

Model-Driven Autonomic Architecture

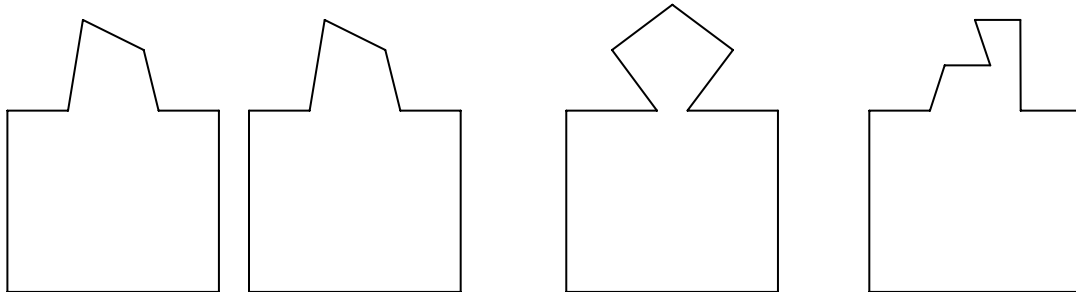
Radu Calinescu

Computing Laboratory, University of Oxford

Oxford University
Computing Laboratory



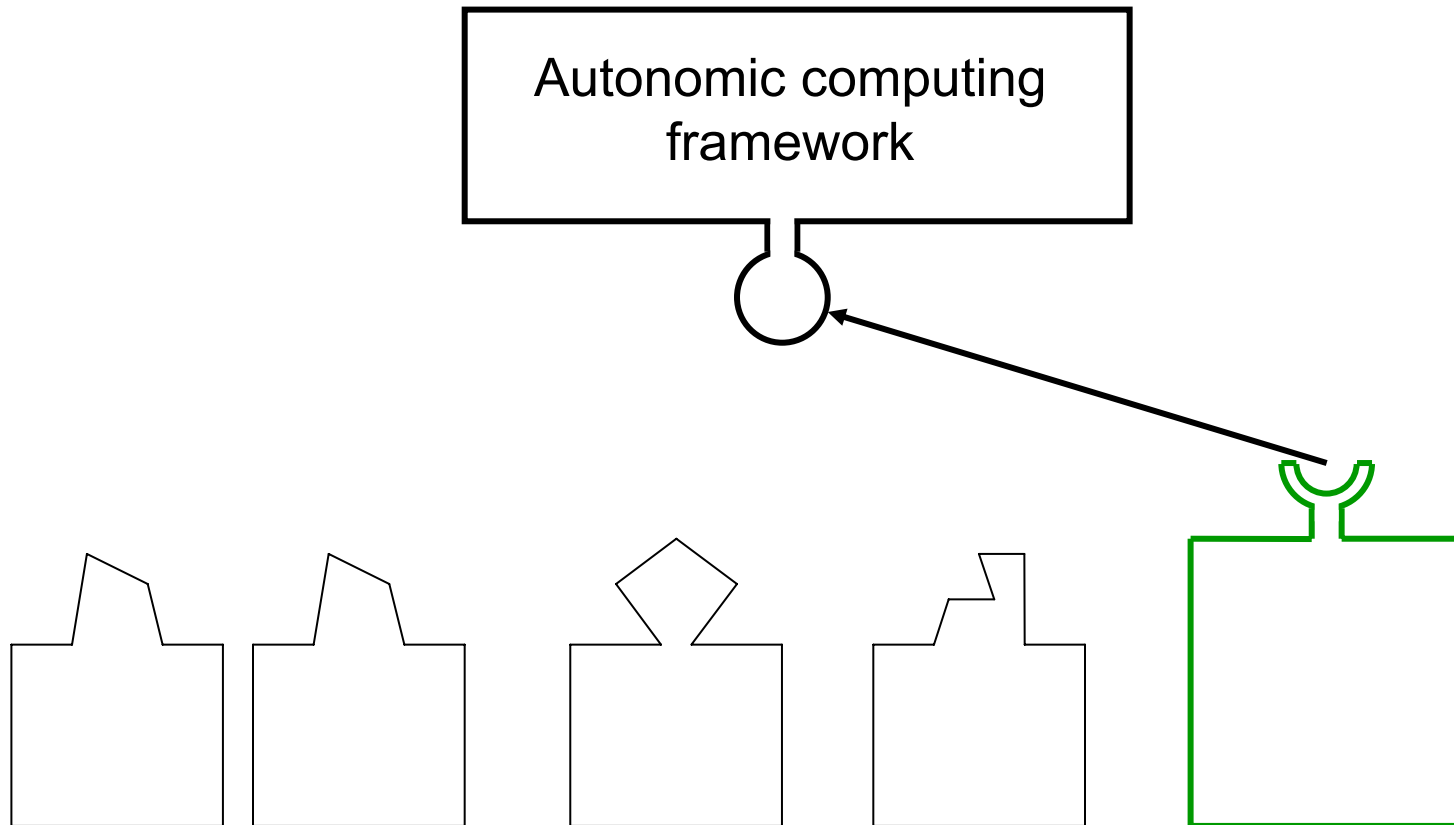
Introduction



- Non-standard resource state/configuration interfaces



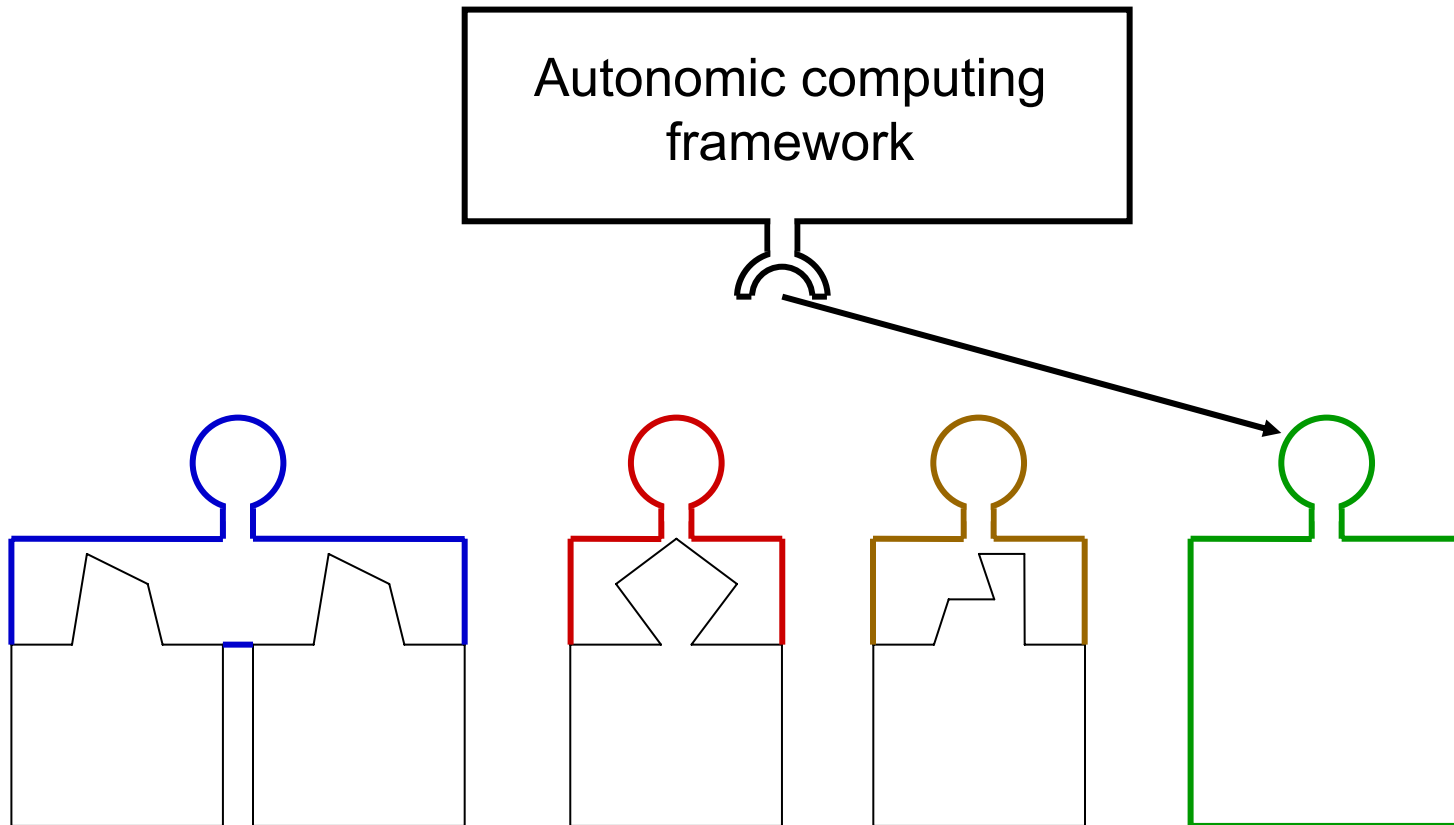
Introduction



- Non-standard resource state/configuration interfaces
- Autonomic-enabled resources use the APIs of an autonomic computing framework



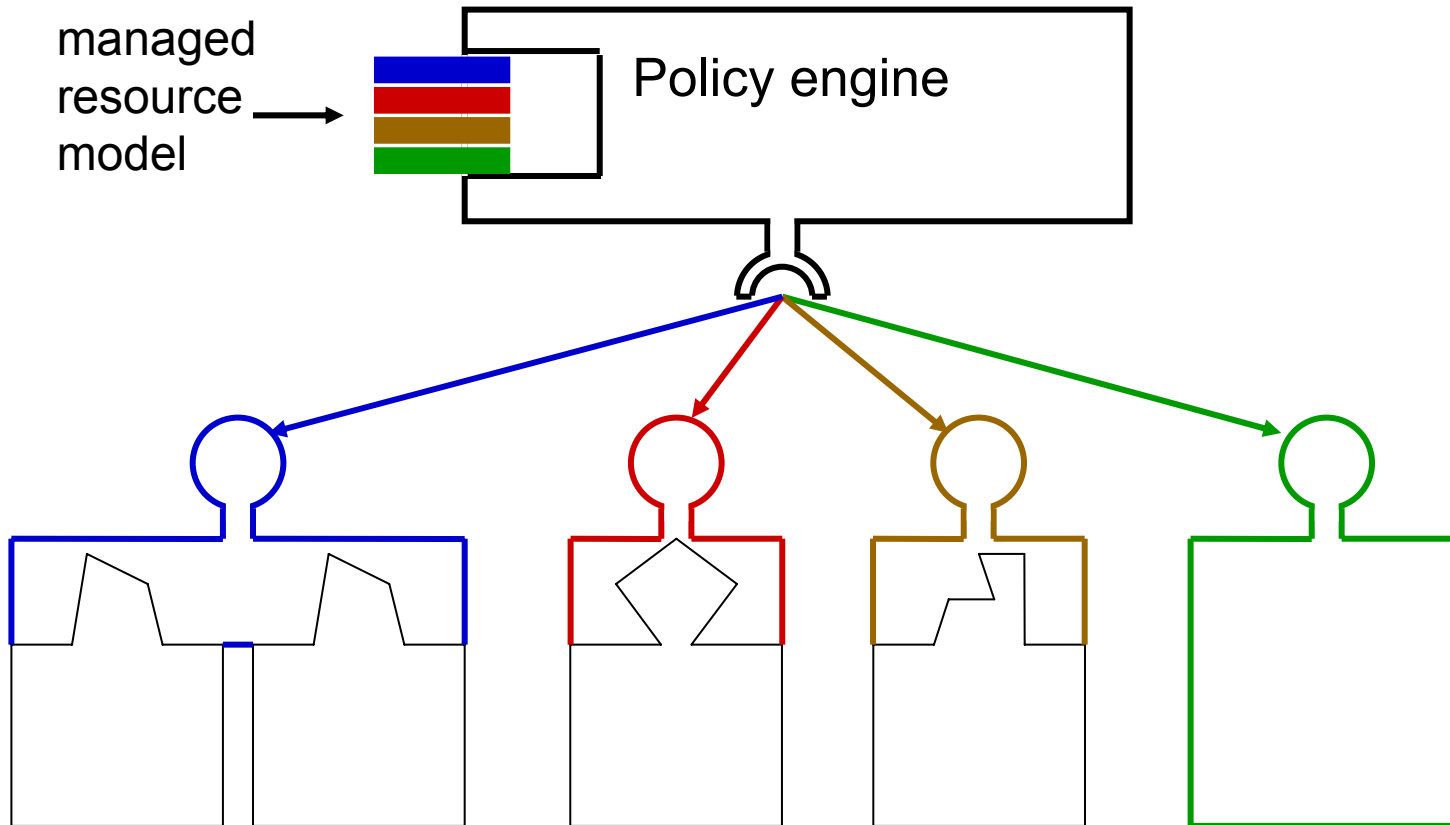
Objectives



- Expose “legacy” resources through a thin manageability adaptor (e.g., WSDM-based)



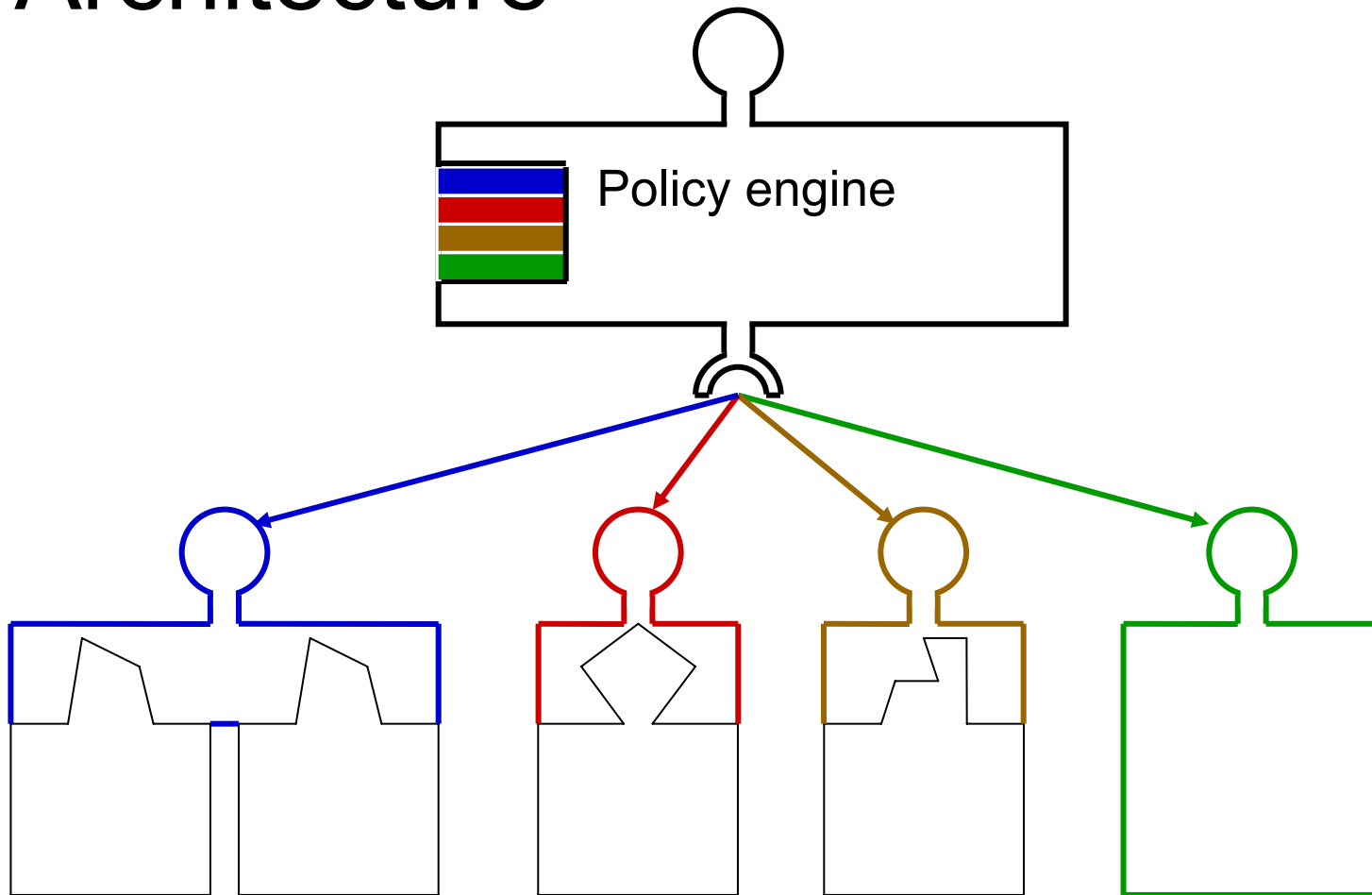
Architecture



- Expose “legacy” resources through a thin manageability adaptor (e.g., WSDM-based)
- Configure *policy engine* using resource model



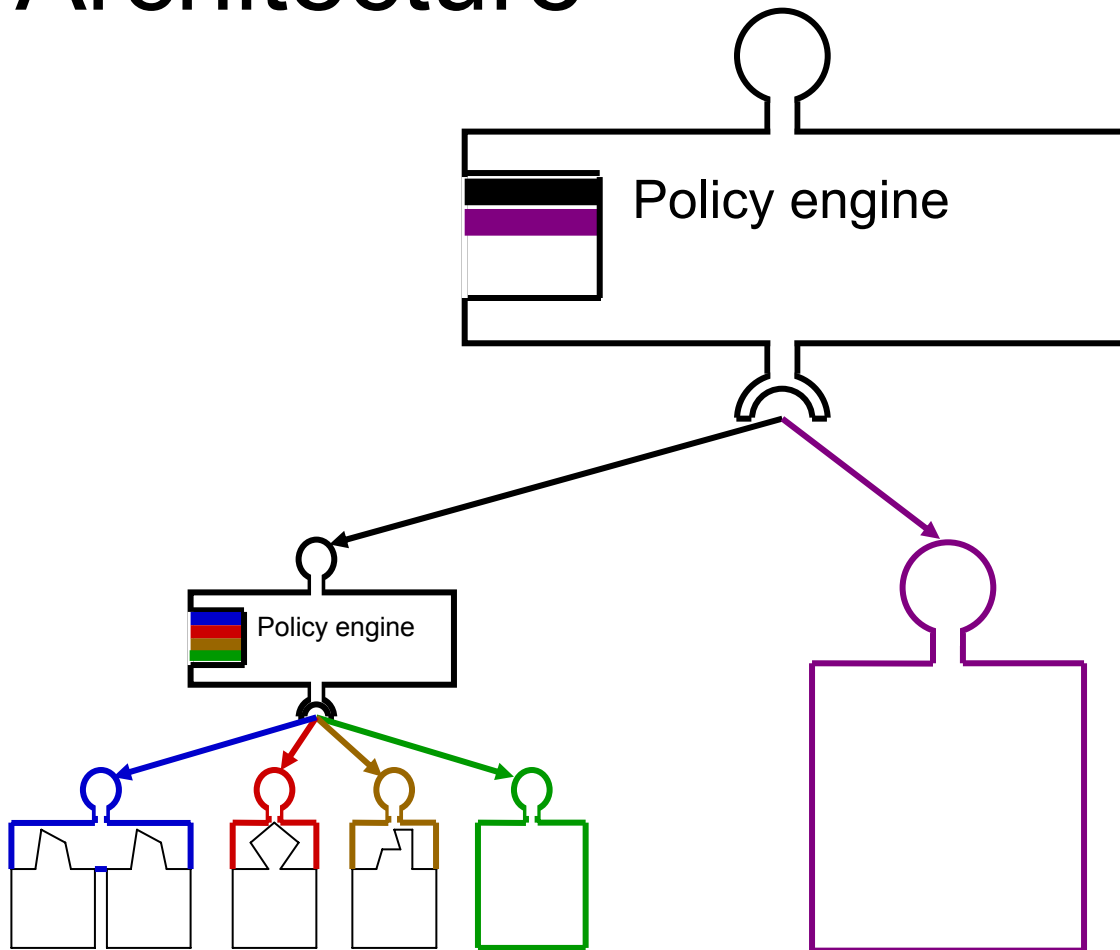
Architecture



- Configure *policy engine* using resource model
- Expose policy engine functionality through the same manageability interface...



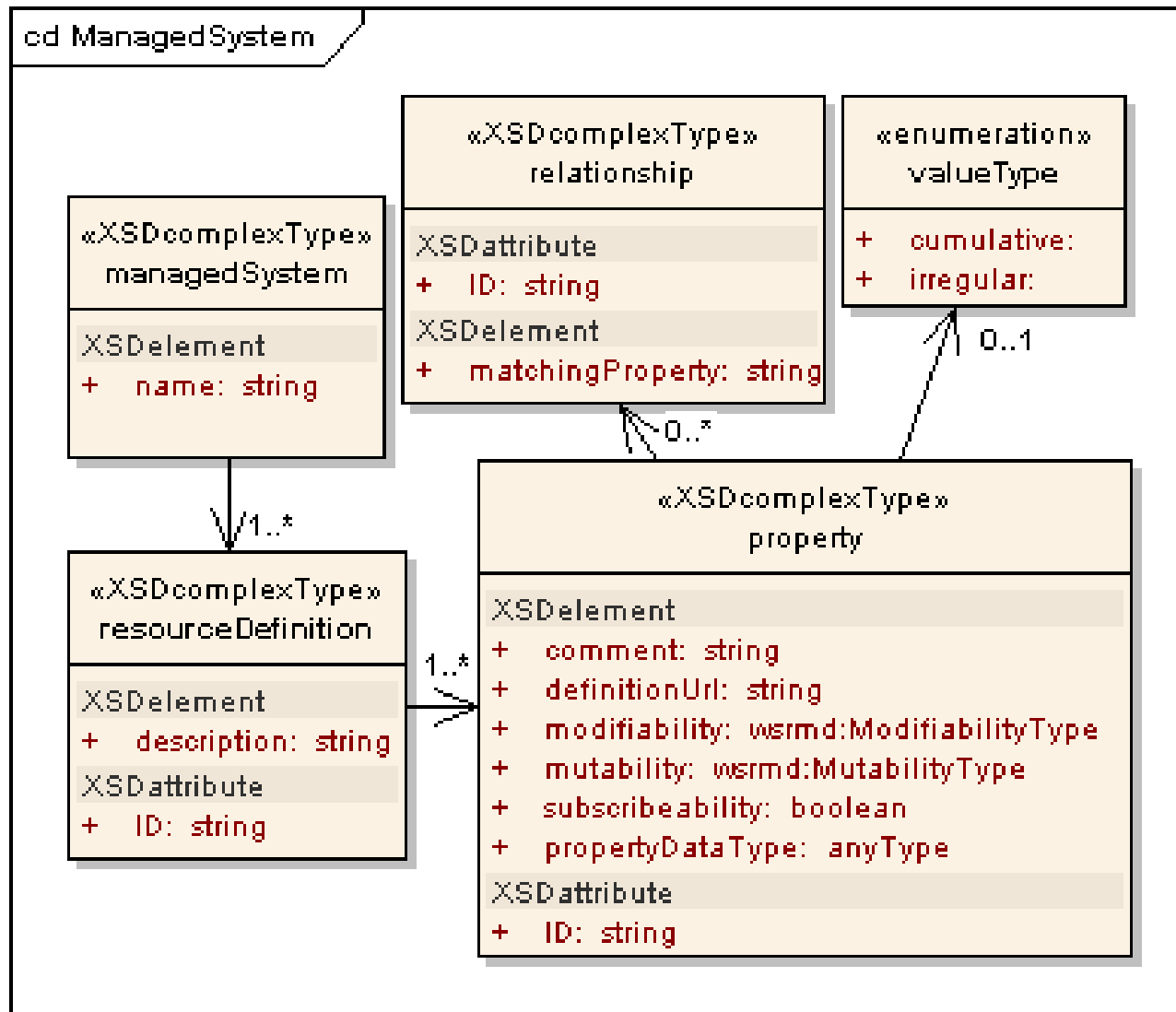
Architecture



- Expose policy engine functionality through the same manageability interface...
- ...so that it can be incorporated into a *system of systems*



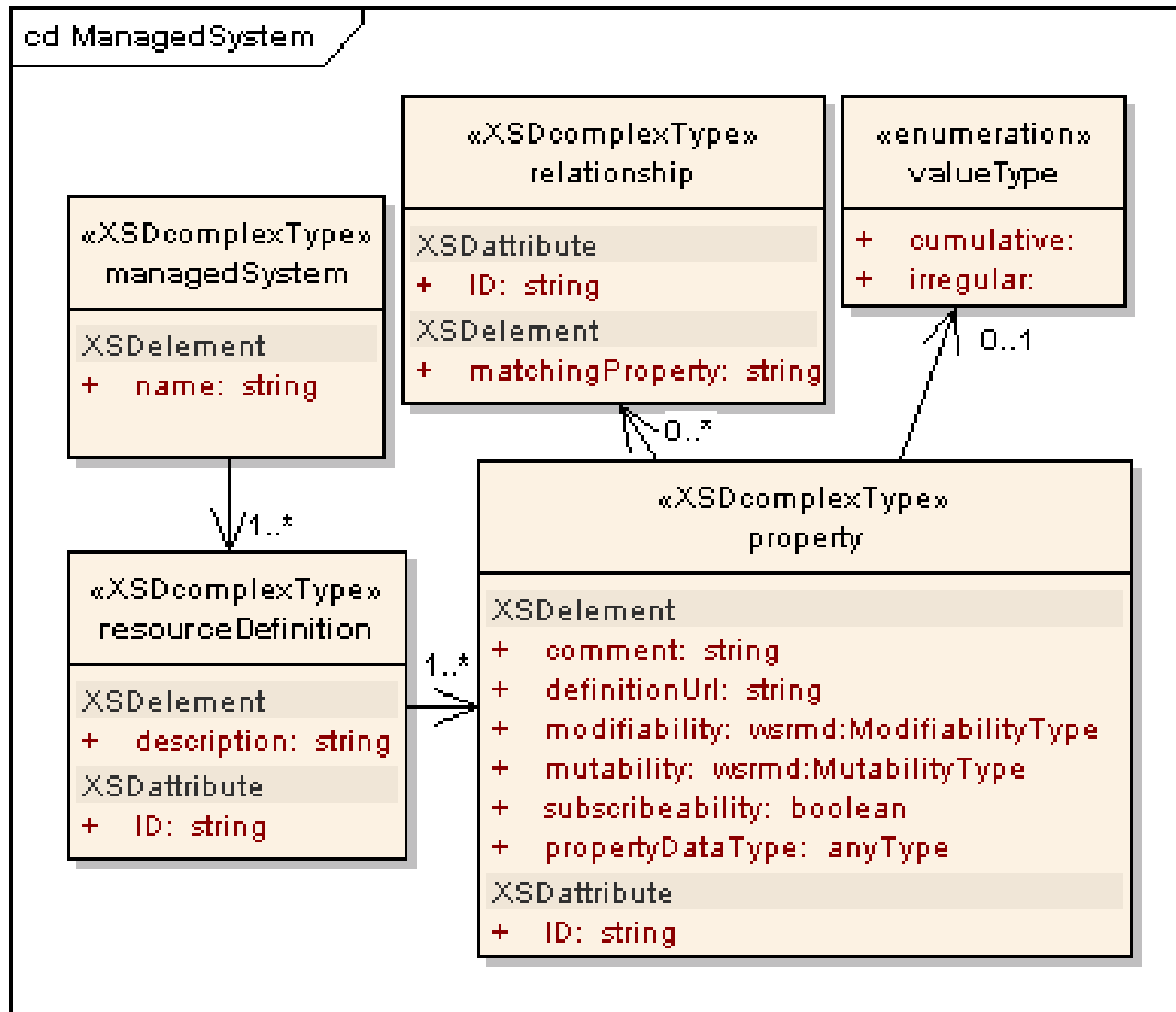
Ongoing work



- Managed system metamodel



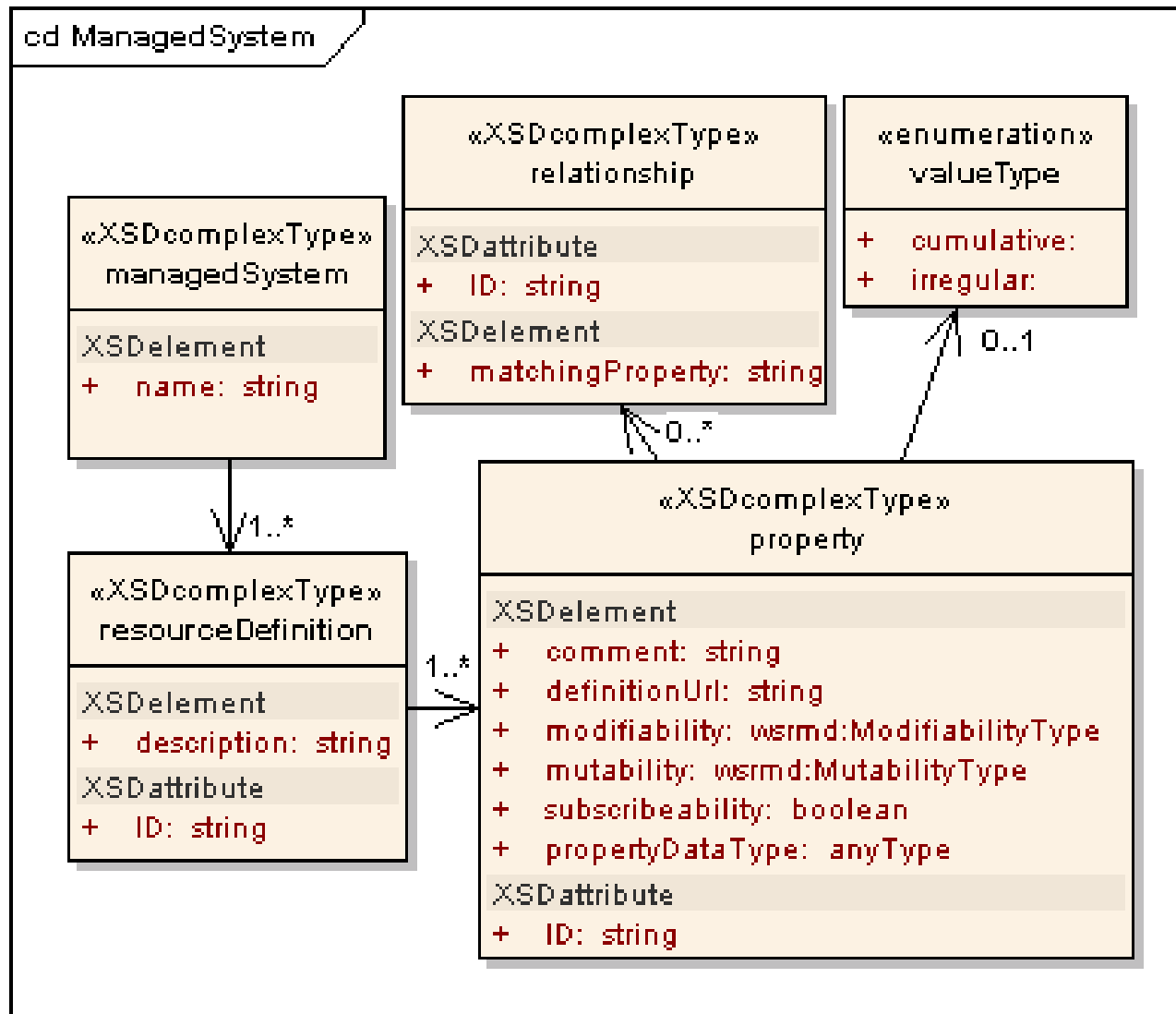
Ongoing work



- Managed system metamodel
- Data centre resource management case studies



Ongoing and future work



- Managed system metamodel
- Data centre resource management case studies
- IT resource metadata repository

