

## Inter-platform Interoperability for Zero Defects Platforms

By Marc Dorchain, Director Research Software AG

### Some questions for you

- How easy is it to get from platform A to platform B without losing historical data?
- Looking for integrated support to access multiple instances of IIoT platforms?
- Are Open standards preferred?

### Motivation

The ZDMP project is aimed at enabling interoperability in several ways. Not only is the interoperability to other external systems necessary, but also the possibility to connect to multiple instances of the ZDMP platform itself.

A standard setup of any platform / tool widely used in many domains, including manufacturing industry, usually consists of at least one development system, followed by a test system, and one or more systems in a production environment (Figure 1).

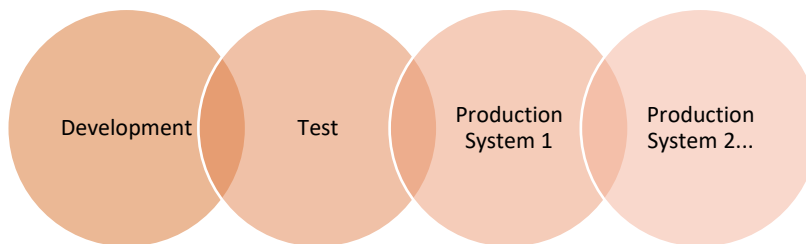


Figure 1:simple system life cycle

Of course, this is a very simple representation that is only meant to illustrate a distribution landscape where only one system is running in one place eg one system is responsible for one factory in a closed network without consideration of other connected systems / platforms.

### More than one enterprise, more than one network, and thus more than one gateway

Reality usually turns out to be more complex. Indeed, in the manufacturing industry usually there is more than one enterprise involved in a supply chain. This means, that structured data exchange is needed, that affects more than one enterprise, more than one network, and thus more than one gateway.

But how do you manage this challenge to expose your microservices and interfaces to the public? How do you do this in a secure way, following latest technologies and in a controlled and supervised way?

### What will ZDMP achieve

In ZDMP, a dedicated API management has been introduced to solve this problem. API Management (see Figure 2) means, that you have one single place where all APIs of the platform are exposed. Only

those partners that possess corresponding permissions are able to access an API, and thus the only access point is the API Gateway. The Gateway has two main functions. The first one is that the Gateway is acting as a kind of firewall blocking all unwanted access from outside and giving developers or admins one single place where all APIs can be found and managed. secondly, it allows usage monitoring, which is important from a business perspective of the platform owner, so that they can see how often an API is going to be called from a supplier or a customer outside of his organisation.

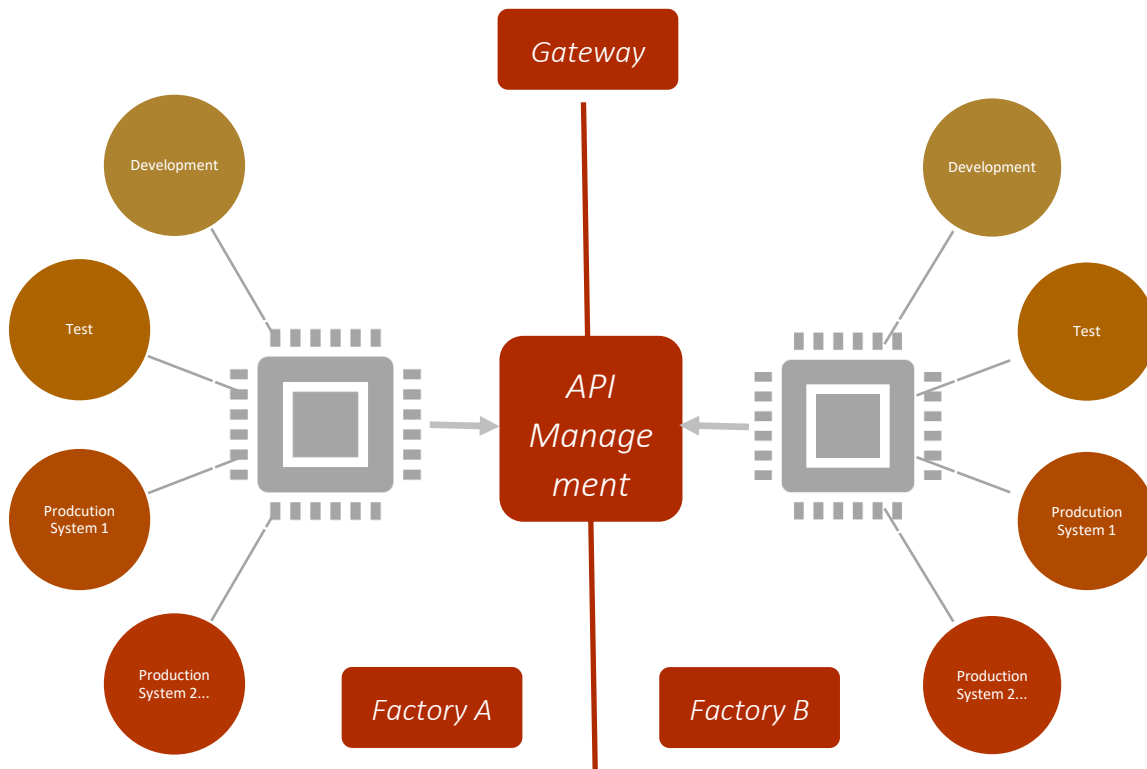


Figure 2: API Management and Gateway

The biggest hurdle for most businesses is how to blend microservices with existing systems in the organisation. Most companies are not going to simply throw out the old systems and introduce the new ones – this just does not make sense. So, can both existing and newly introduced systems, co-exist alongside each other?

Yes, but there is a steep learning curve. A microservices-based integration architecture requires new patterns for event-based communication, micro-gateways for policy enforcement, and network-level controls to be managed with a service mesh. And, as a side effect, complexity of micro-services is going to be reduced.

With an API Management platform for centralised operation of all your APIs, microservices, and meshes, Applications can be successfully deployed and managed with confidence. Context-specific rules can be created for customers that are personalised and targeted – without coding. This helps to achieve the agility that comes as a benefit of microservices without the complexity that can accompany microservices architectures.

### ZDMP Links

• Component	None (Uses other Components)
• Work Package	WP2: Business Challenge: Vision, Market, Use Cases, and Interlinking WP6: ZDMP Platform Building

• Tasks	T2.6: Cross Platform Analysis and Inter-Linking T6.5 Inter-platform Interoperability
---------	---

## References/Acknowledgements

[1]. Platform technologies of the ZDMP project - <https://www.zdmp.eu/technologies>