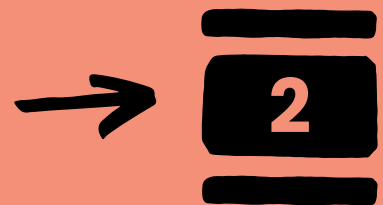


A COLLECTION OF CASE STUDIES

# The large-scale modernisation series

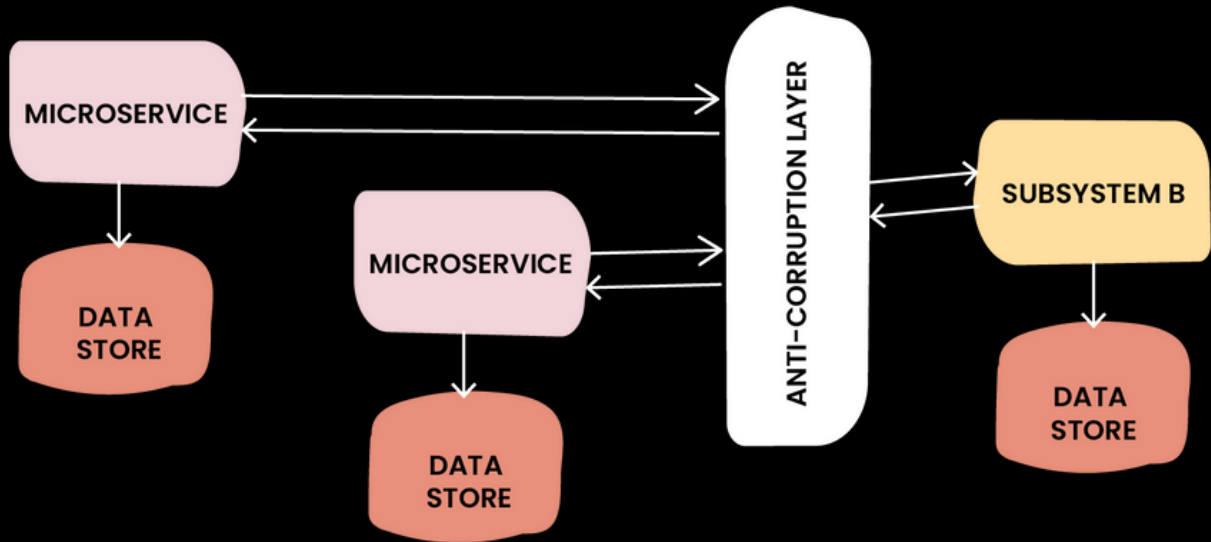
Wrapping the API's



# The problem

01

## Subsystem A



**Queensland Department of Transport and Main Roads (TMR)** was in need of modernising a large legacy system. The legacy system was on a dated mainframe with significant costs.

The strategy was to implement an anti-corruption layer (facade) around the legacy system and then gradually replace it with micro-services. There are 100's of endpoints that require implementation.

**A pilot showed that it would take 3 to 5 days to implement each endpoint.** With hundreds of these to implement, the time would simply blow out and costs would be too high.



STARTING WITH THE DOMAIN MODEL

# The solution

02

TMR commissioned a pilot to investigate using automation on the project and approached WorkingMouse.

The WorkingMouse approach is to apply the principles of Jidoka (automation with a human touch) by using bots to code the majority of what would have manually been done.



## Automation at scale

This is achieved by using a codebot. A codebot enables developers to fashion specific software engineering tools to achieve high-levels of automation, in other words, automate software development.



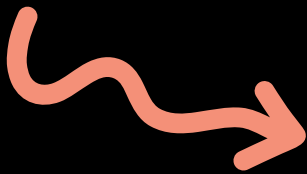
STARTING WITH THE DOMAIN MODEL

# The outcome

# 03

TMR's new custom built bot used the WSDL descriptions of the APIs to generate **over 90% of the code** that was being written manually.

The time taken to implement an endpoint was reduced from 3 to 5 days down to 1 to 2 days. Conservatively, it is an estimated **60% saving**.



**With the high-levels of automation, TMR is now able to implement 100's of endpoints and continue their large-scale modernisation.**



STARTING WITH THE DOMAIN MODEL