# Service oriented architecture for A1 production

The vision for the new production environment is to adopt a Service Oriented Architecture (SOA) and specifically the Micro-Service based one.

This approach will increase flexibility in compositionality, orchestration of core technical and business activities. Moreover it will permit workflow design and management combining the business and technical activities in transparent and easy to monitor manner.

The compositionality and orchestration principle is reflected in Figure 1 and Figure 2. The top service plays the function of orchestrator. Typically this role is plaid by a BPMN process executed in a workflow engine. It also provides an intimate representation of the business processes in the organisation. Activities composing the business are either manually performed by a user or automatic execution of procedures in/by available information systems. Each process activity aims at fulfilling a role or provide a functionality which in turn in supported by a micro-service running outside of the workflow engine. This way the micro-services are consumed on demand by business-process activities.

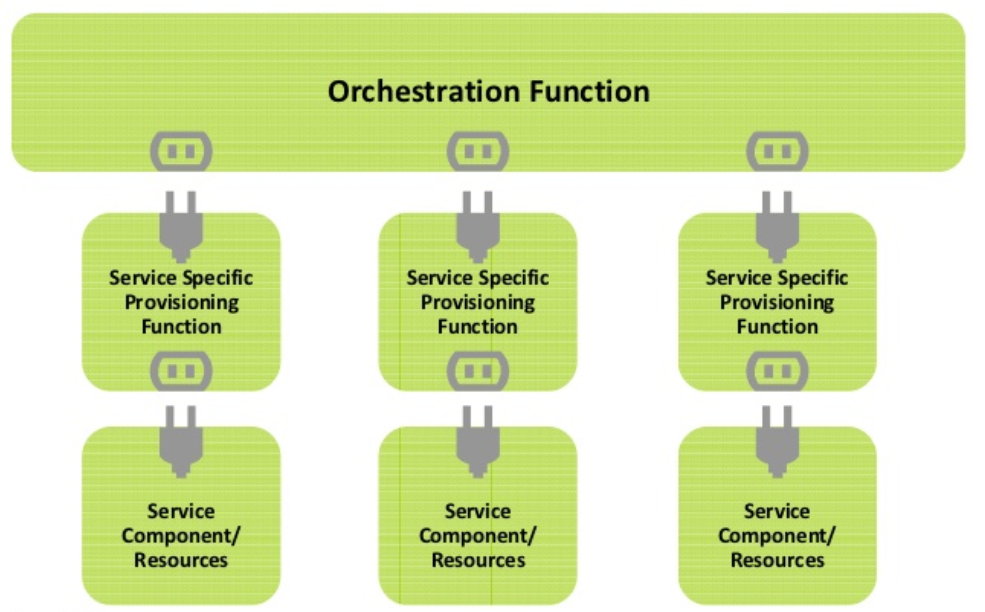


Figure 1 Compositionality and orchestration of services

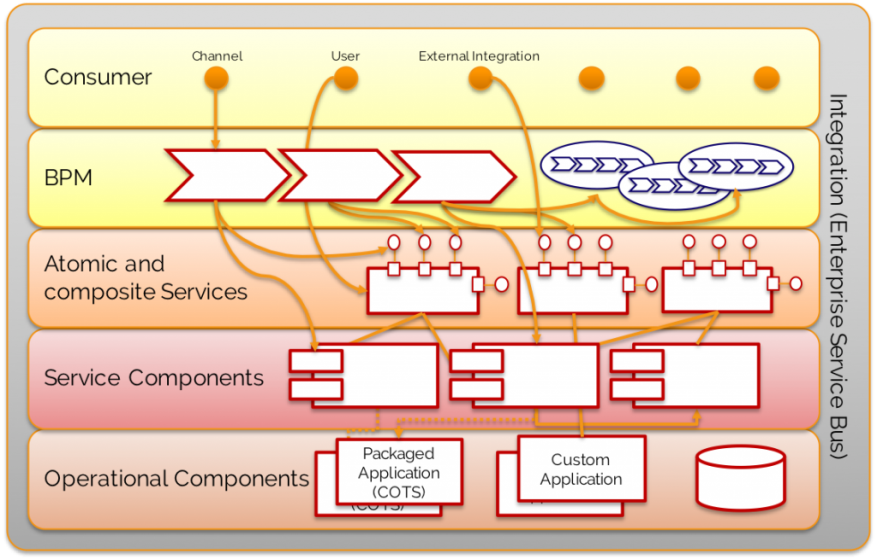
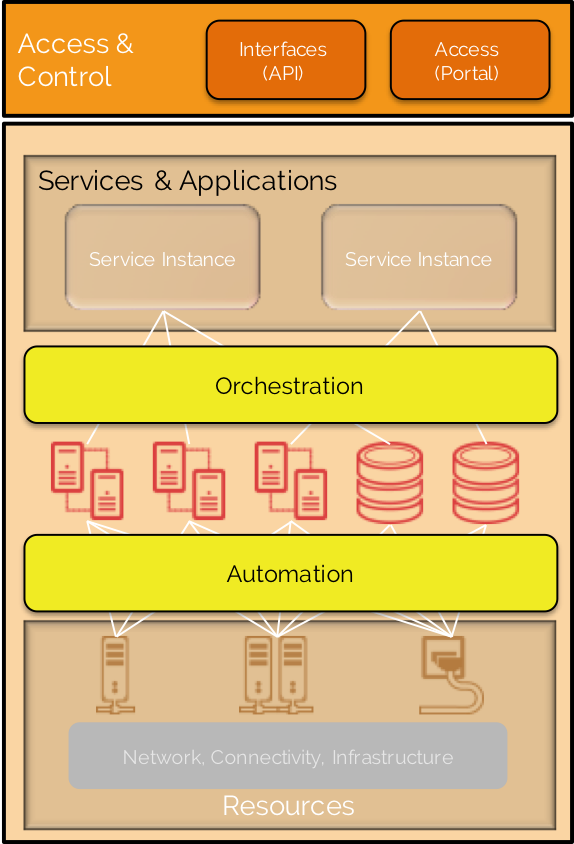


Figure 2 Layers of business process abstraction



# DEV server exercise

The development server is the environment to experiment with containerizing, launching, inter-connecting and automating deployment process for various components needed in the A1 sector.

Containerized services and applications is one of the de facto approaches for deploying and maintaining micro-service conglomerates. Docker is the state of the art system that enables containerization applications. It becomes especially powerful when combined with Docker Compose, Kubernetes or OpenShift. We assume here that the reader is familiar with benefits of containerization[[1]](#footnote-1), container-orchestration systems[[2]](#footnote-2) and automatic application deployment[[3]](#footnote-3).

Ideally each service needs to run in a separate Docker container.

…

TBD

# References

* <https://martinfowler.com/articles/microservices.html>
* <https://www.slideshare.net/alanmcsweeney/orchestration-and-provisioning-architecture-for-effective-service-management>
* <https://zeebe.io/blog/2018/08/bpmn-for-microservices-orchestration-a-primer-part-1/>
* <http://smileit.at/blog/system-versus-service-orchestration/>
* <https://sparxsystems.com/press/articles/pdf/EAforADP.pdf>

1. <https://dzone.com/articles/top-10-benefits-of-using-docker> [↑](#footnote-ref-1)
2. <https://www.redhat.com/en/topics/containers/what-is-container-orchestration> [↑](#footnote-ref-2)
3. <https://www.red-gate.com/blog/software-development/5-big-benefits-automated-deployment> [↑](#footnote-ref-3)