

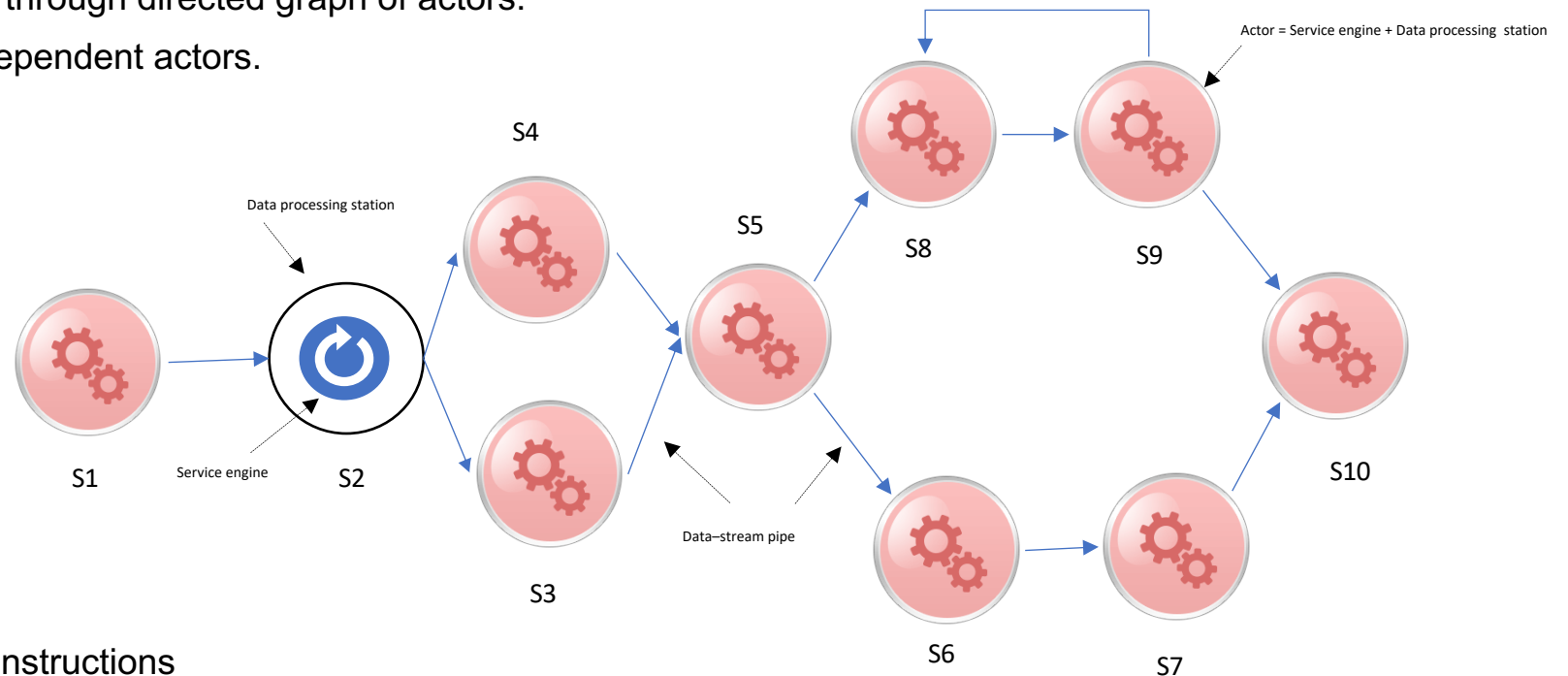
ERSAP

In Nutshell

By Vardan Gyurjyan

In a few words

- Reactive event driven **actors** (micro-service), communicating data through **data-stream pipes**, controlled by the application **orchestrator**.
- Stream of data quanta, flowing through directed graph of actors.
- Application is a network of independent actors.

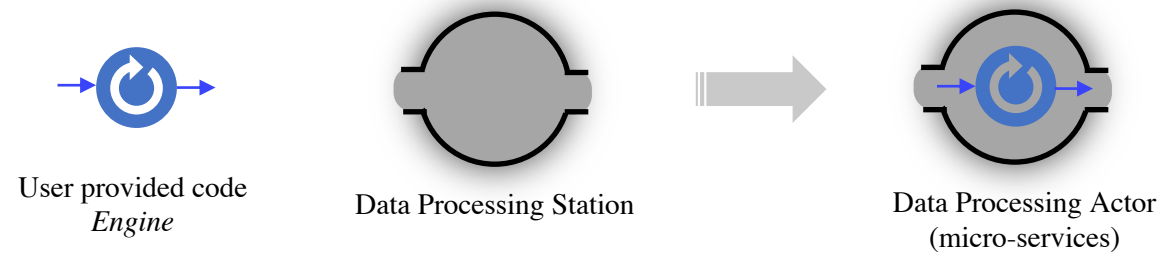
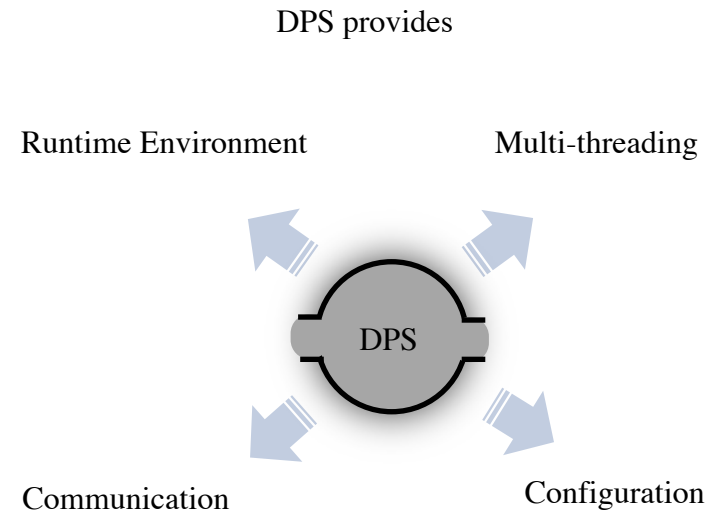


- Data moves across actors not instructions
- Actors communicate with each other by exchanging the data quanta across predefined connections by message passing, where connections are specified externally to actors.
- User provided data processing single-threaded algorithms (engines) are presented as fully scalable actors in the framework.

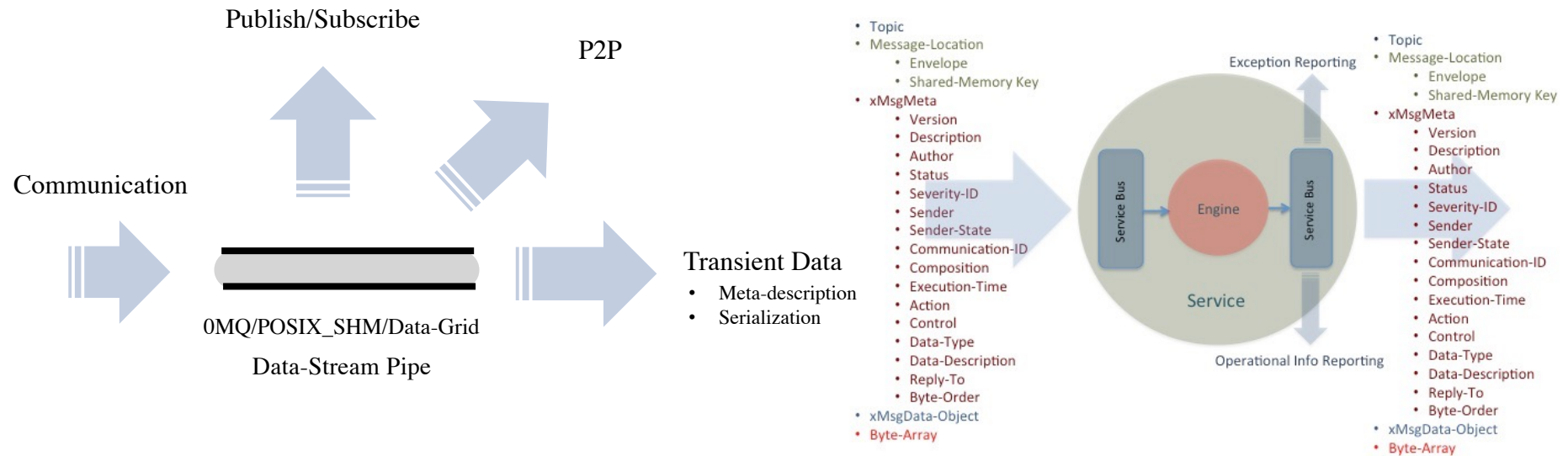


Actor = User engine + Data processing station (DPS)

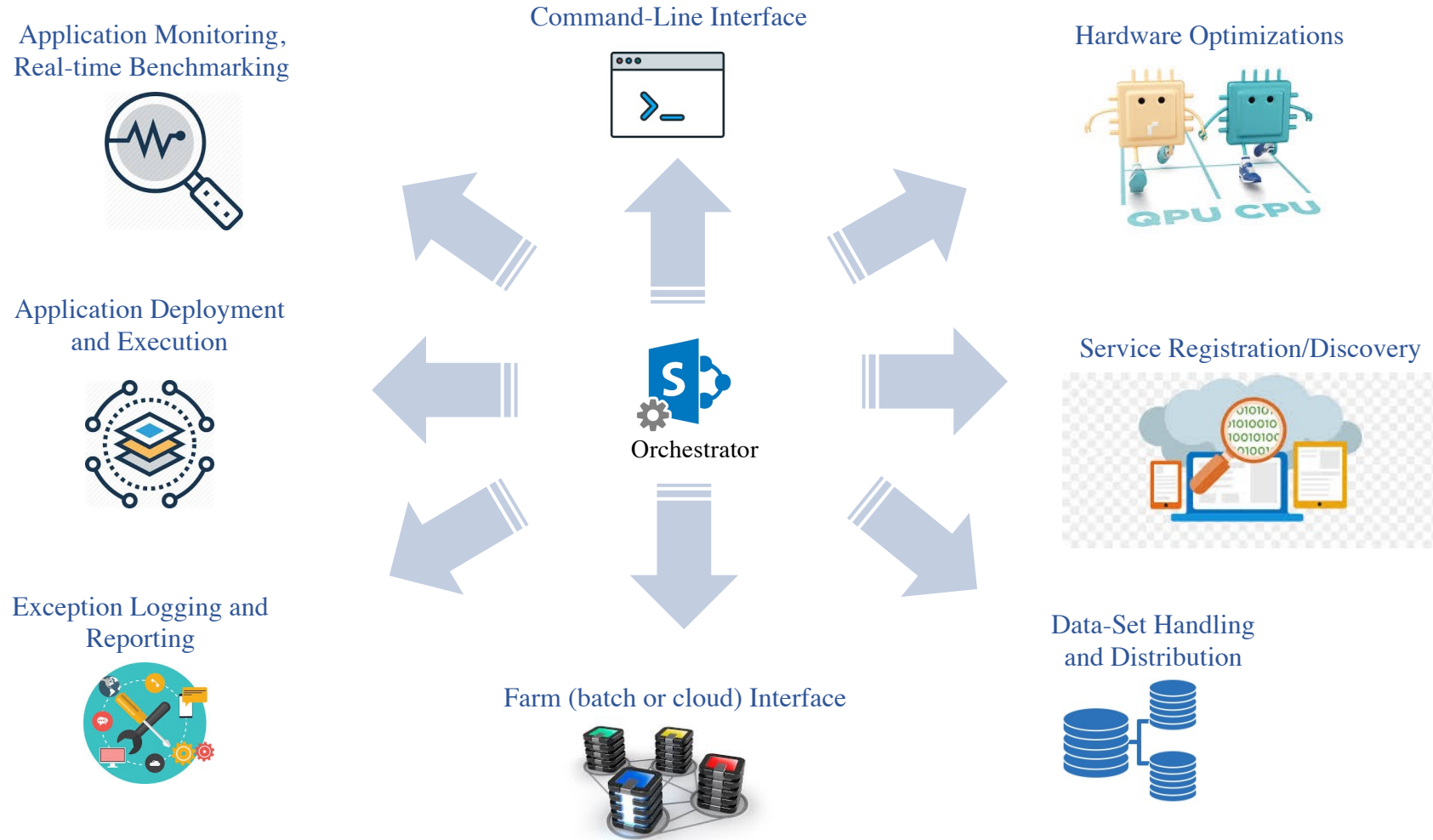
- Engine follows data-in/data-out interface.
- Engine gets JSON object for run-time configuration.



Data Stream Pipe



Application Orchestrator



Advantages of the actor model

- Artifacts are small, simple and independent
 - Easier to understand and develop
 - Reduced develop-deploy-debug cycle
- Easy to migrate to data
- Scales independently
- Independent optimizations
- Improves fault isolation
- Easy to embrace hardware heterogeneity
- Eliminates long term commitment to a single technology stack.

INDRA-ASTRA data pipeline ERSAP prototype

