

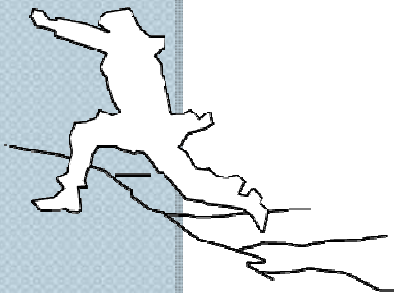


# UrbanFlood

## CIS – a Software Framework for Early Warning Systems

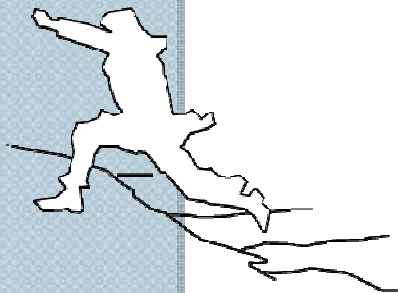


Bartosz Baliś, Marek Kasztelnik, Tomasz Bartyński,  
Grzegorz Dyk, Tomasz Gubała, Piotr Nowakowski,  
Marian Bubak

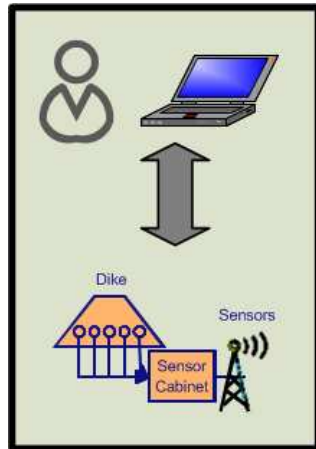


# Plan

- Motivation
- The approach: Service-Oriented Framework for building complex systems
- The Common Information Space
- CIS-powered Flood Early Warning System

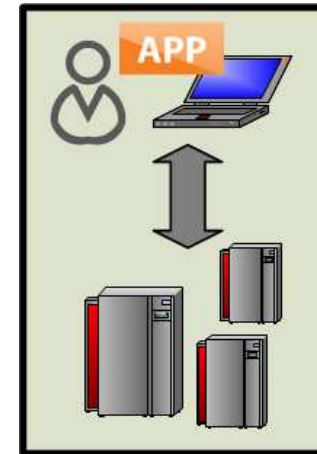


# Motivation

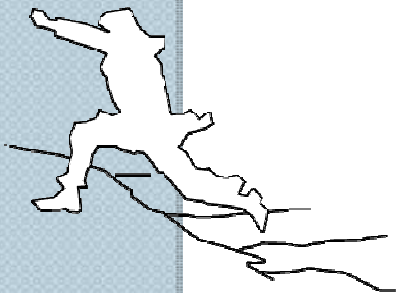
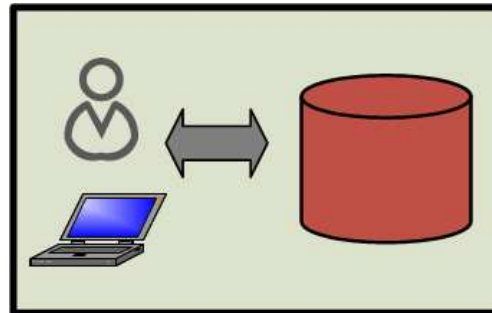


Dike +  
Sensor data

Flood simulation +  
Local HPC resources

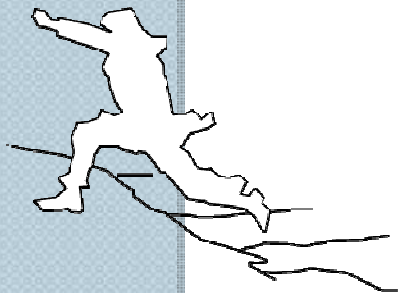


Data sets  
(e.g. terrain topography)



# Motivation

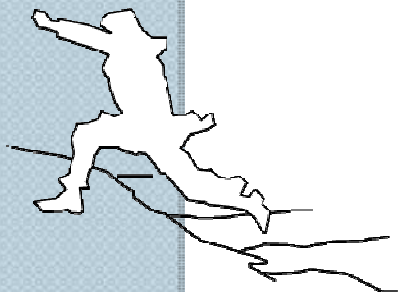
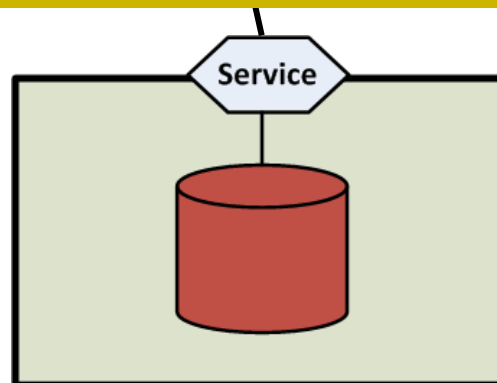
- Bring together many resources needed in a complex system
- Early Warning System = sensors + ICT infrastructure + domain models
  - E.g. Flood EWS = dike sensors + hardware & software + dike stability / AI / inundation / ... models



# Service-Oriented Approach

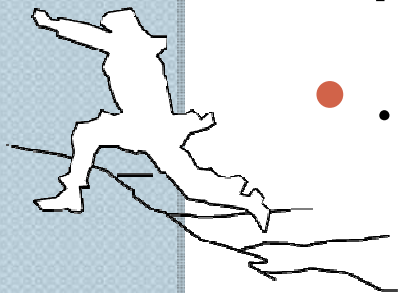


**Exposing resources as services:  
first step towards complex systems**



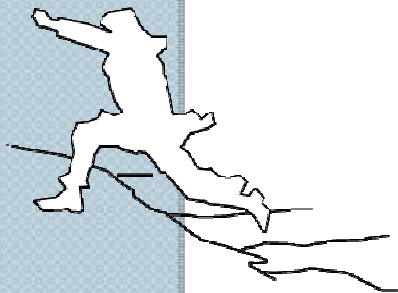
# Towards complex systems: common problems

- Foundations for software services
- Communication between services
- Coordination of execution (workflow)
- Monitoring & management of services
- Allocation of resources to services
- Fault tolerance
- Provenance tracking
- ....

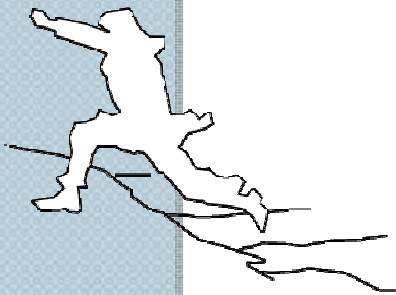
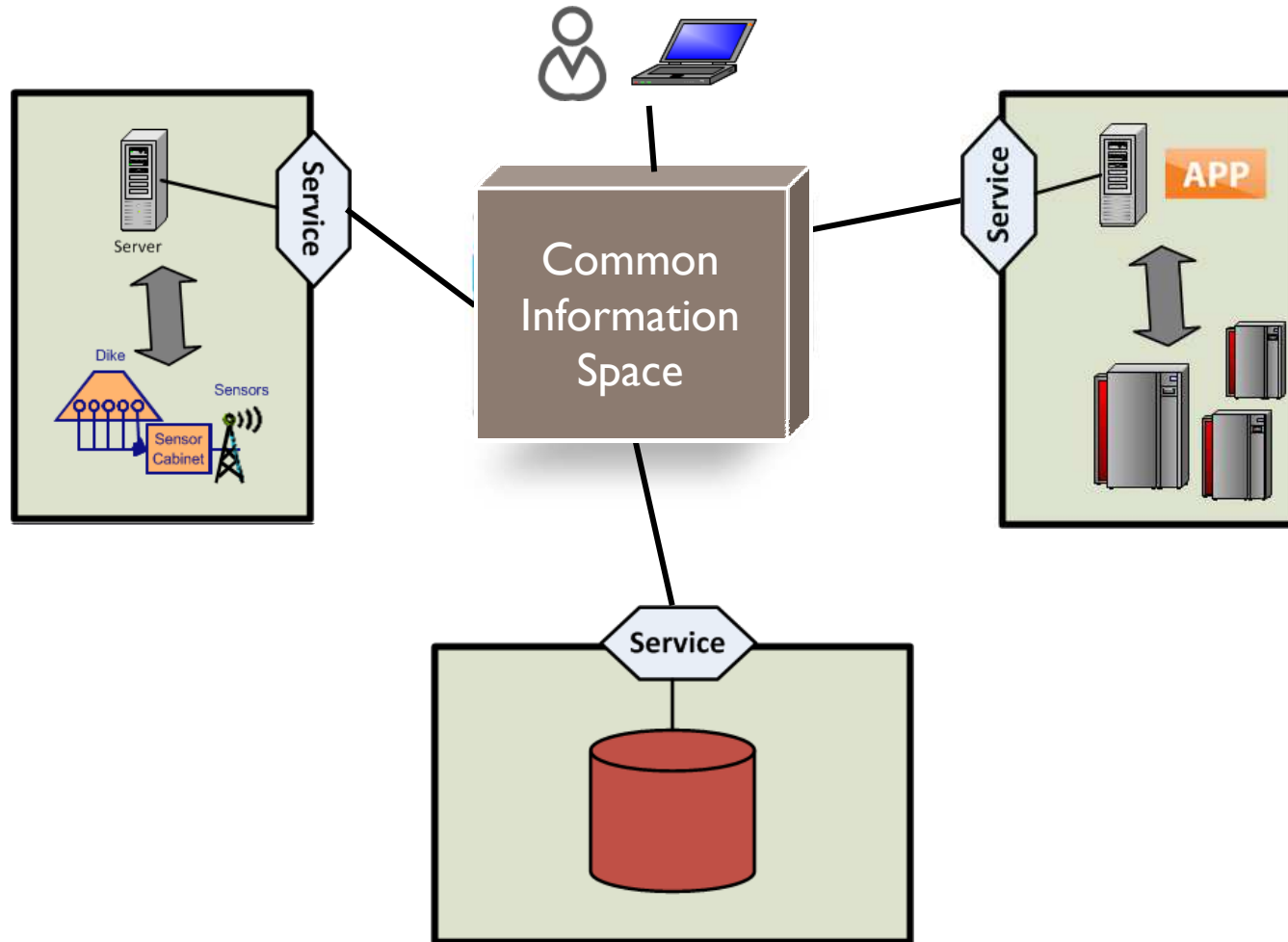


# Role of a software framework

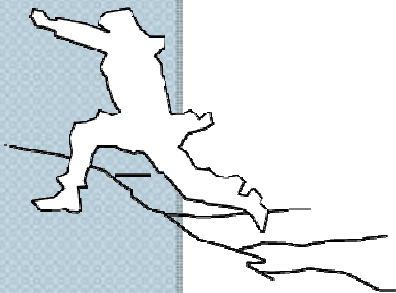
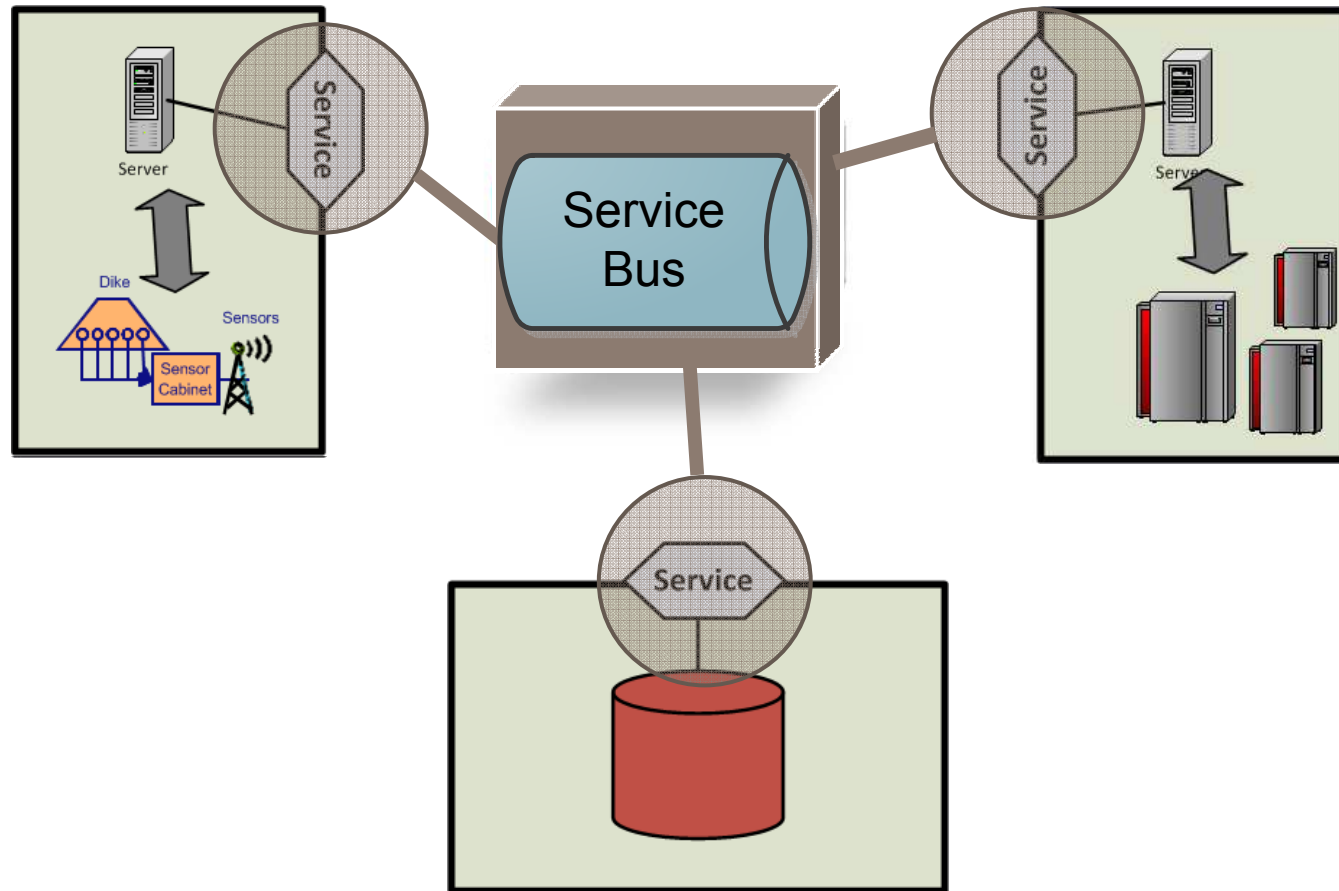
- Framework solves common problems of a specific type of systems
- Creators of concrete systems need to provide only system-specific logic
- CIS: software framework for Early Warning Systems



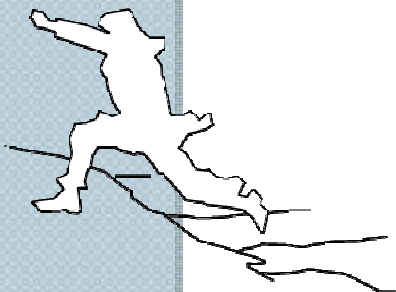
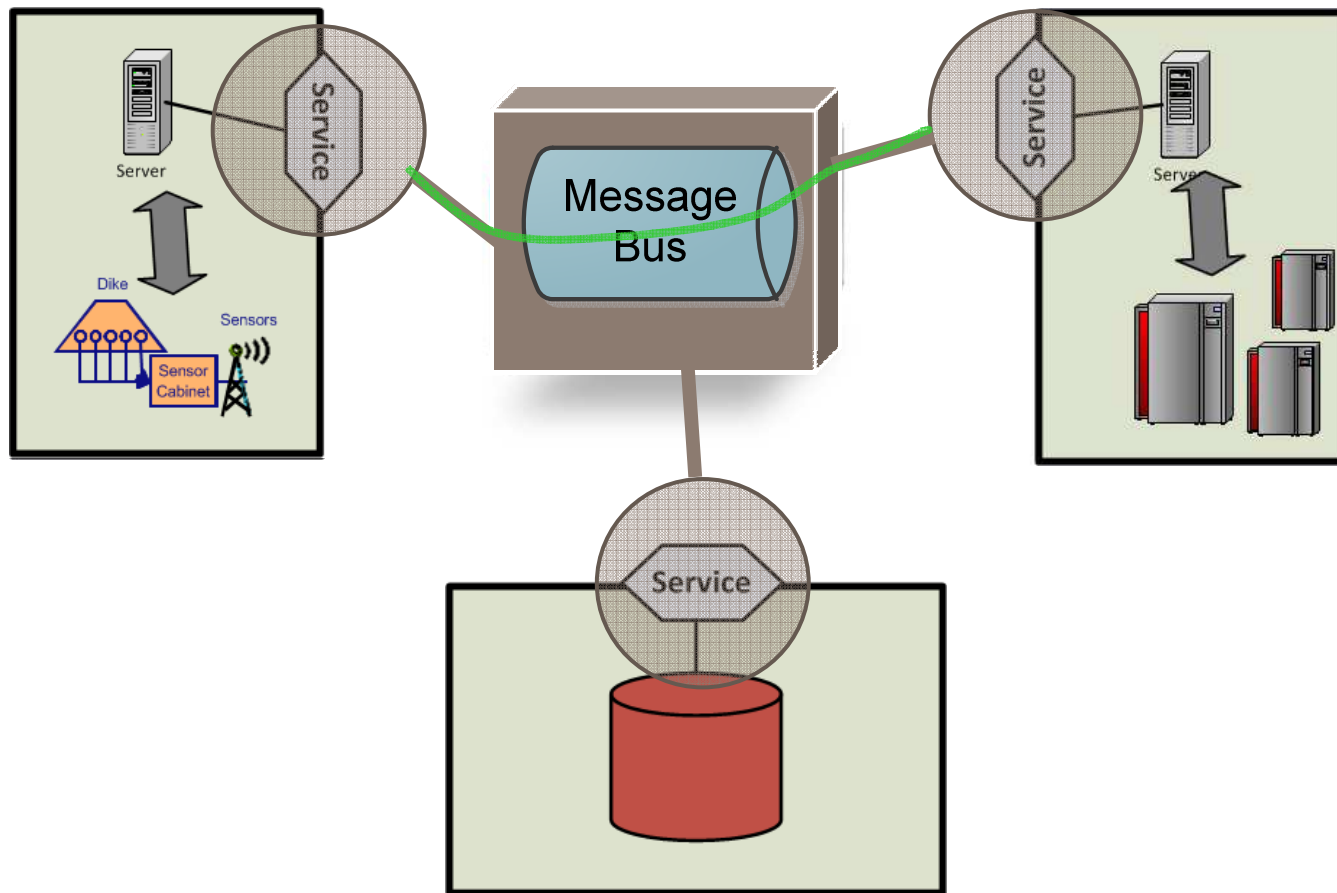
# Common Information Space



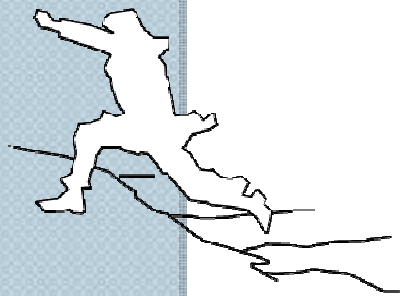
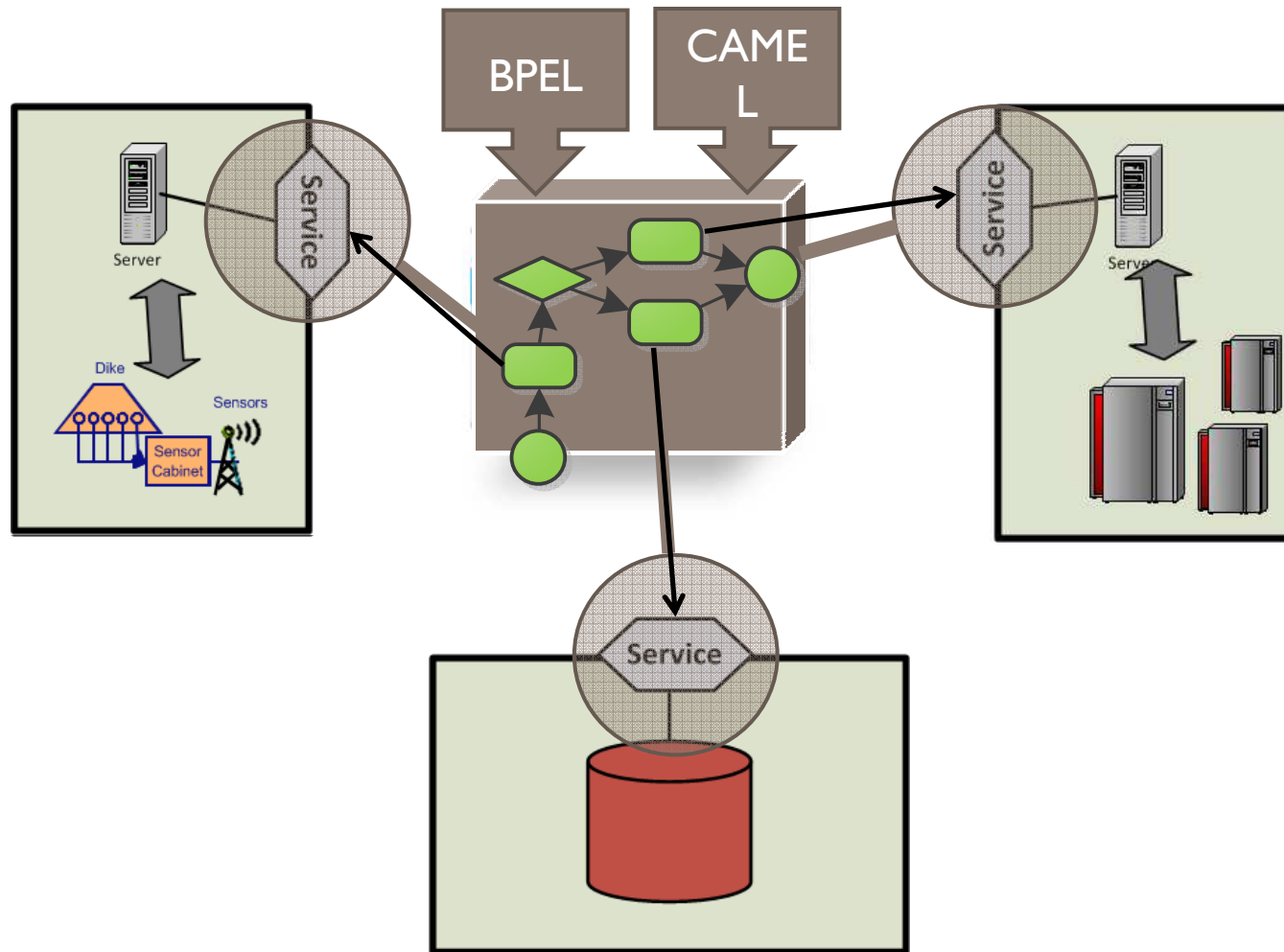
# CIS: Service foundations



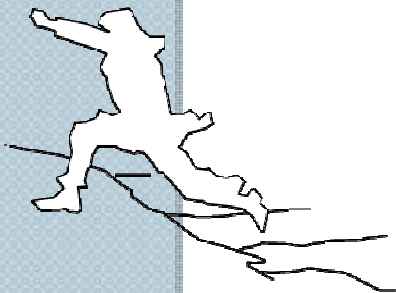
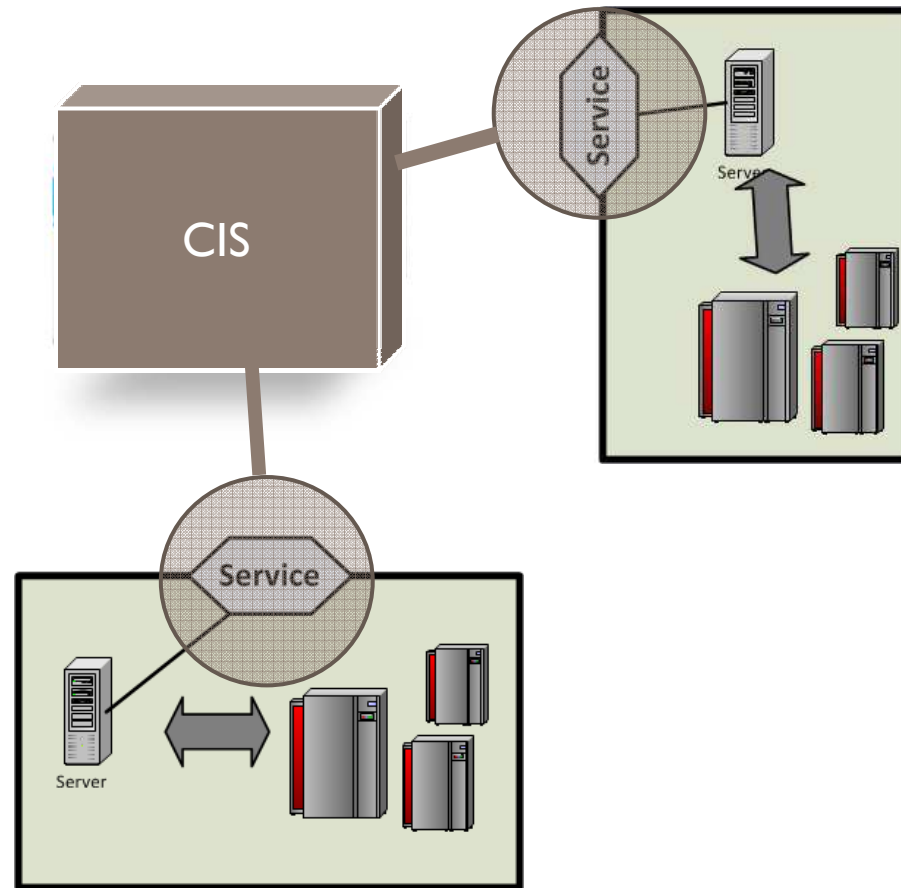
# CIS: Communication



# CIS: Coordination

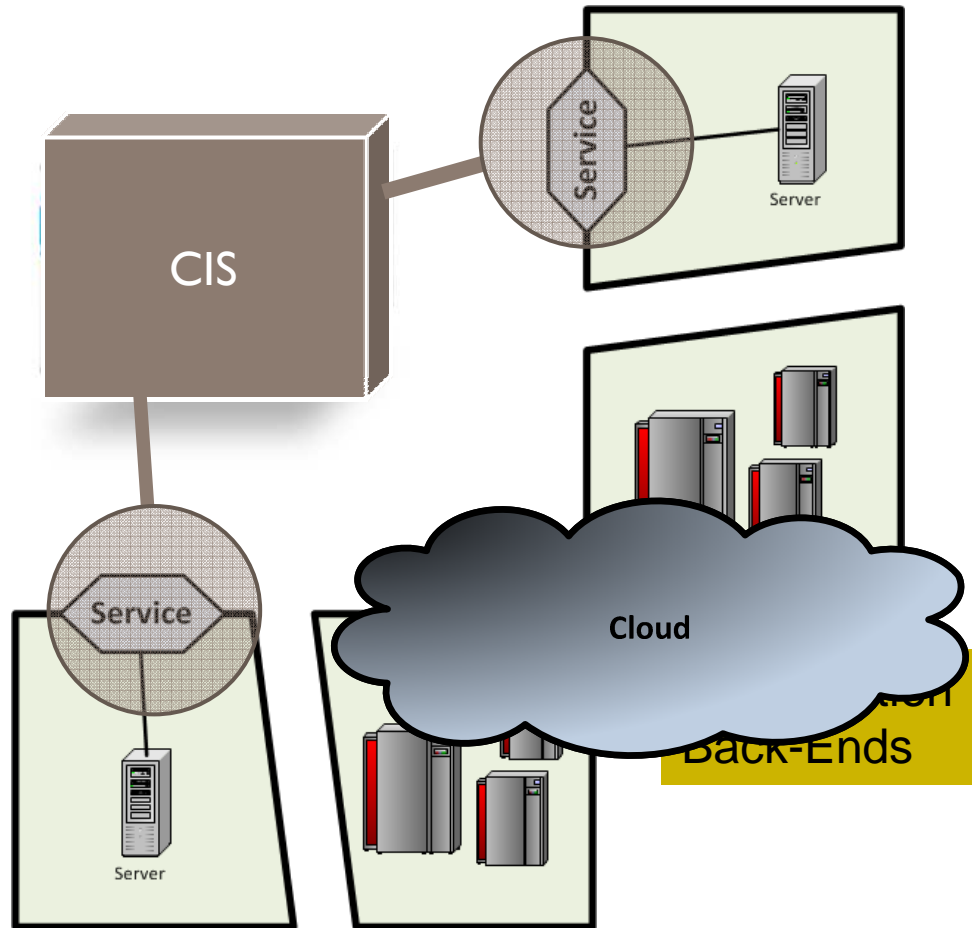


# CIS: Leveraging clouds



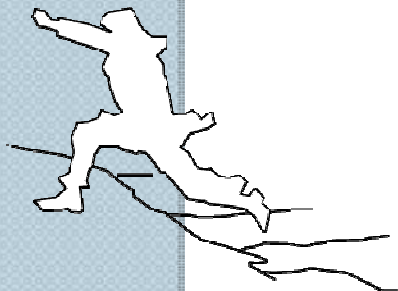
# CIS: Leveraging clouds

Service Front-End

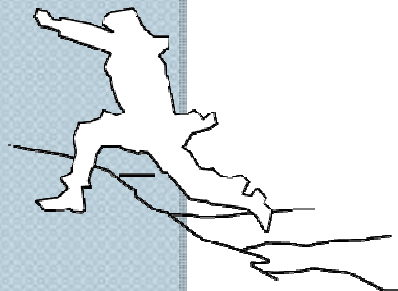
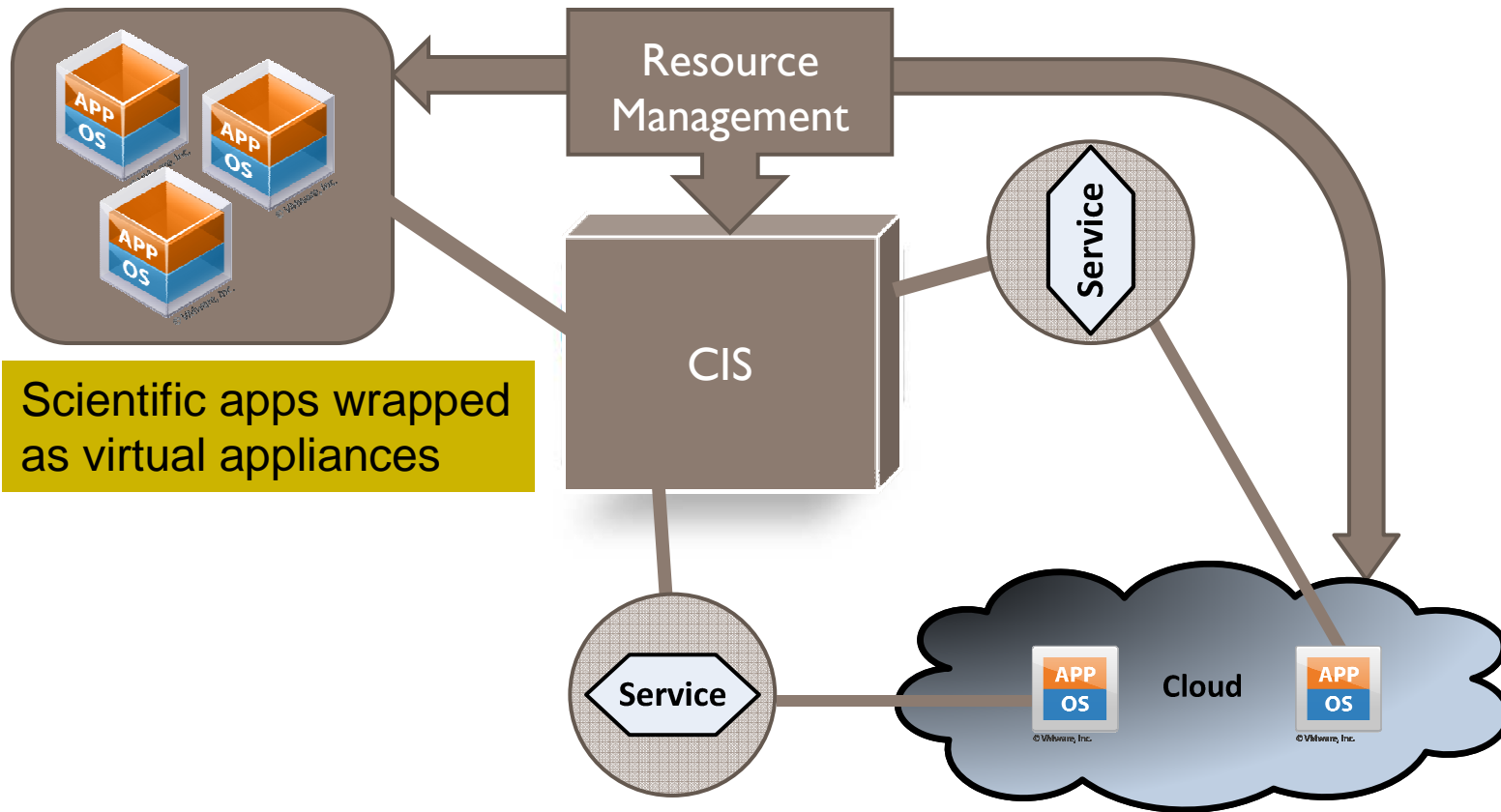


Service Front-End

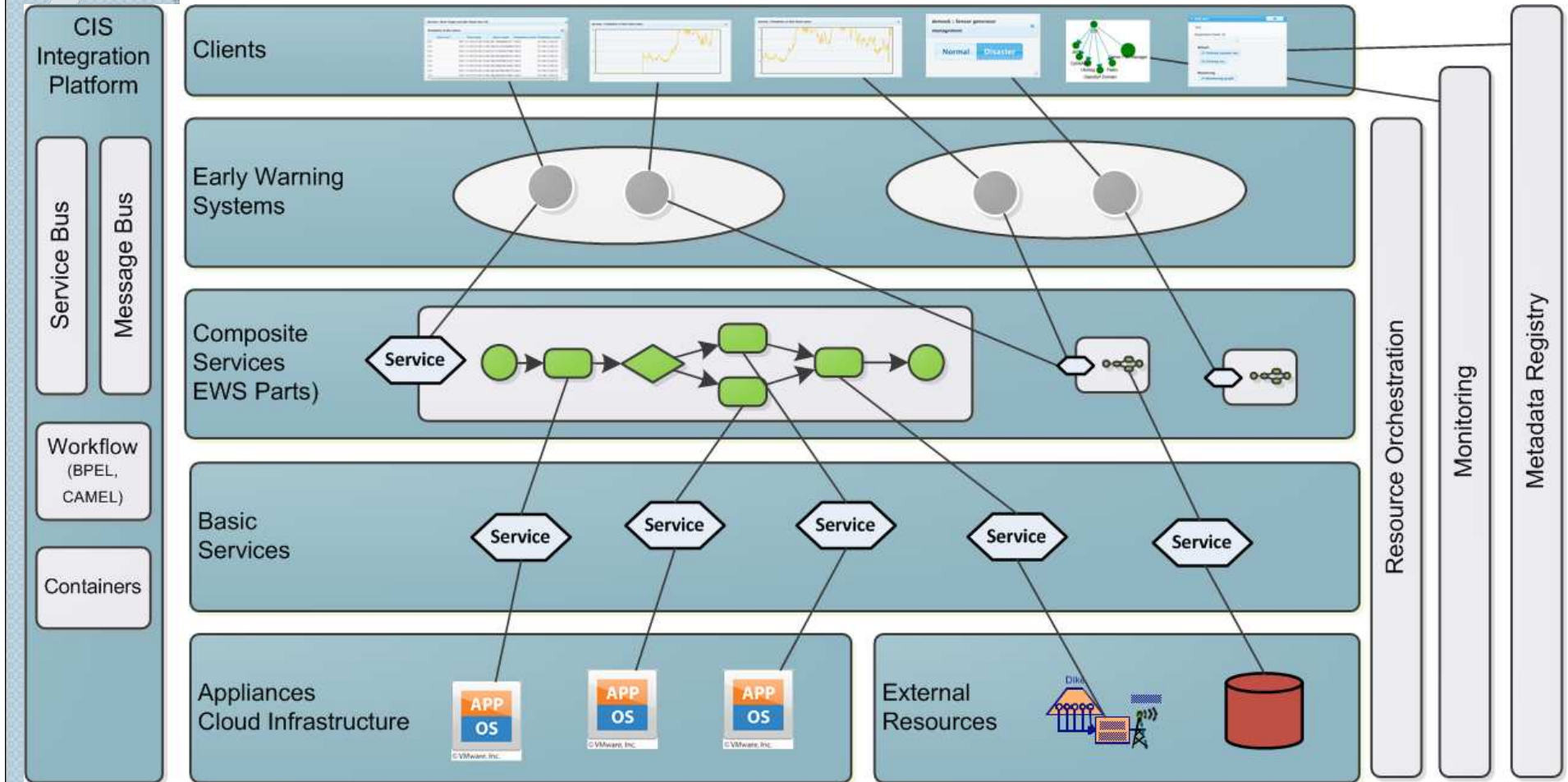
Back-Ends



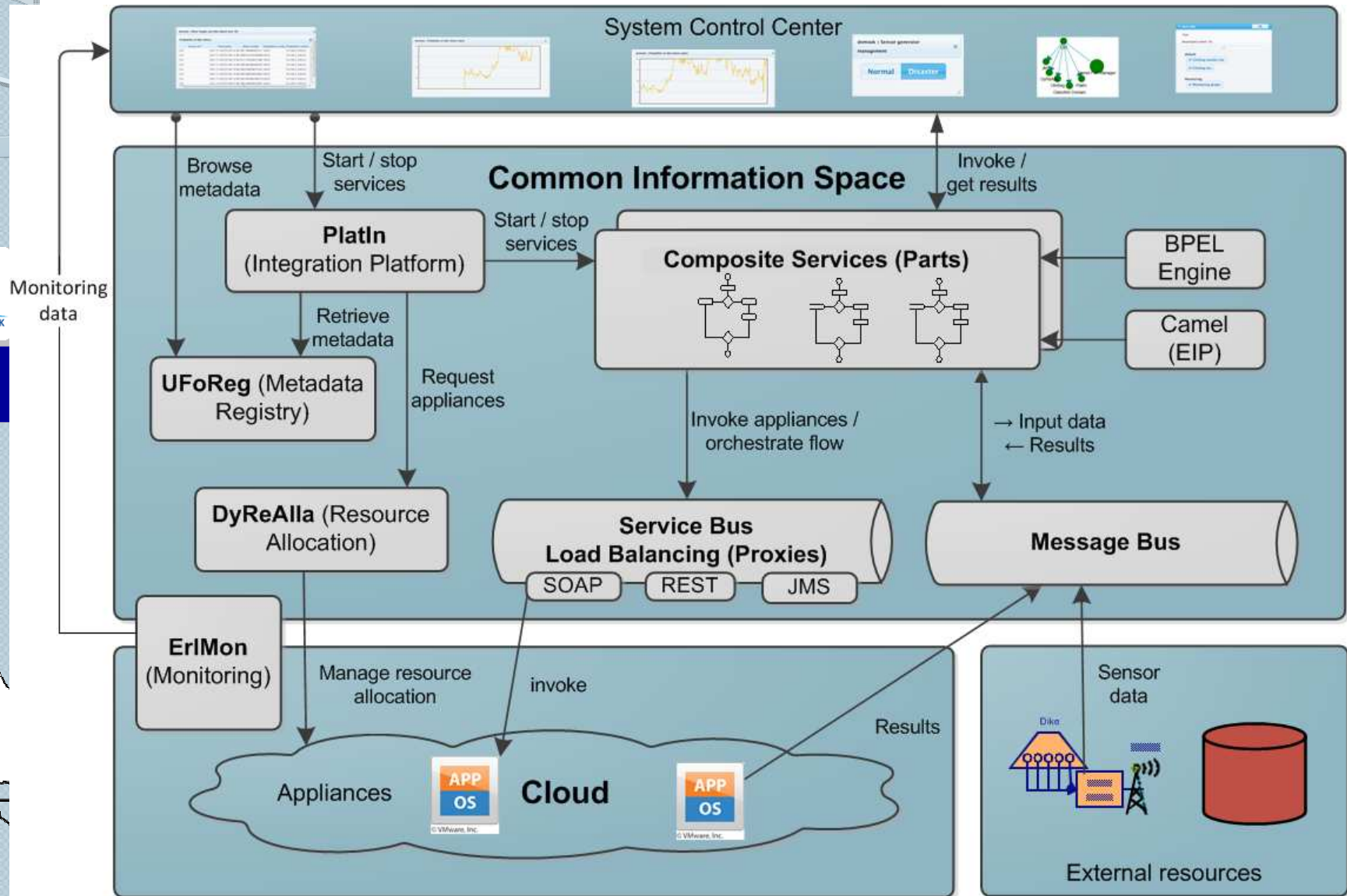
# CIS: Leveraging clouds



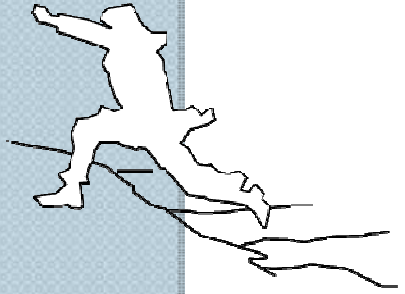
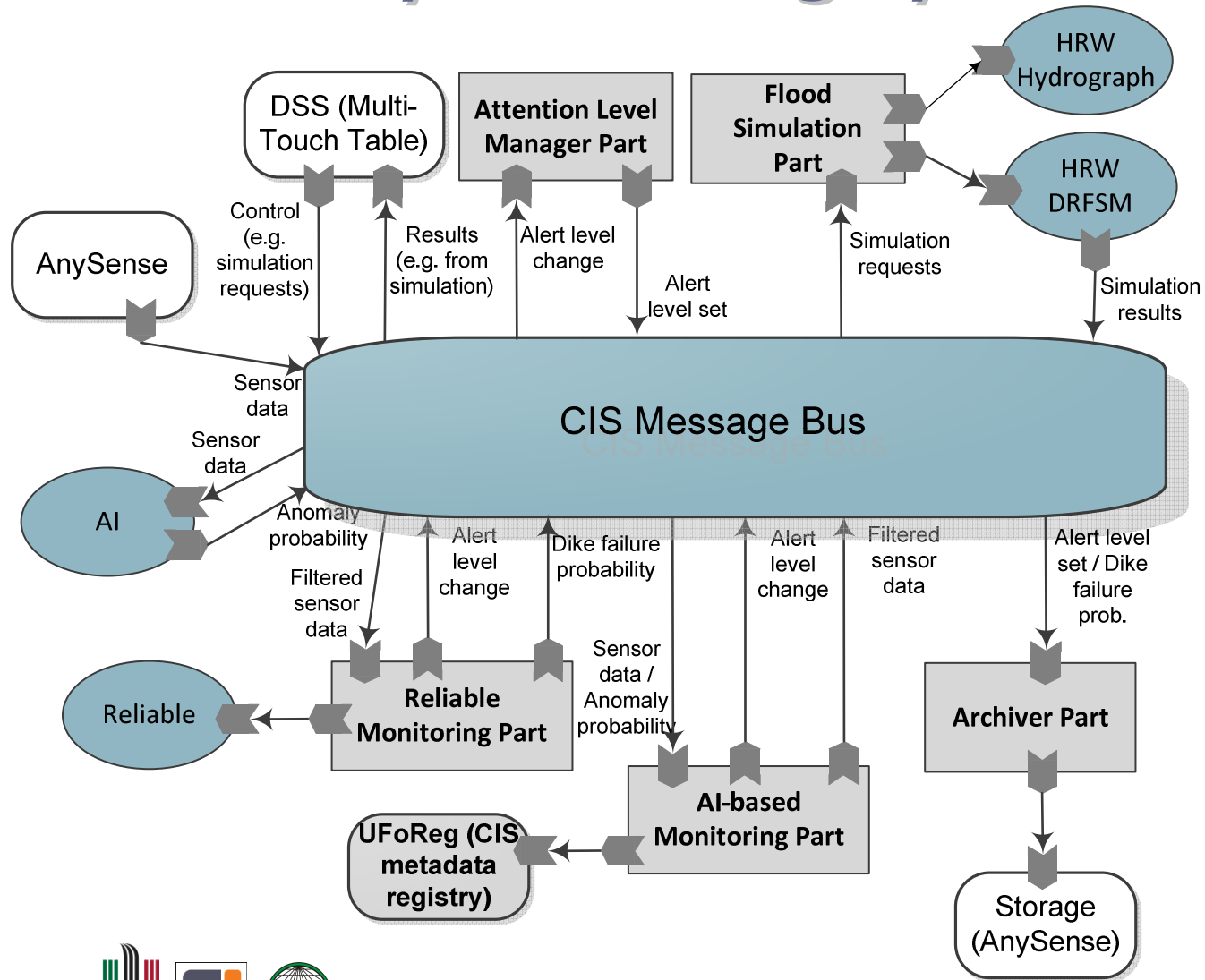
# CIS-powered System



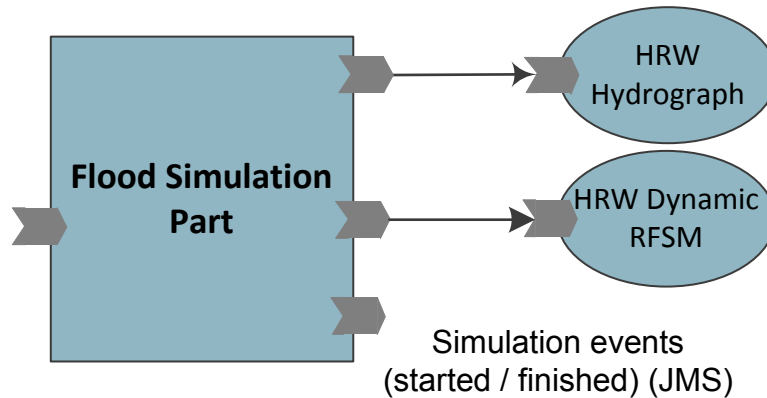
# CIS Architecture



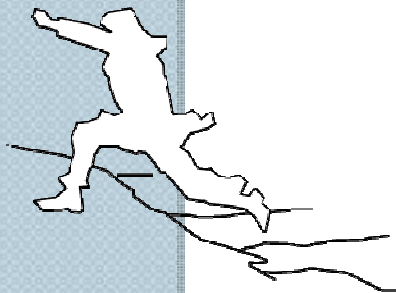
# Flood Early Warning System



# Service example: inundation simulation (Flood simulation part)

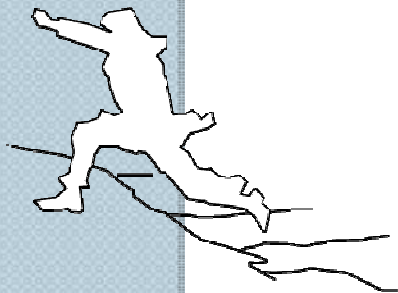


- Two legacy apps exposed as one convenient service
- BPEL integration
- Input: simulation specs
- Output: series of messages with simulation results



# Conclusion

- CIS helps quickly build complex systems by providing solutions to common problems
- Service-Oriented approach
- Management of computation backends – leveraging clouds





[urbanflood.cyfronet.pl](http://urbanflood.cyfronet.pl)  
[dice.cyfronet.pl/cis](http://dice.cyfronet.pl/cis)

