

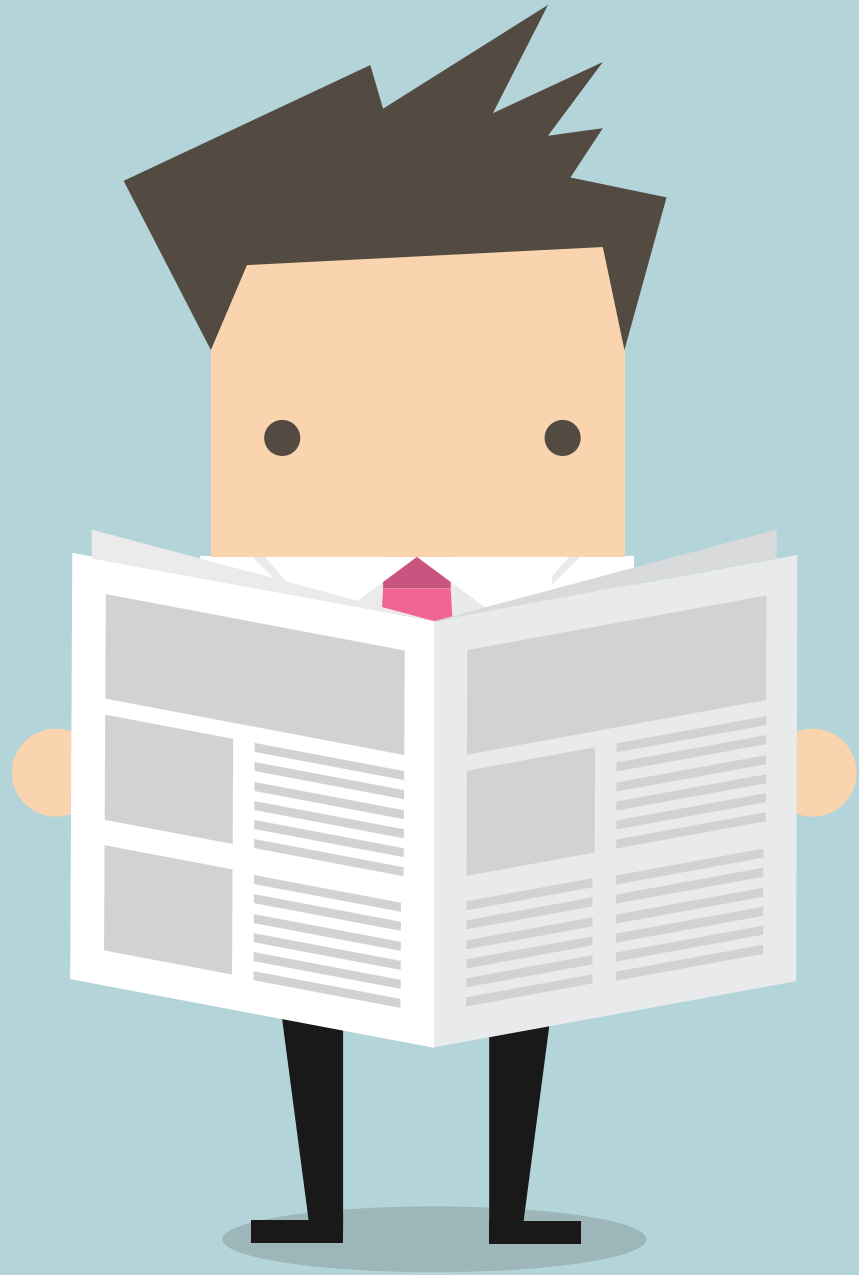
The (Software) Poverty Trap

David Kitchen

DrupalCon Vienna 2017

Disclaimers

- These are my own views and not those of Johnson & Johnson
- Choice of characters for each role is based on availability of artwork
- Apologies for any British cultural references



A Story About Poverty & Debt

Jim





Jim

Father
Agricultural
Laborer

Grandfather
Agricultural
Laborer

Great Grandfather
Agricultural
Laborer

Great
Grandmother

Grandmother

Great Grandfather
Agricultural
Laborer

Great
Grandmother

Mother
Cleaner

Miner

Miner

Domestic Servant



A poverty trap is "any self-reinforcing mechanism which causes poverty to persist." If it persists from generation to generation, the trap begins to reinforce itself if steps are not taken to break the cycle.



Poverty Traps

for Countries

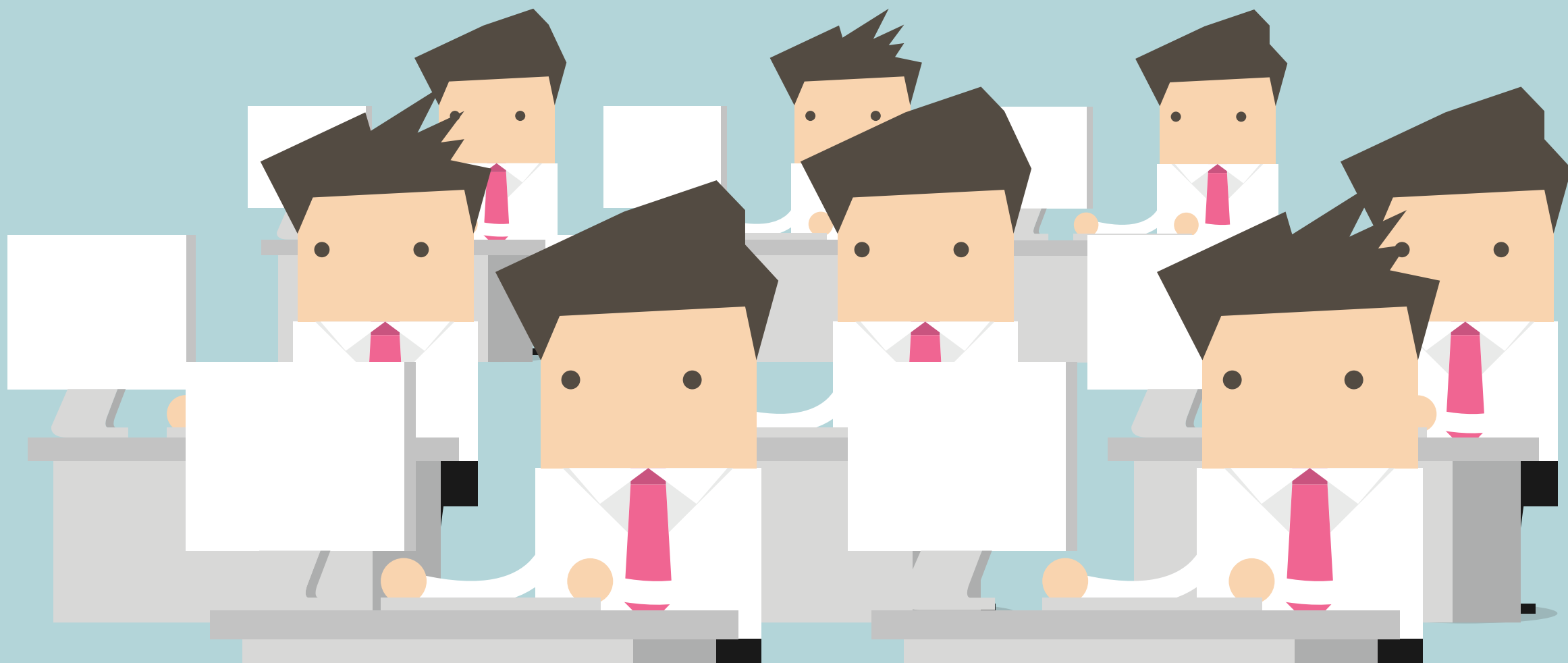
- Conflict
- Natural resources
- Landlocked Countries
- Bad Governance

for Companies

- Culture and Retention
- Buyer/Supplier Monopoly
- Bad Governance

The four poverty traps defined in *The Bottom Billion* by Paul Collier

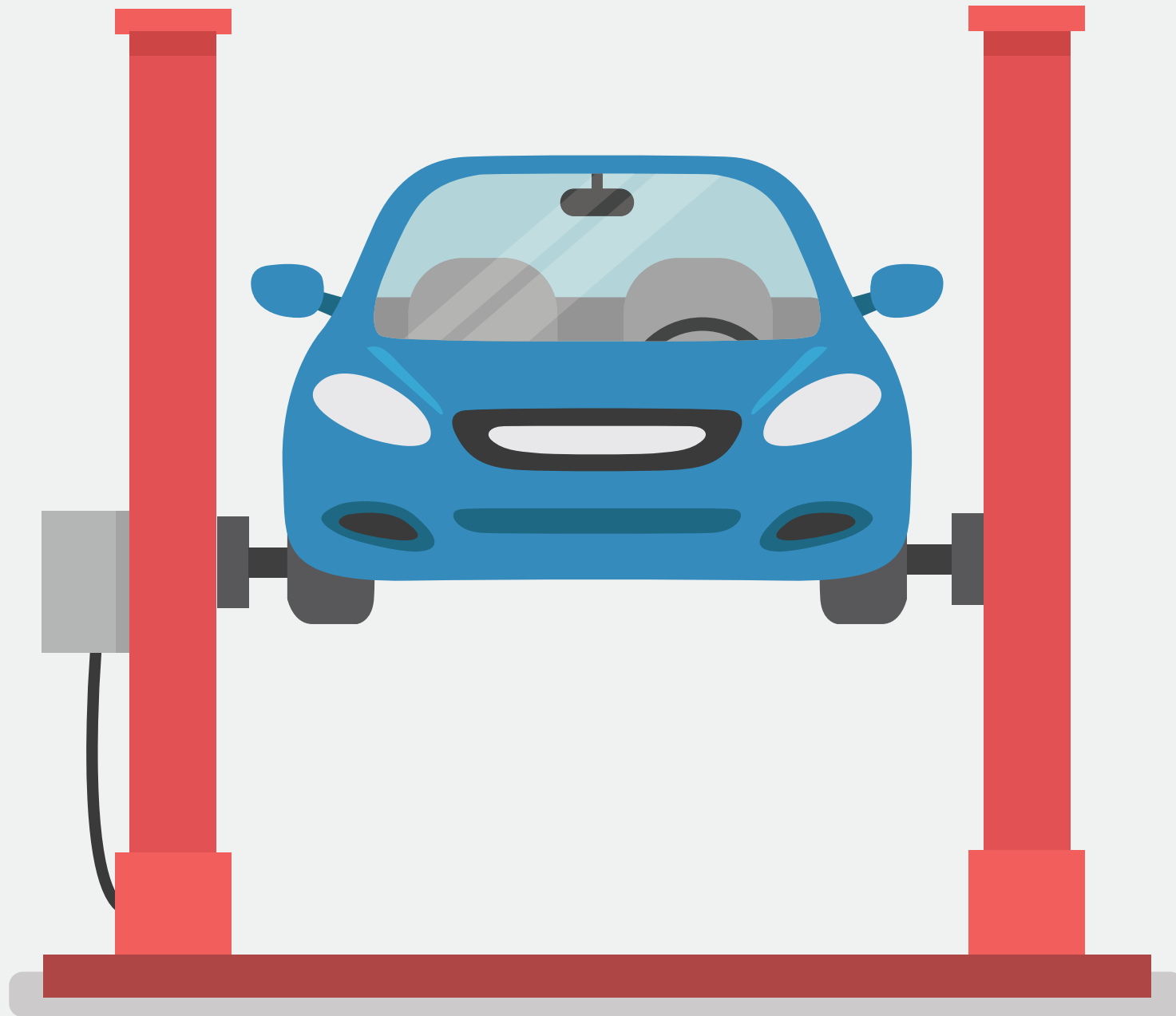












INVOICE

Placeholder text for invoice details, consisting of multiple lines of gray horizontal bars.



OMG!















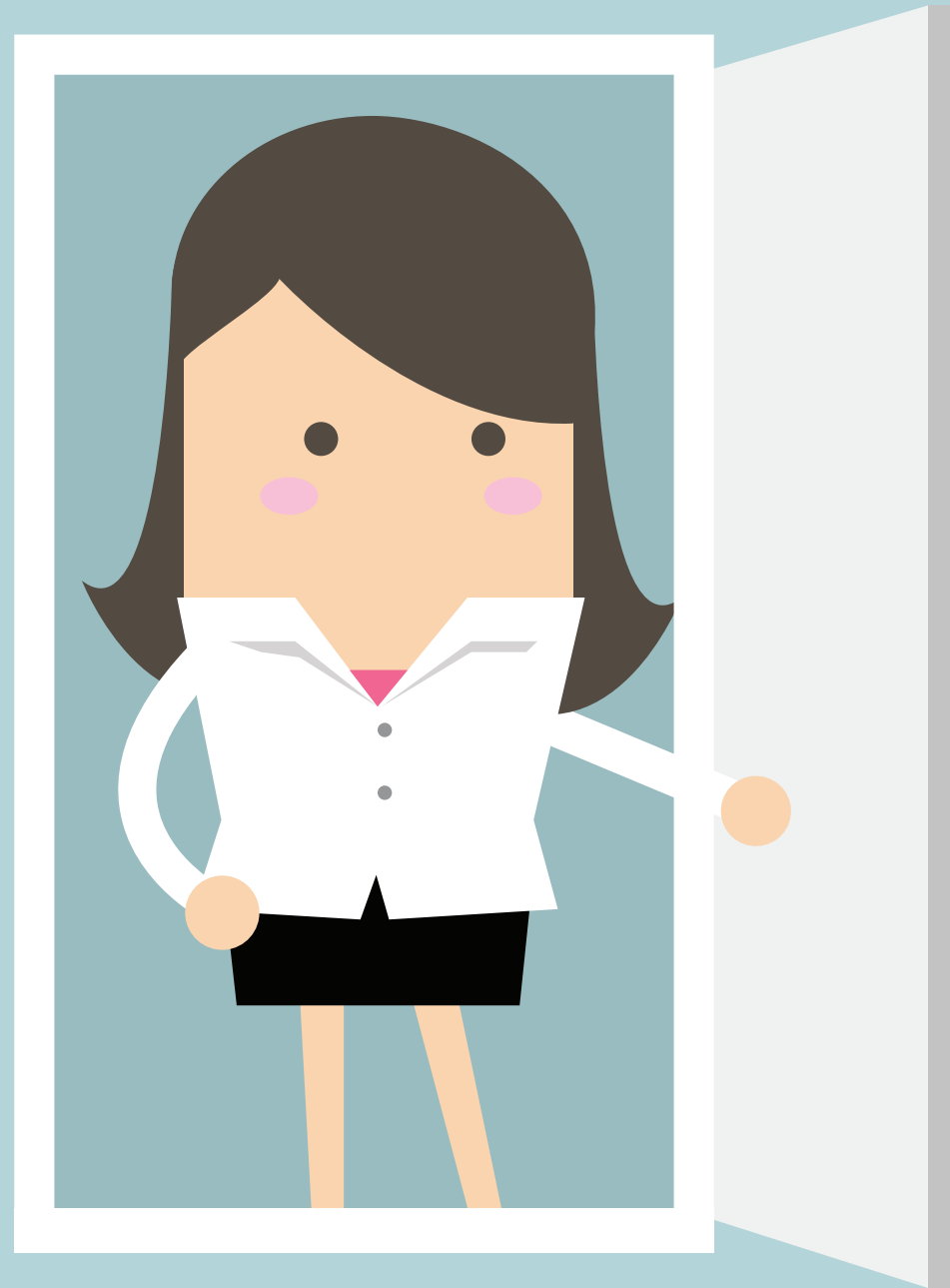


A debt spiral refers to a situation where an individual (or firm or country) sees ever-increasing levels of debt. This increasing levels of debt and debt interest becomes unsustainable, eventually leading to debt default.



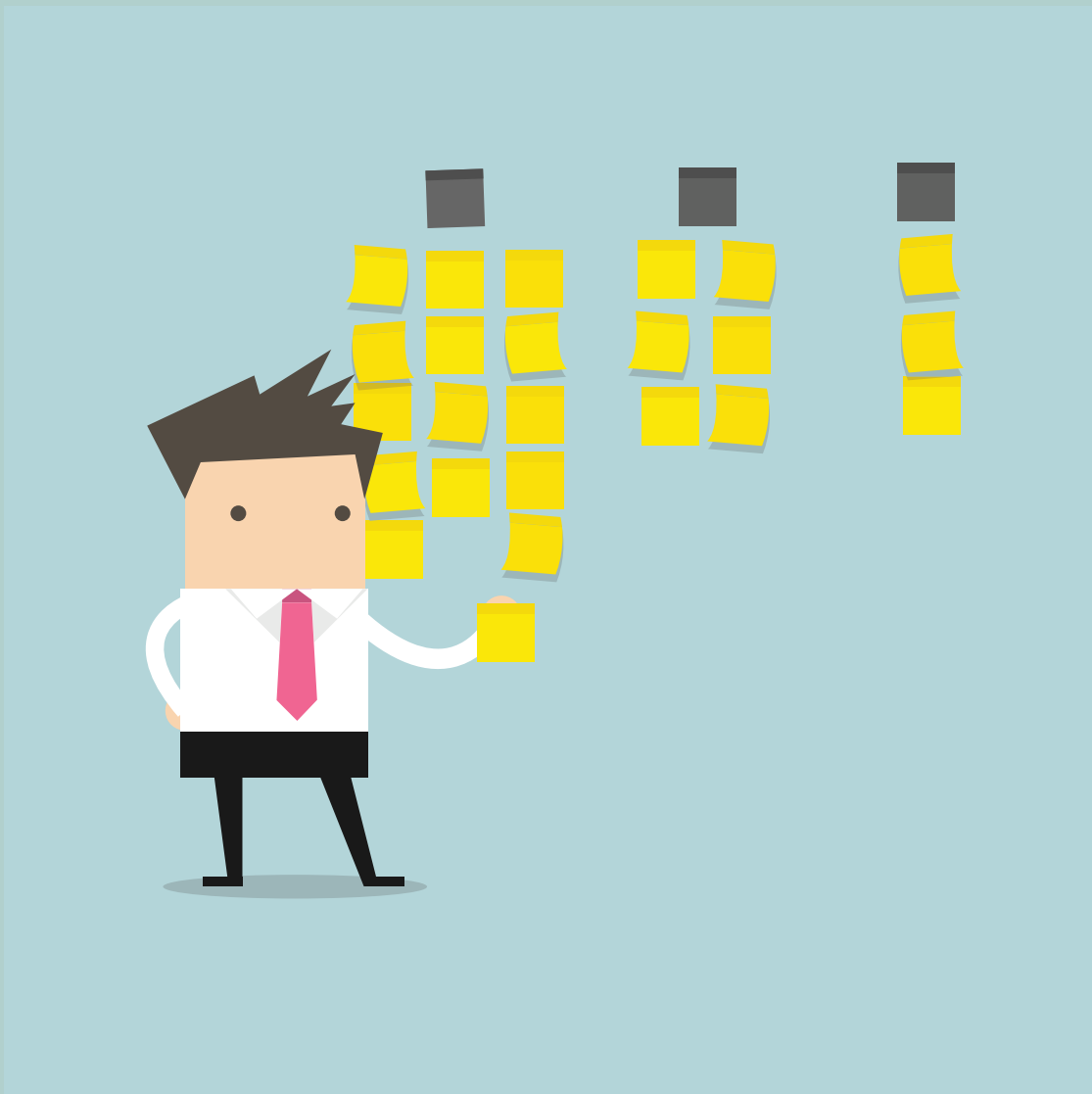
A Similar Story?

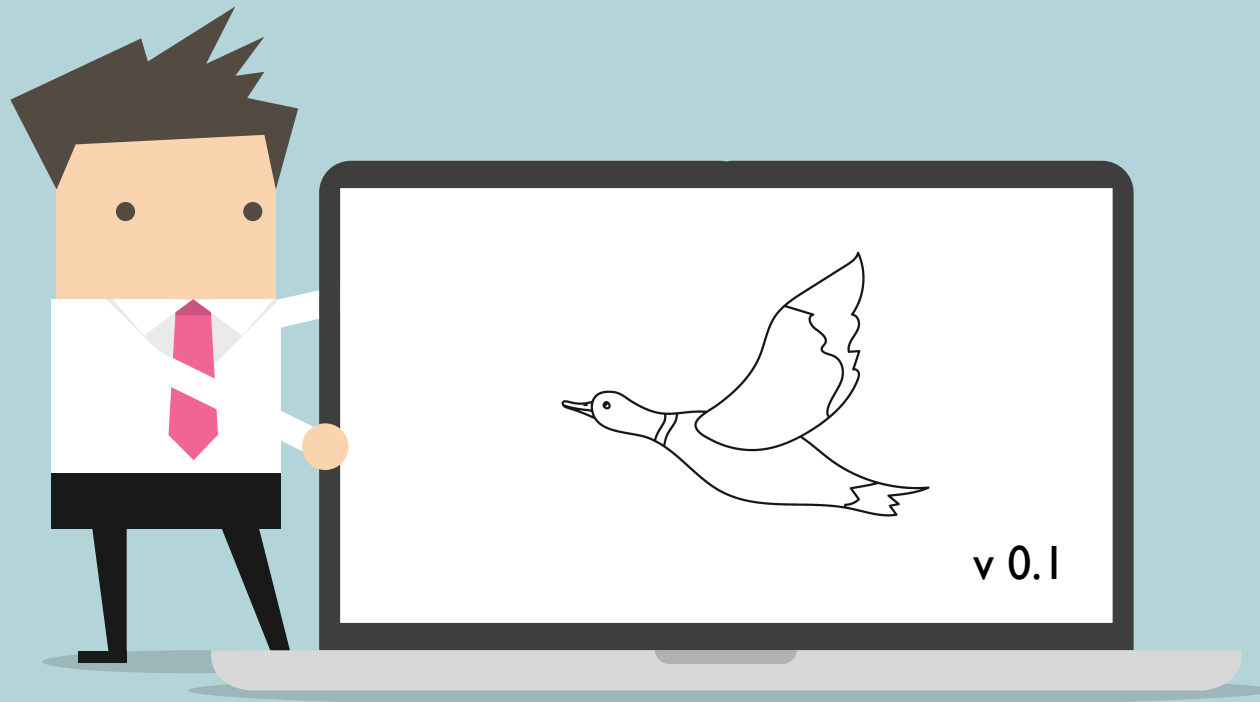
Rosie



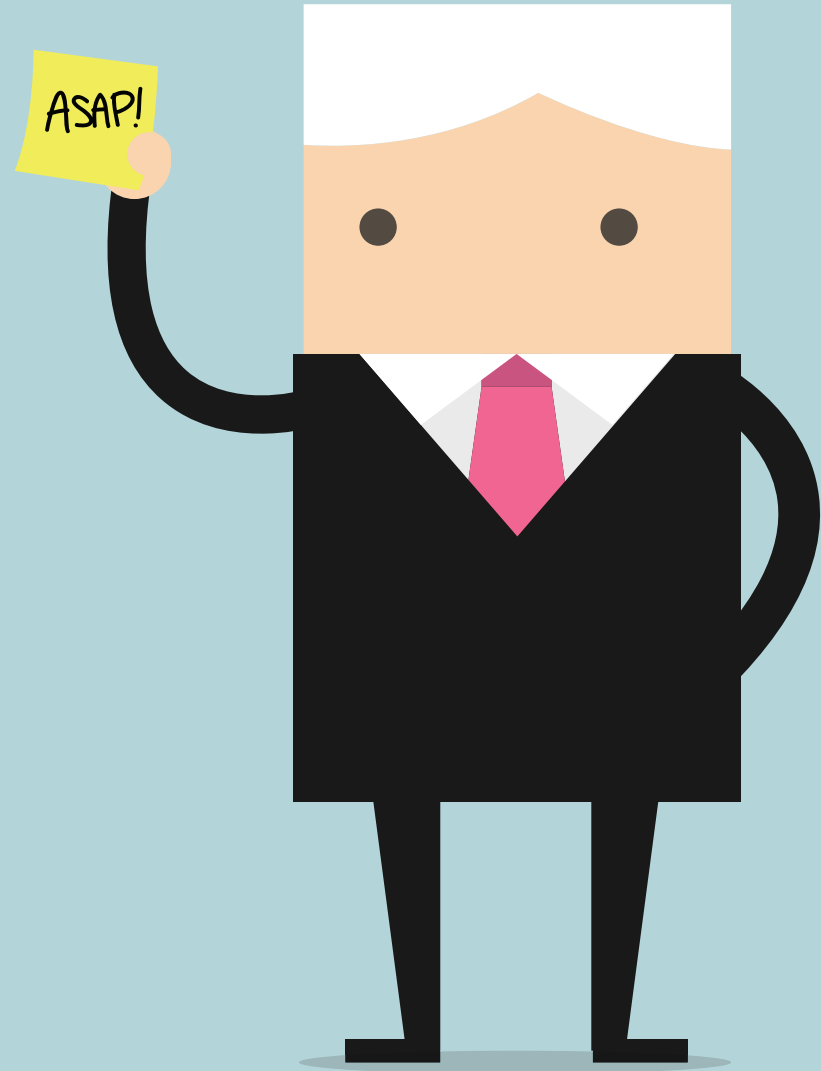






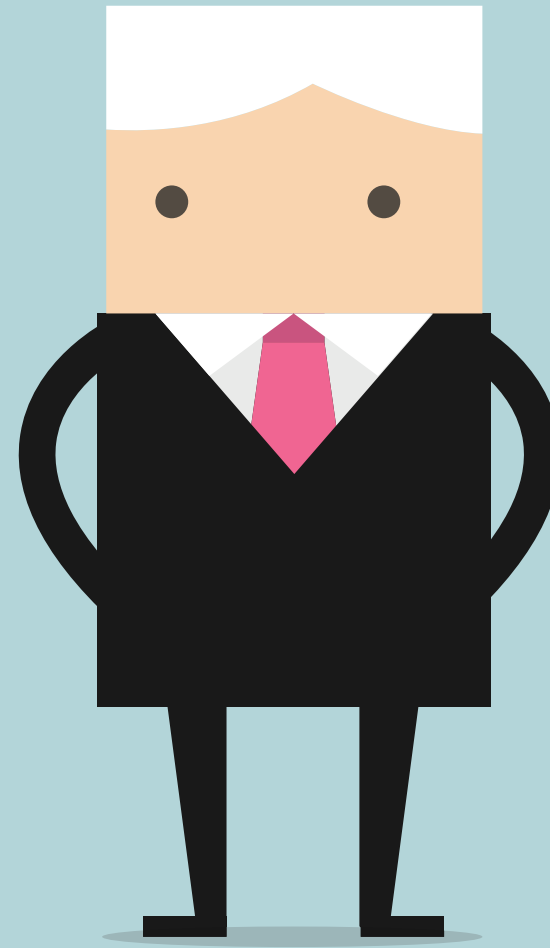


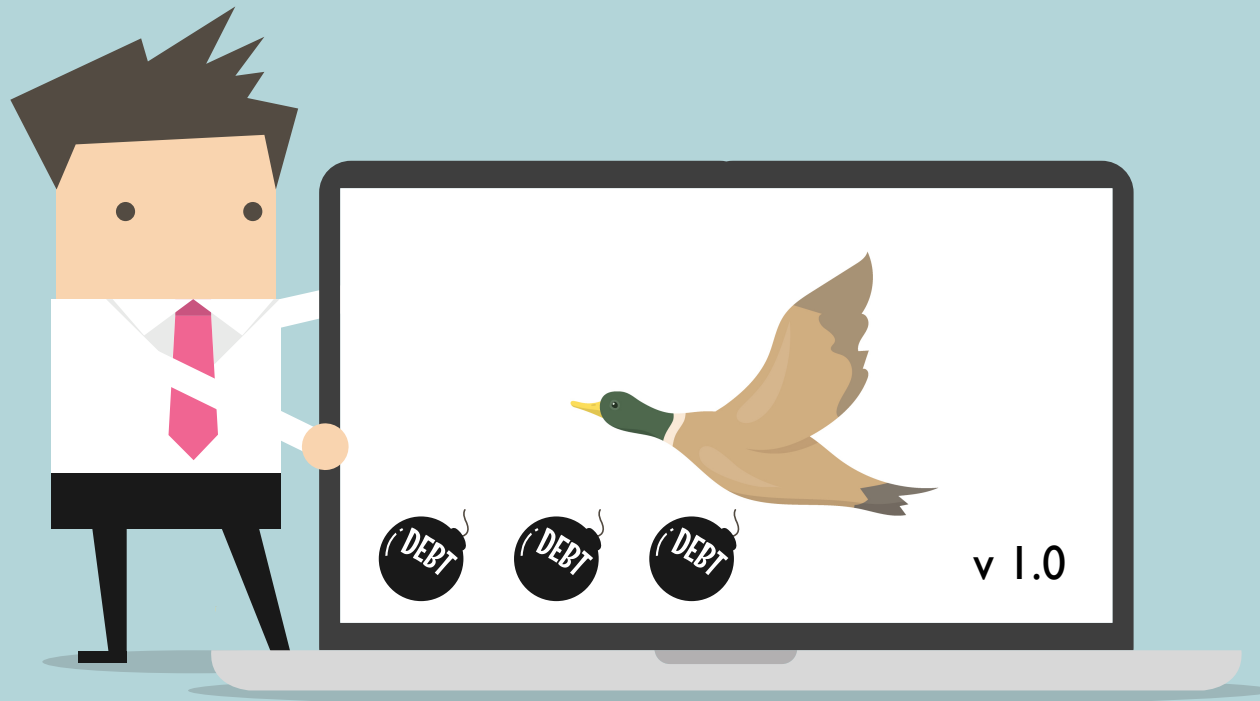












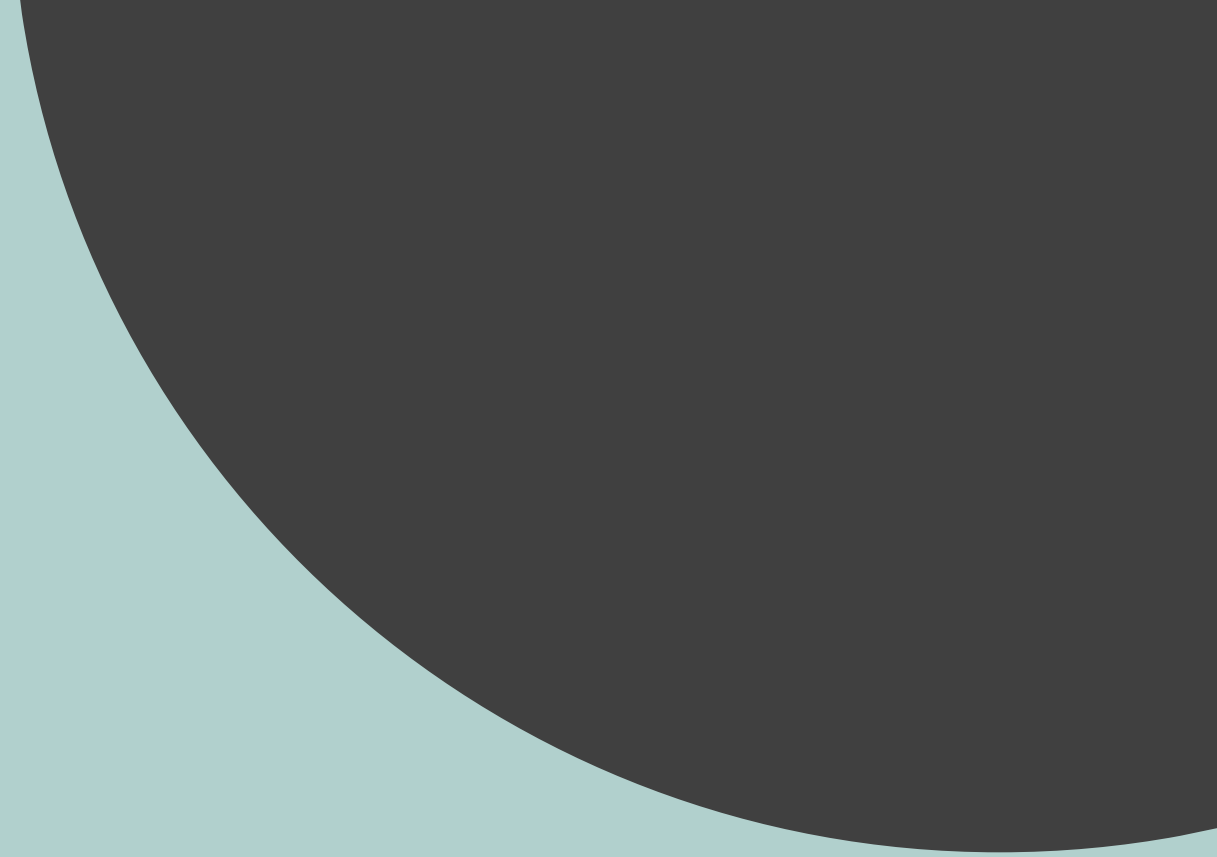
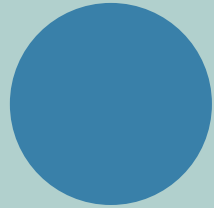
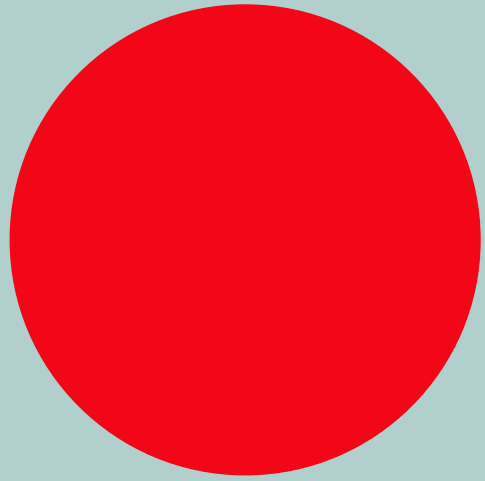
“After Scrum adoption, the most visible symptoms of dysfunction in our software development department were related to agile engineering practices, where teams were accumulating a huge amount of technical debt.”

Visualizing and Managing Technical Debt in Agile Development: an Experience Report

Paulo Sérgio Medeiros dos Santos; Federal University of Rio de Janeiro, System Engineering and Computer Science Department

Amanda Varella, Cristine Ribeiro Dantas, & Daniel Beltrão Borges; Petrobras, Exploitation and Production Business Solutions

Agile Processes in Software Engineering and Extreme Programming: 14th International Conference, XP 2013, Vienna, Austria, June 3-7, 2013, Proceedings



How do you feel about the
technical debt on your current
project?









Financial Thinking

In software-intensive systems, technical debt is a collection of design or implementation constructs that are expedient in the short term, but set up a technical context that can make future changes more costly or impossible. Technical debt presents an actual or contingent liability whose impact is limited to internal system qualities, primarily maintainability and evolvability.

The I6162 definition of Technical Debt

As defined at the Dagstuhl Managing Technical Debt in Software Engineering Seminar, April 2016

<http://www.dagstuhl.de/I6162>

What is Technical Debt?



Principle, the cost to repay

- **Initial Principle**
savings gained by taking some initial approach or shortcut in development
- **Current Principle**
the cost that it would now take to develop a different or better solution



Interest, the cost of not repaying

- **Recurring interest**
additional cost of incurred by the project because of the technical debt
- **Accruing interest**
additional cost developing new software depending on not-quite-right code





Types of Debt

Mortgage

Low rate of interest
Possibly a big principle

- Code formatting
- Code smells



Loan

Higher rate of interest

- Performance issues
- Documentation



Credit Card

Initial interest free period, followed by high interest

Acceptable interest for a short period

- Quick fixes
- Business opportunity





Measure and Track Your Debt

Burn Up Chart



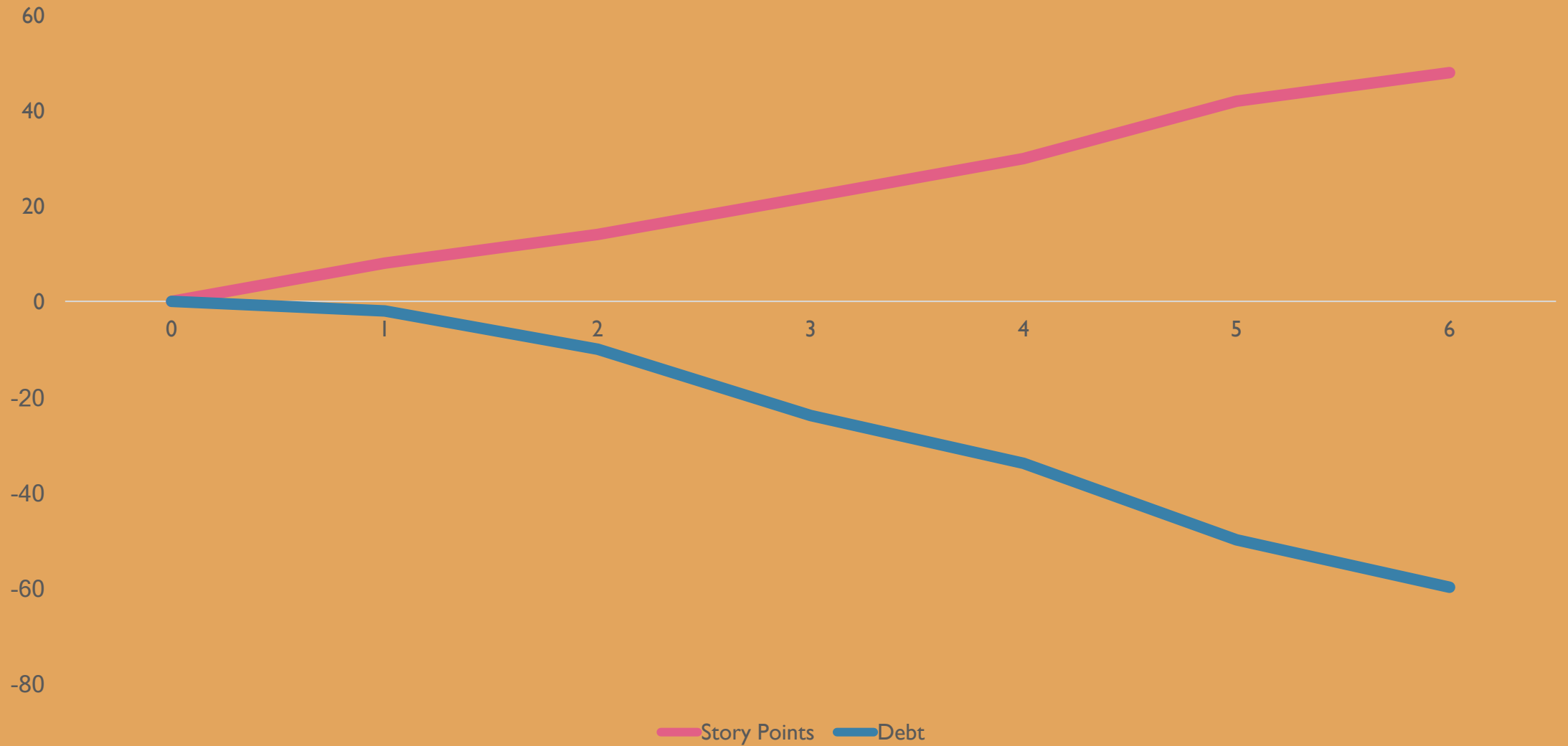
Business Balance Sheet

Assets		
Fixed Assets	10,000	
Current Assets	8,000	
Total Assets		18,000

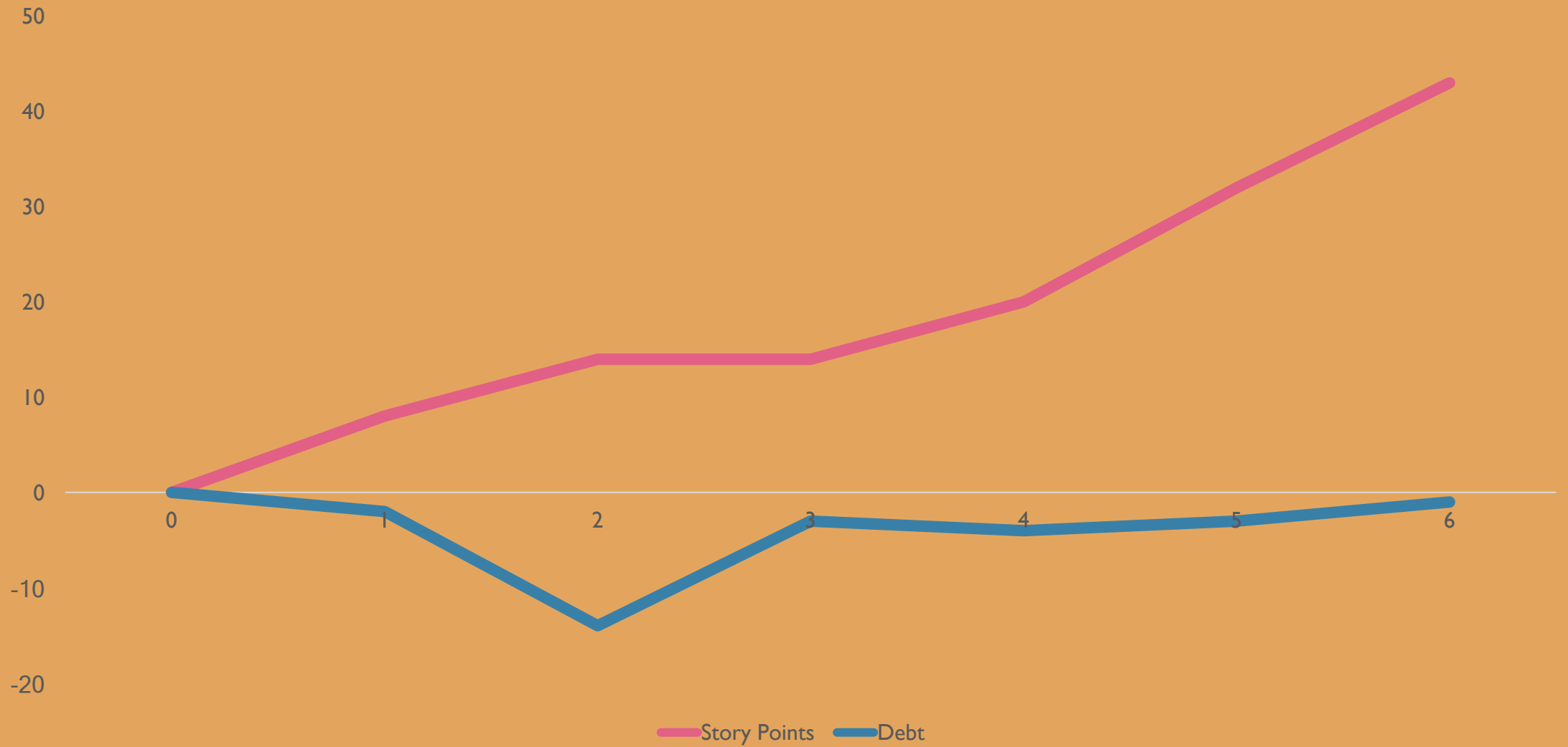
Business Balance Sheet

Assets		
Fixed Assets	10,000	
Current Assets	8,000	
Total Assets		18,000
Liabilities		
Creditors falling due within one year	4,000	
Creditors falling due after one year	15,000	
Total Liabilities		19,000
Net Assets		(1,000)

Velocity vs. Debt



Velocity vs. Debt





Good Financial Management





Quality Gate Passed

Bugs & Vulnerabilities

Leak Period: last 30 days
started 2 months ago

0 A
Bugs

0 A
Vulnerabilities

0
New Bugs

0
New Vulnerabilities

FindBugs is a program that uses static analysis to look for bugs in Java code. It can detect a variety of common coding mistakes, including thread synchronization problems, misuse of API methods.

2.4k

Lines of Code

Java 2.1k

XML 290

Quality Gate

(Default) [SonarQube way](#)

Quality Profiles

(Java) [Sonar way](#)

(XML) [Sonar way](#)

Home

Issues

Sources

Developer connection

Key

org.sonarsource.sonar-findbugs-plugin:sonar

Events

All

Version: 3.4.4

November 28, 2016

Quality Gate: Green (was Red)

November 28, 2016

Code Smells

2d A
Debt

started a year ago

73
Code Smells

0
New Debt

0
New Code Smells

Duplications

0.0%
Duplications

0
Duplicated Blocks

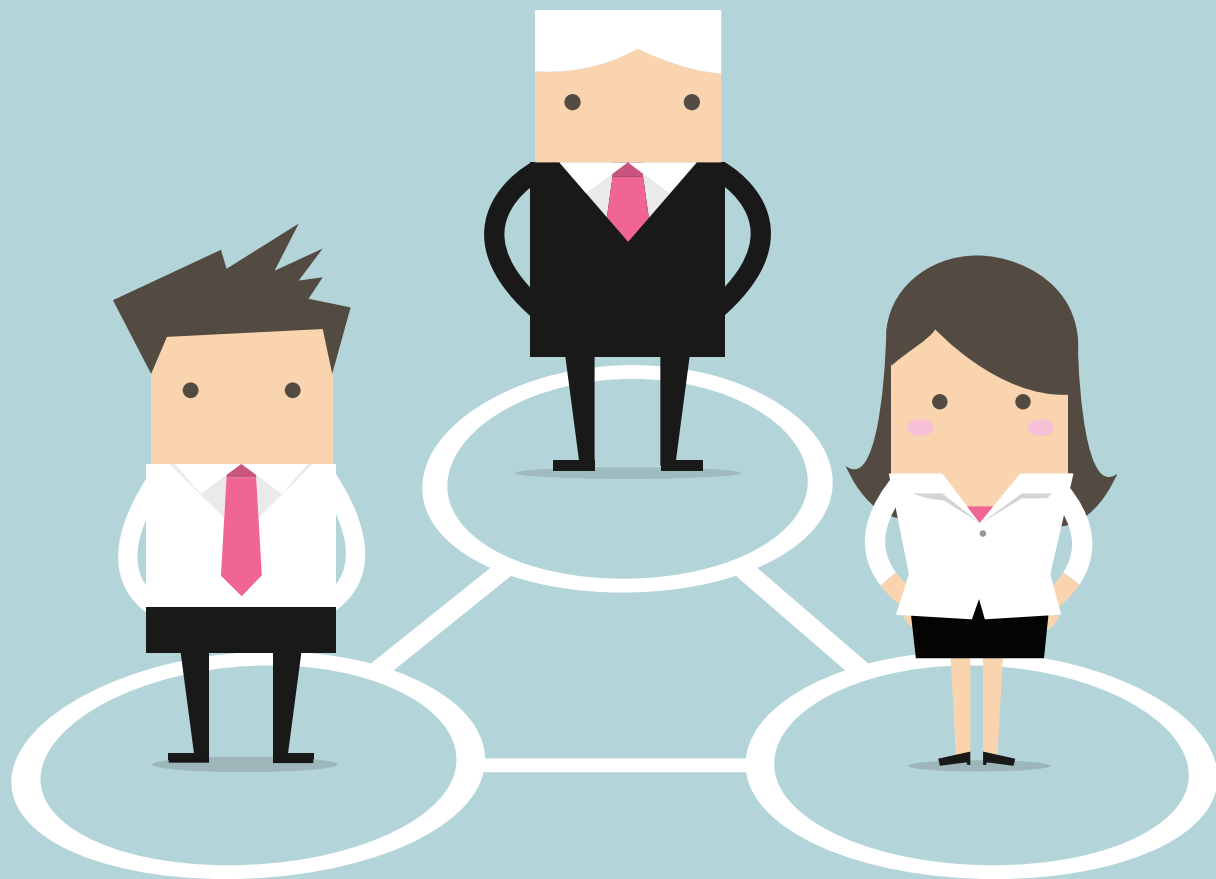
—
Duplications on New Code



**Harvard
Business
Review**

Your Agile Project Needs a Budget,
Not an Estimate by Debbie Madden
29 Dec 2014





Depreciation

- Debt incurs Interest
- Value Depreciates
 - Competition
 - Needs change



The Sunk Cost Fallacy

The Misconception

You make rational decisions based on the future value of objects, investments and experiences

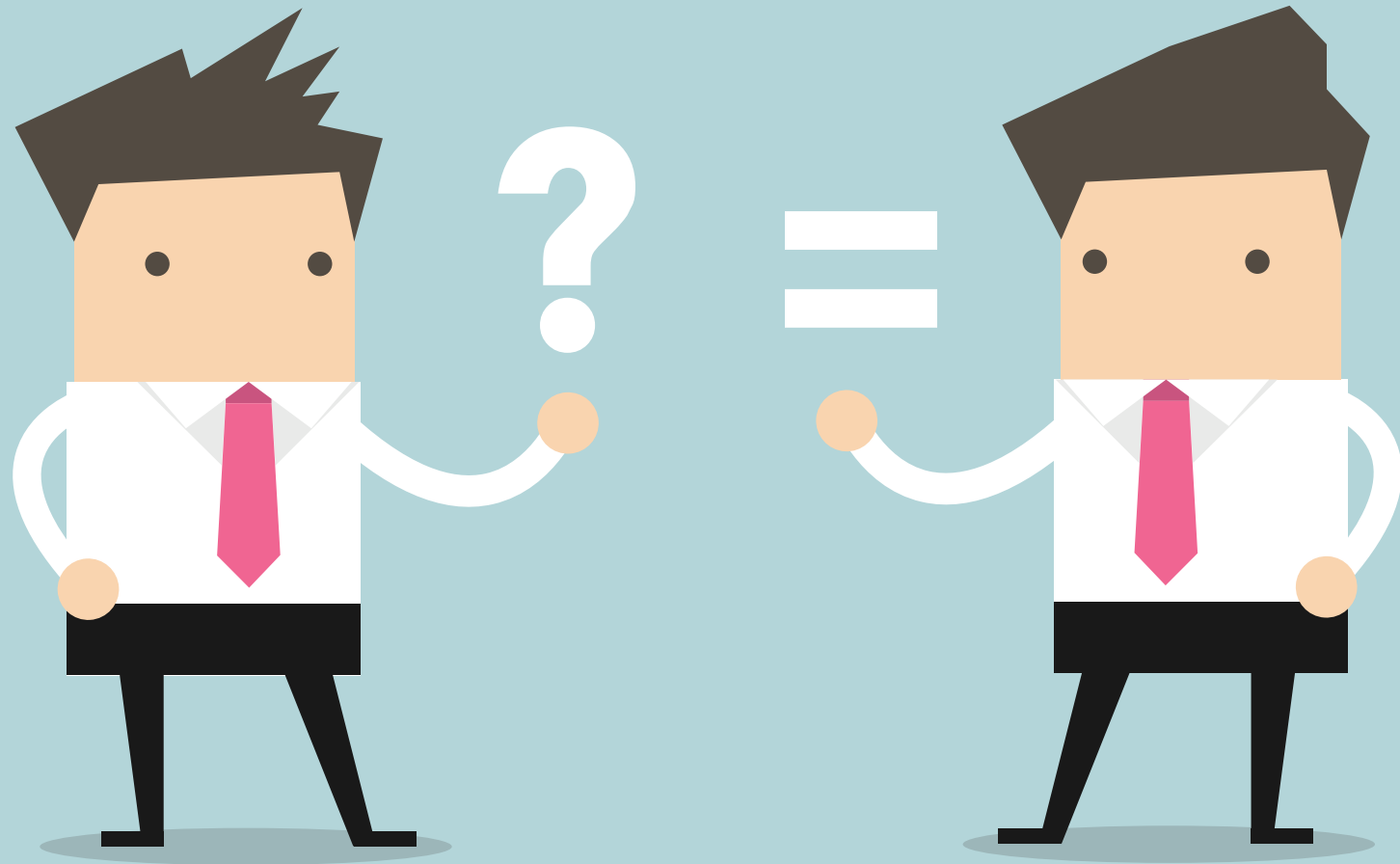
The Reality

Your decisions are tainted by the emotional investments you accumulated, and the more you invest in something the harder it becomes to abandon it.

Bankruptcy might be the best option







Further Information

- [TechDebt 2018](#): The inaugural *TechDebt* Conference will be held jointly with the International Conference on Software Engineering 2018 in Gothenburg, Sweden, May 27–28, 2018
- [Managing Technical Debt Workshop Series 2010 to 2017 Archive](#)
- [The Software Development Poverty Trap, Alexandros Marinos](#); original inspiration for this presentation



Please review the session:

<https://events.drupal.org/vienna2017/sessions/software-poverty-trap>

David Kitchen

Senior Software Engineering Manager

Johnson & Johnson

@dwkitchen

david@dwkitchen.com