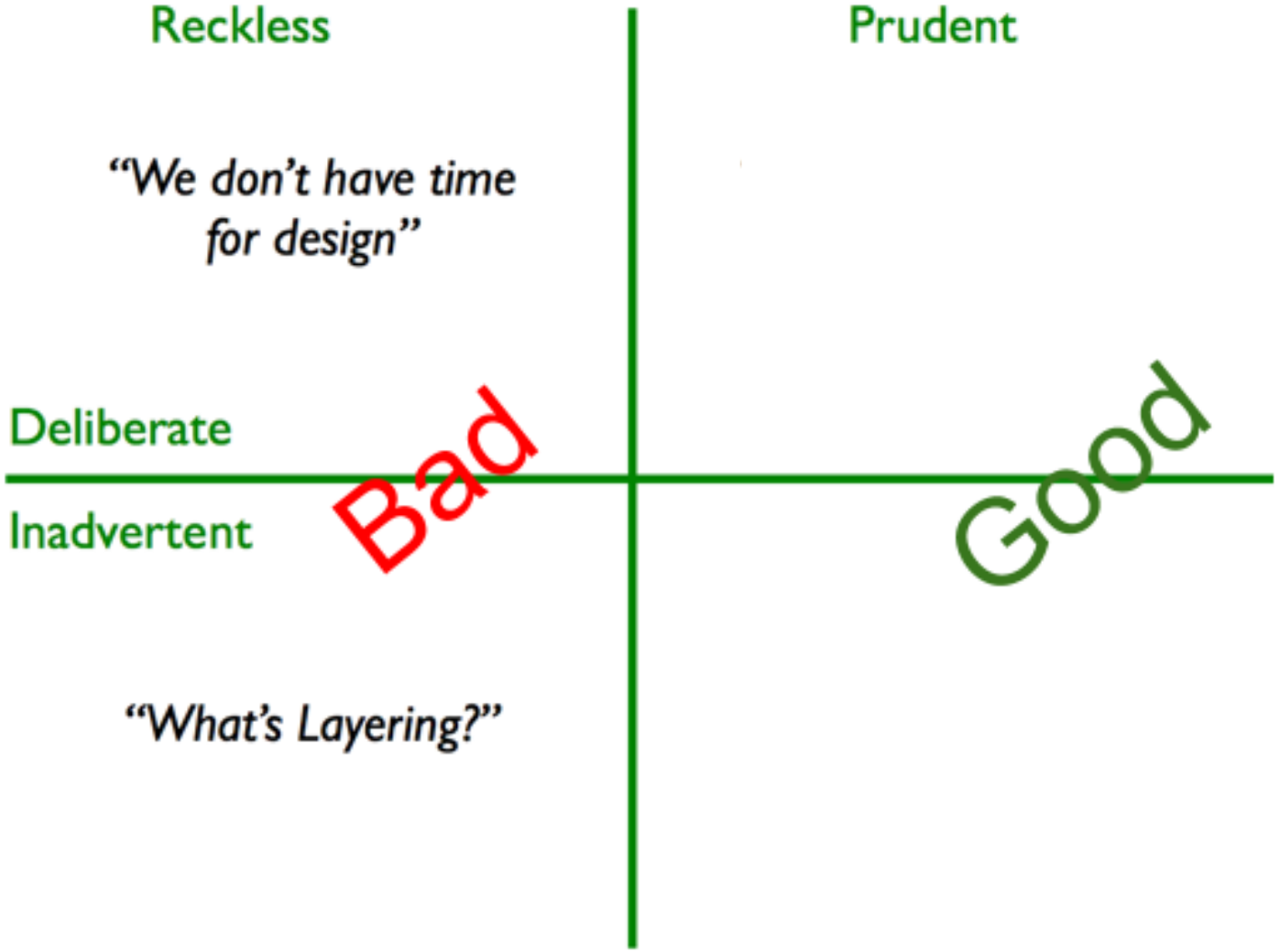
Source: commadot.com

WHAT IS TECHNICAL DEBT?

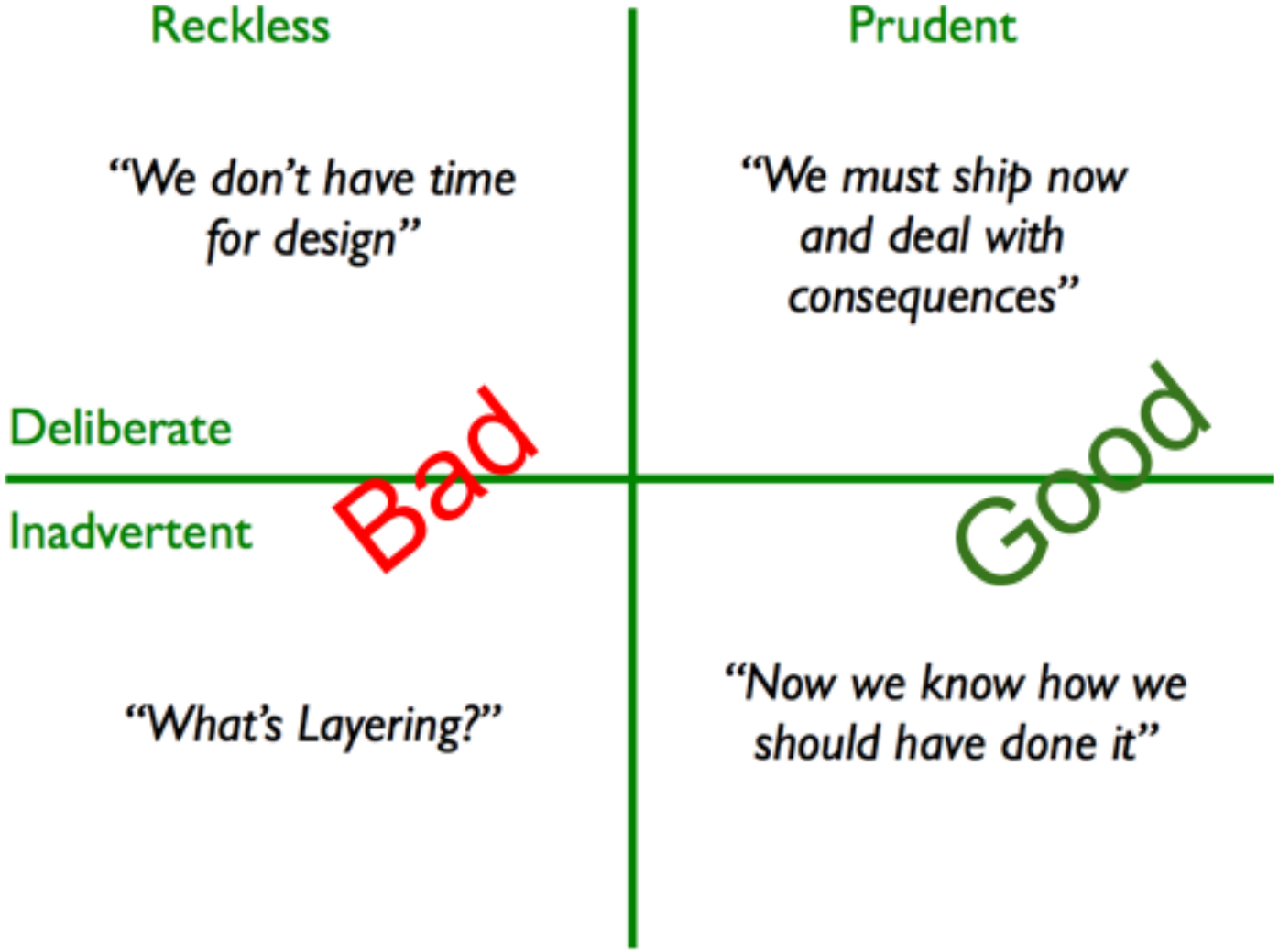


Source: commadot.com

TECHNICAL DEBT
MITIGATION



TECHNICAL DEBT
MITIGATION



WHAT IS TECHNICAL DEBT?

is **inevitable** in real systems

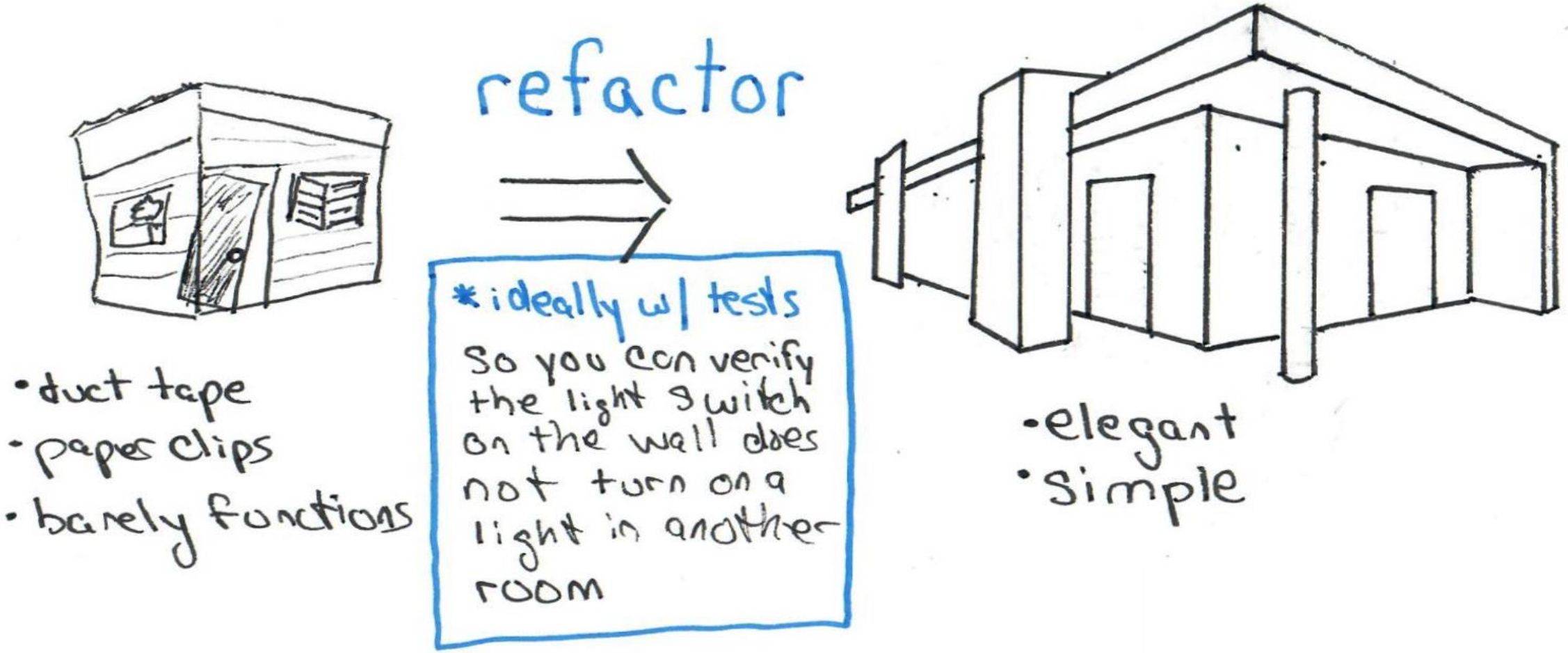
- it's not bad to go into technical debt, any more than it is to borrow money to improve a business

happens because sometimes we need to get features out the door (**deadlines**)

can be "repaid" by refactoring

- **but you need to refactor to a better design**

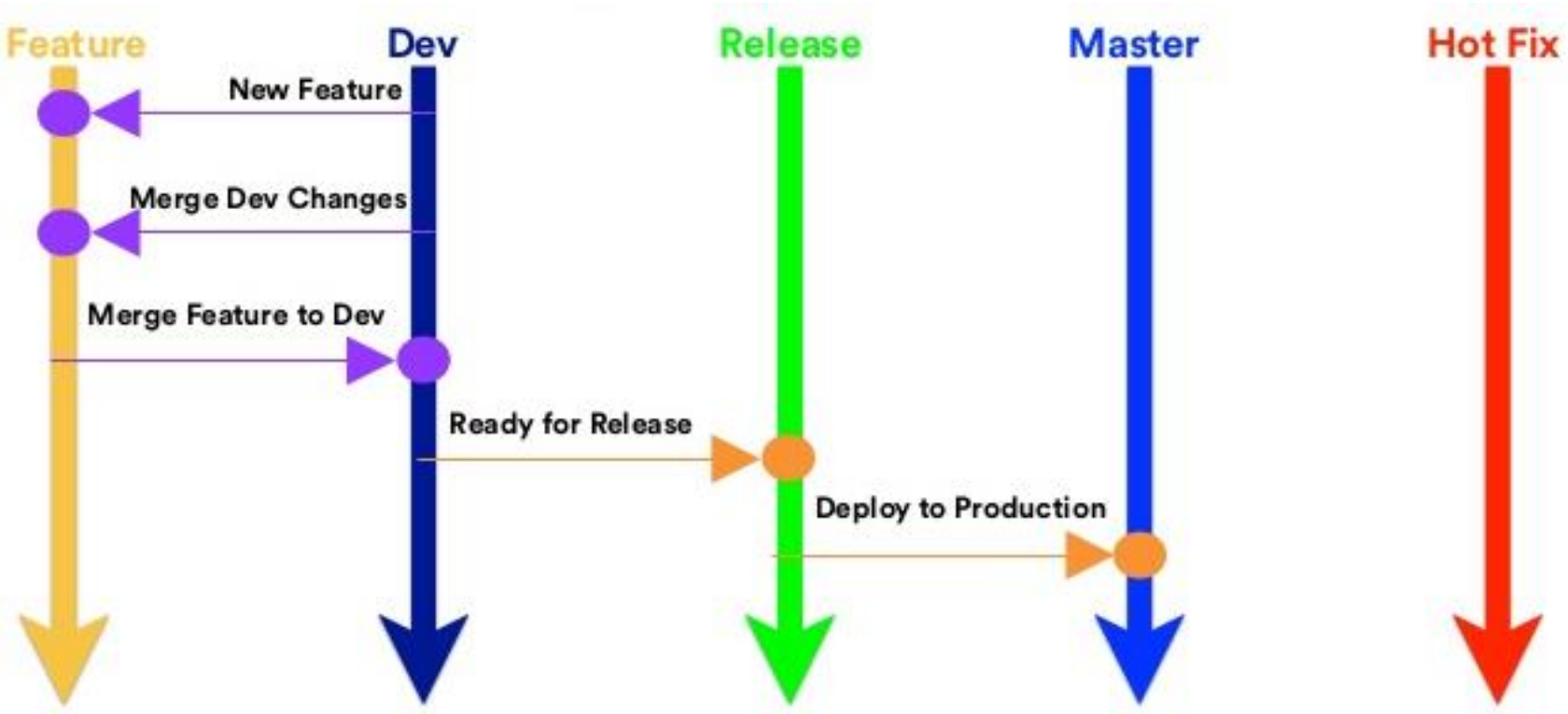
WHAT IS TECHNICAL DEBT IN RELATION TO REFACTORING?



Source: justideas.io

TECHNICAL DEBT IN LEAN

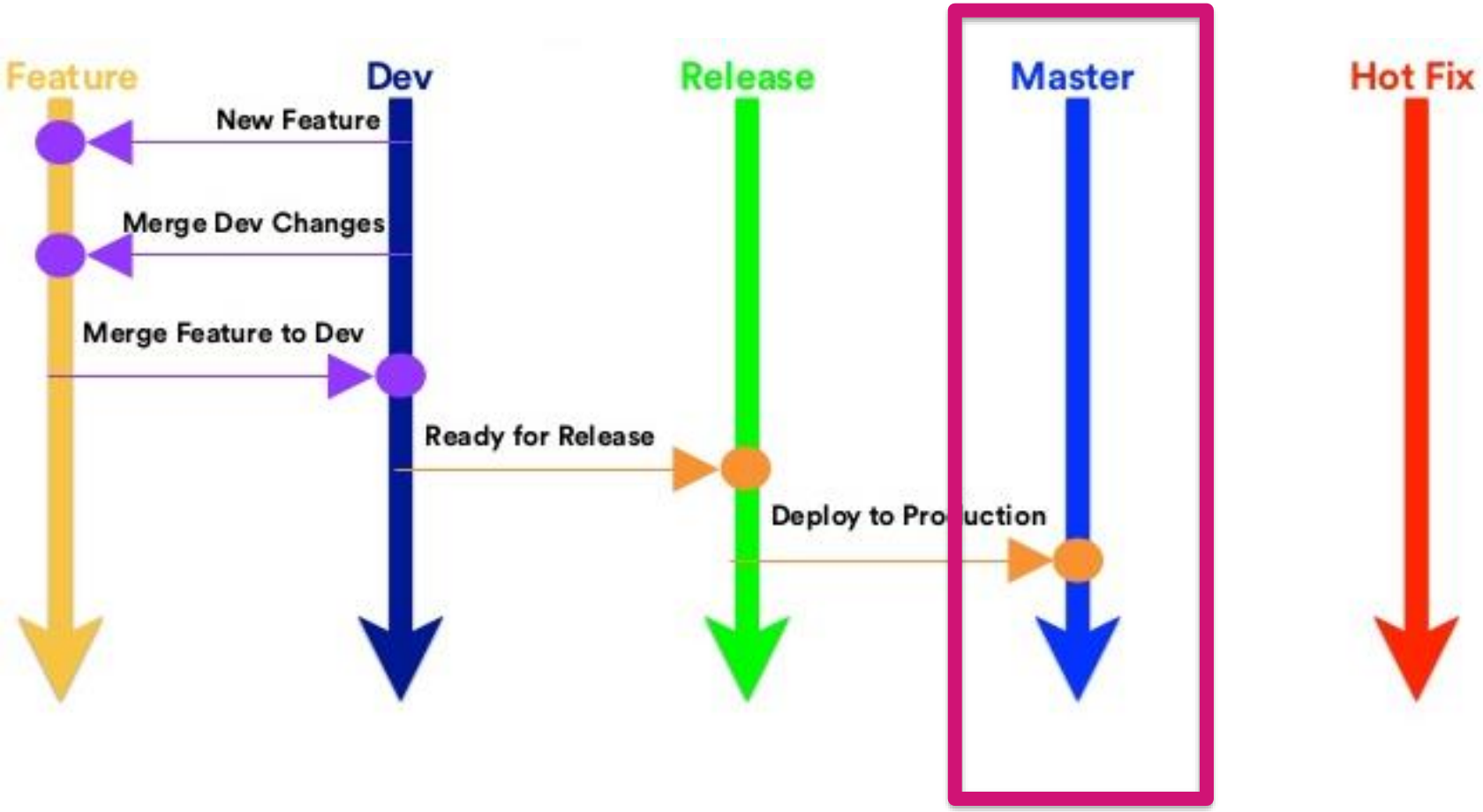
FEATURE BRANCHING STRATEGY



Source: Atlassian

TECHNICAL DEBT IN LEAN

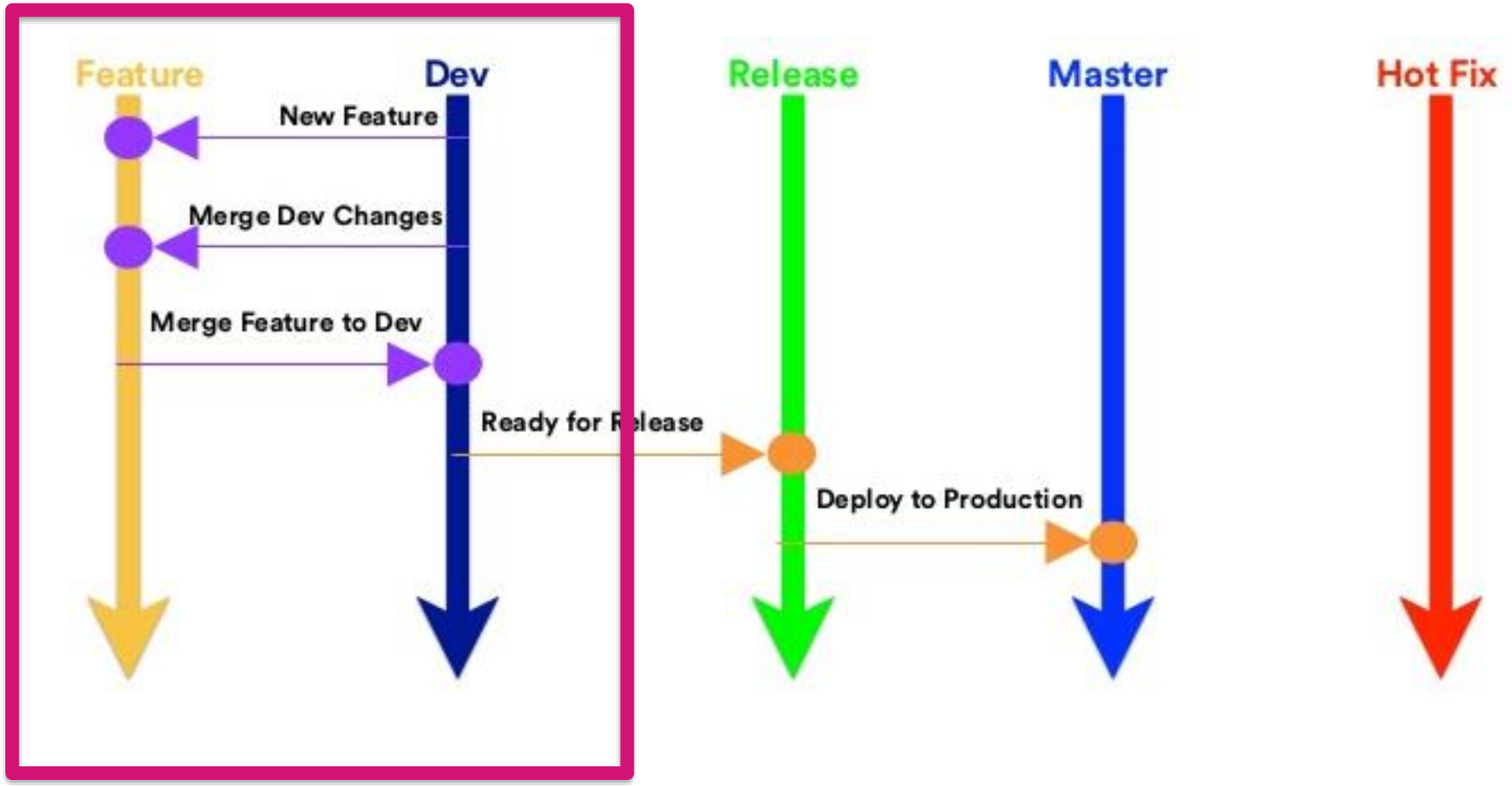
FEATURE BRANCHING STRATEGY



Source: Atlassian

TECHNICAL DEBT IN LEAN

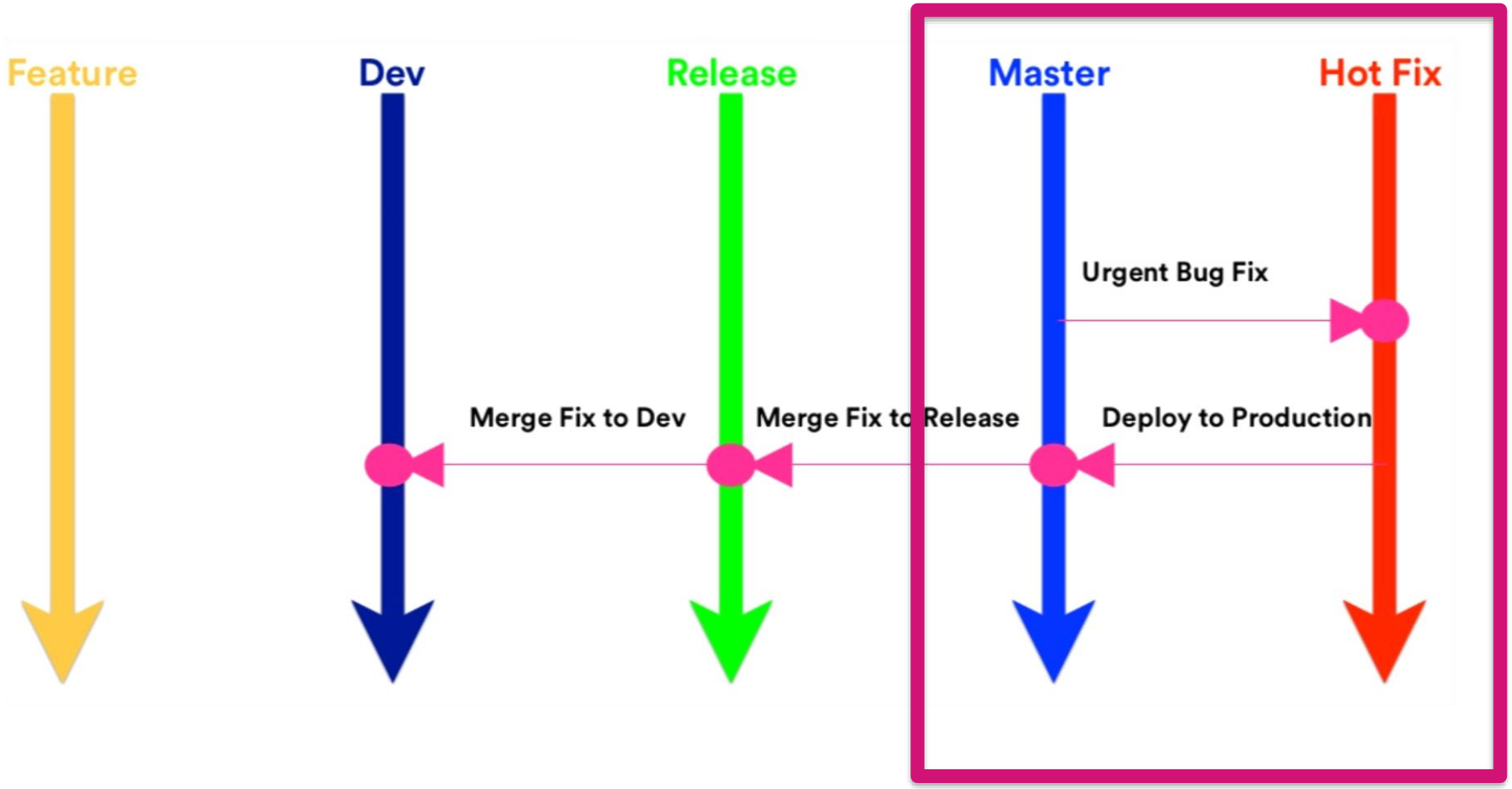
FEATURE BRANCHING STRATEGY



Source: Atlassian

TECHNICAL DEBT IN LEAN

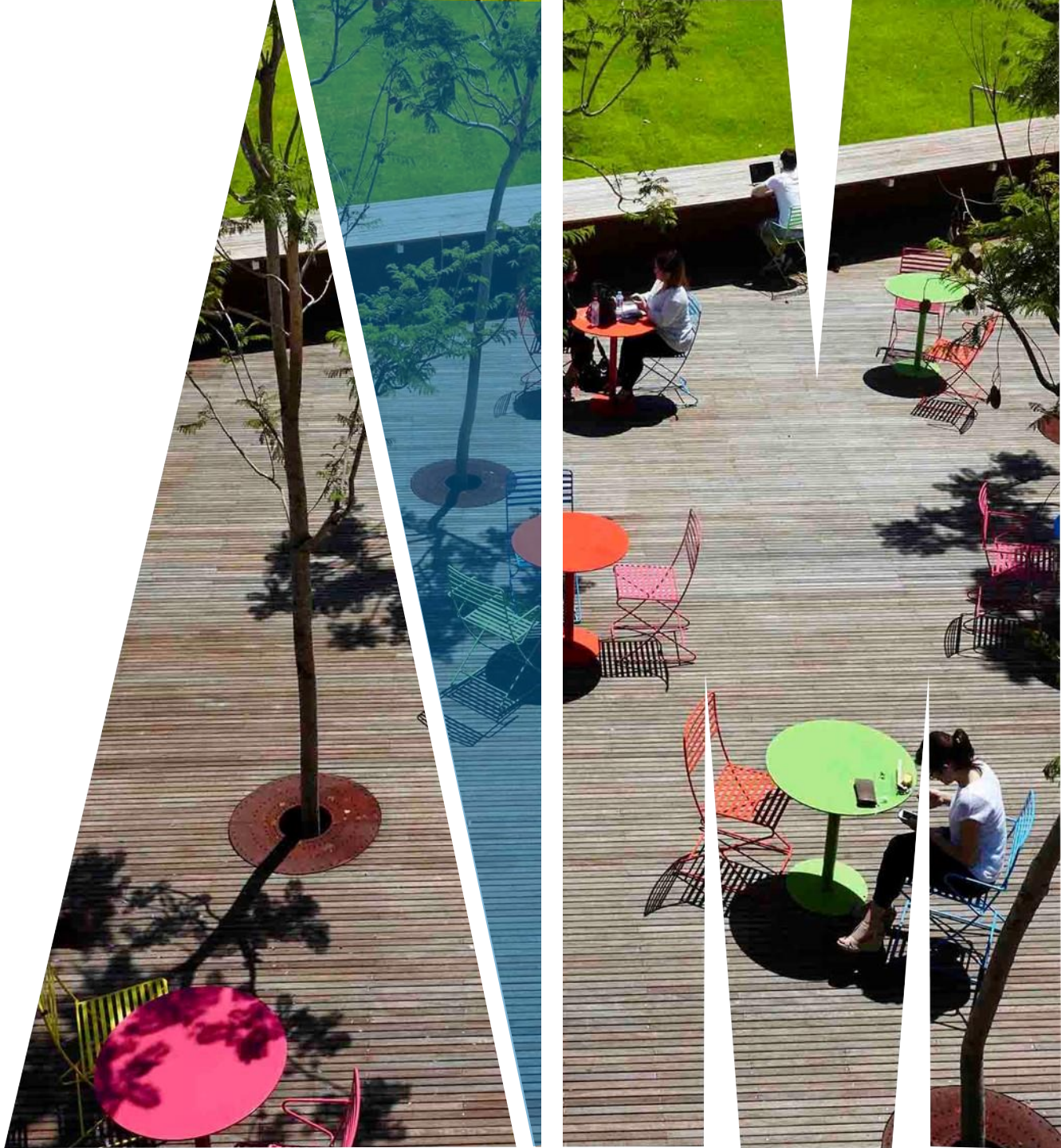
FEATURE BRANCHING STRATEGY



Source: Atlassian

MONASH University

Thanks



MONASH University