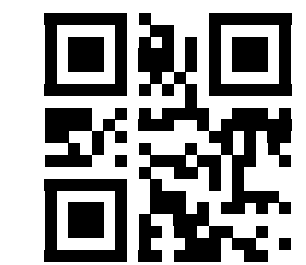
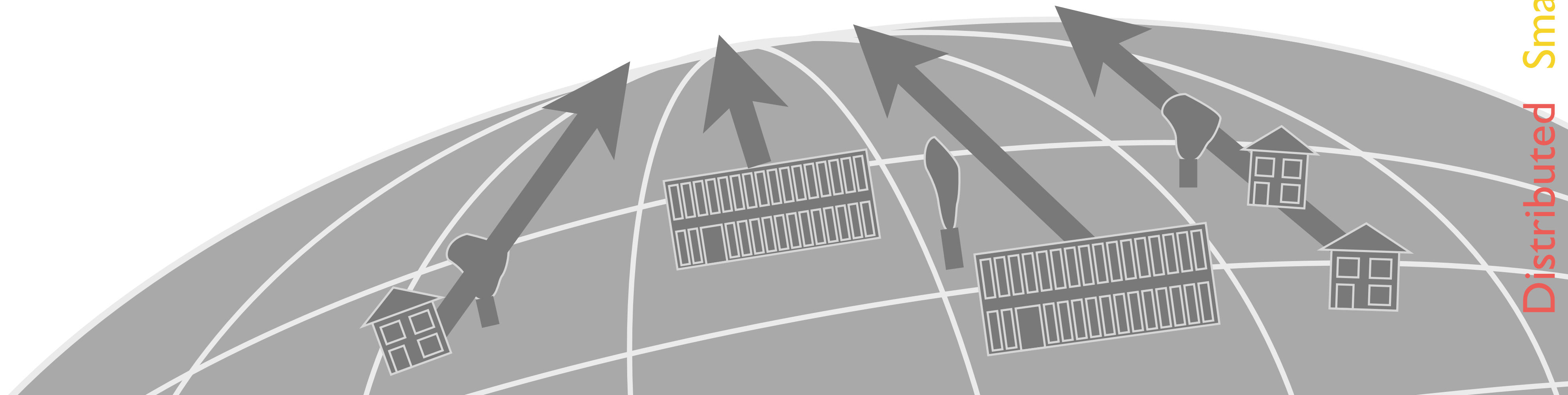




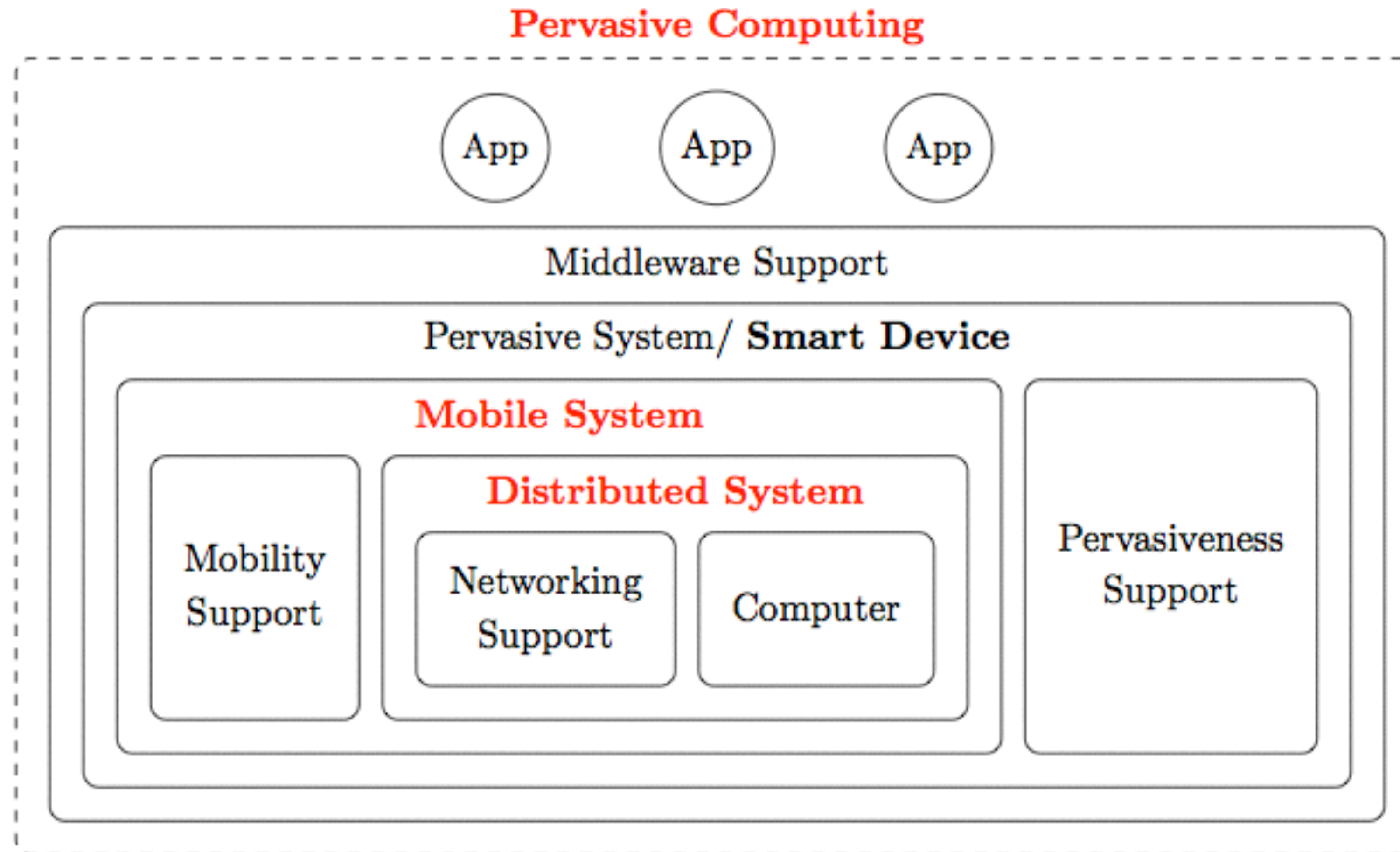
Distributed Smart Space Orchestration System

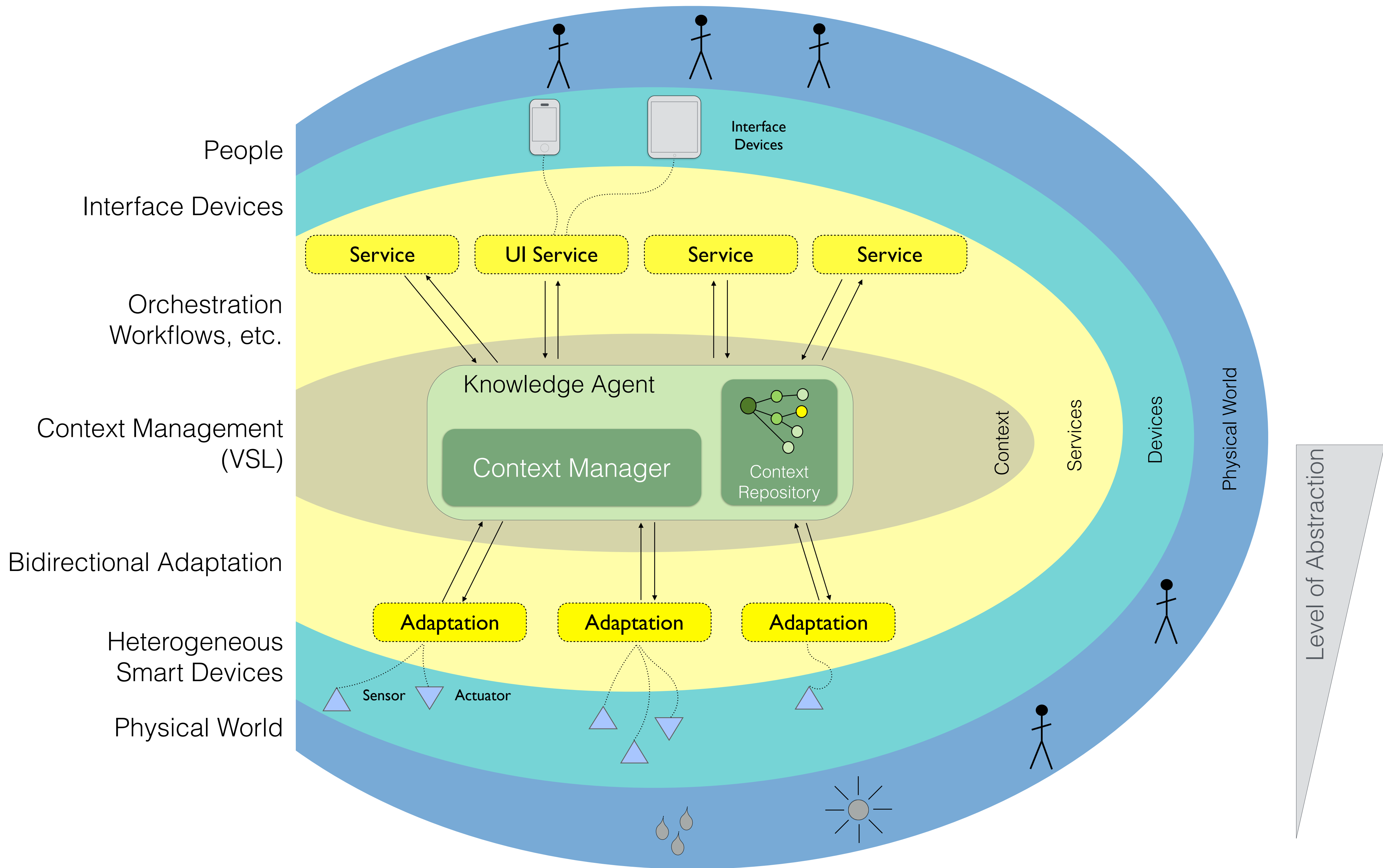
Marc-Oliver Pahl



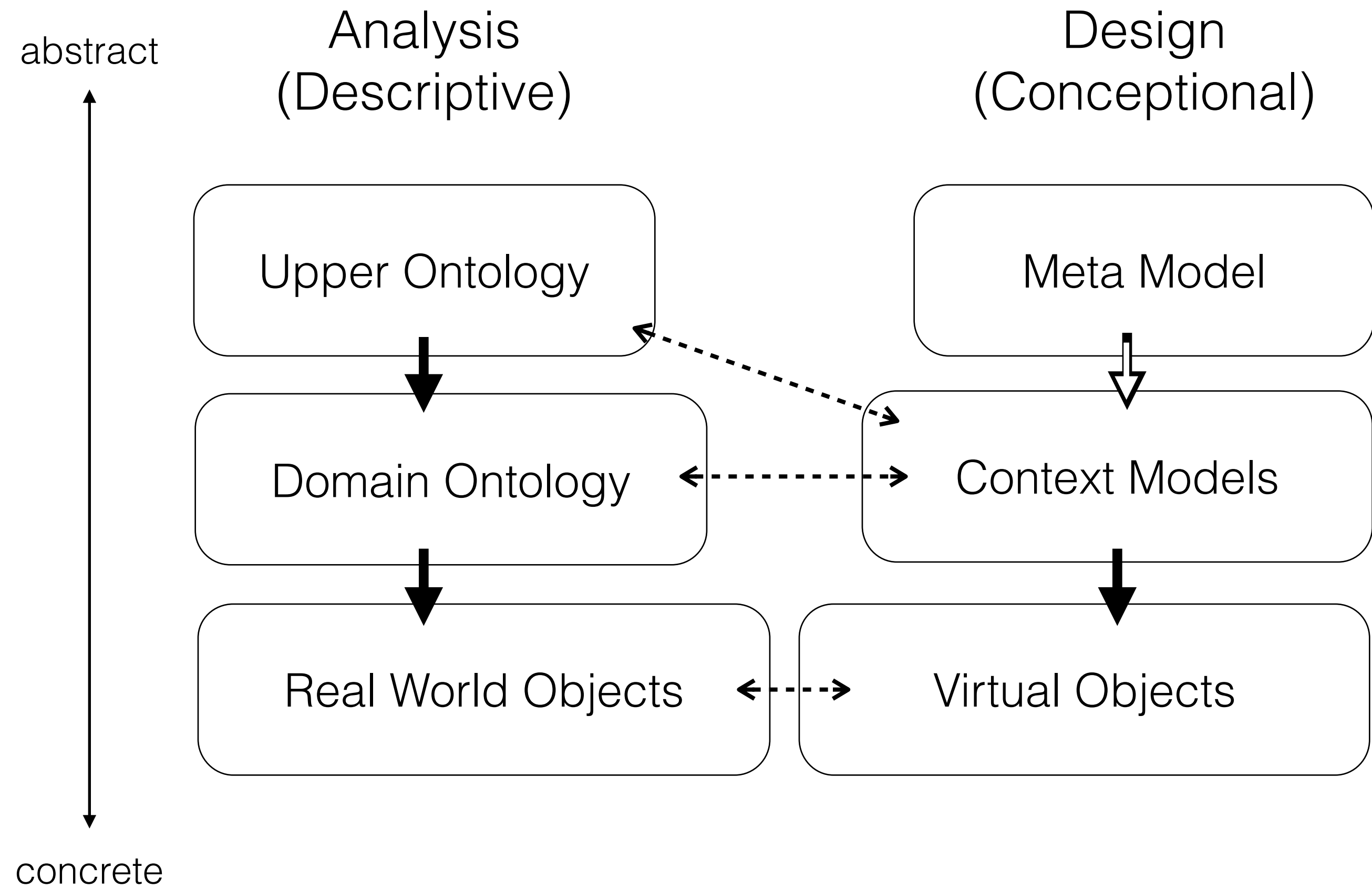
ds2os.org/

Architectural Components of PerCom





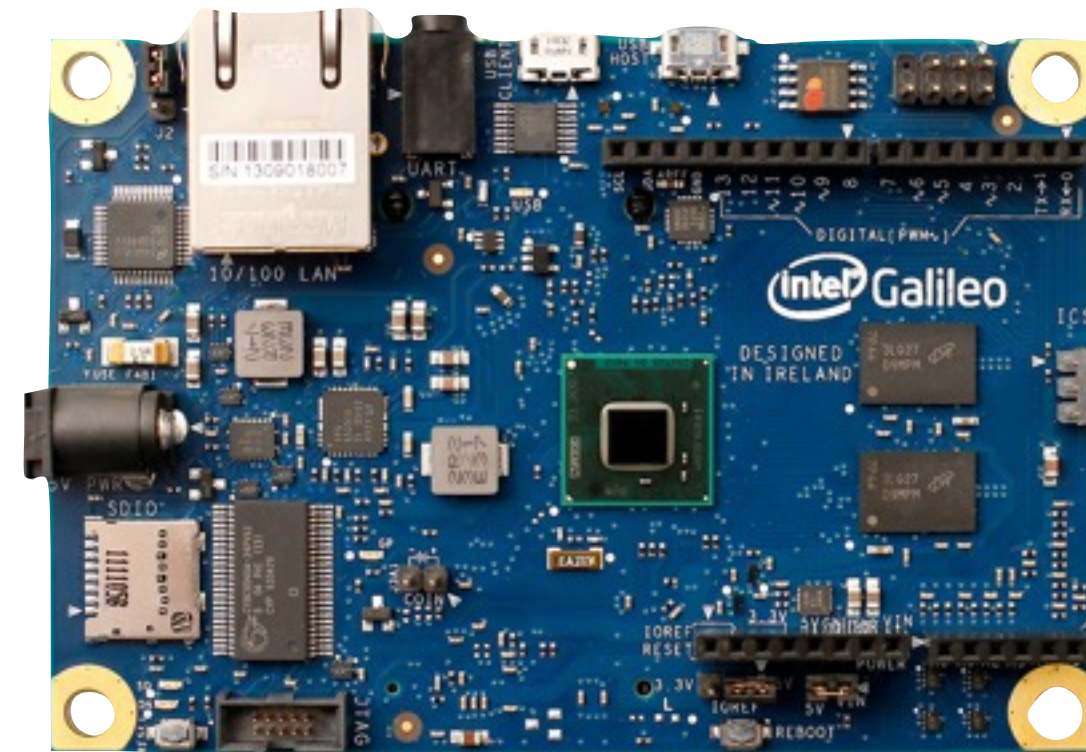
Context



Legend: $\leftarrow \text{---} \rightarrow$ Correspondence \downarrow Entity Inheritance ("is-a") \downarrow Uses

Middleware

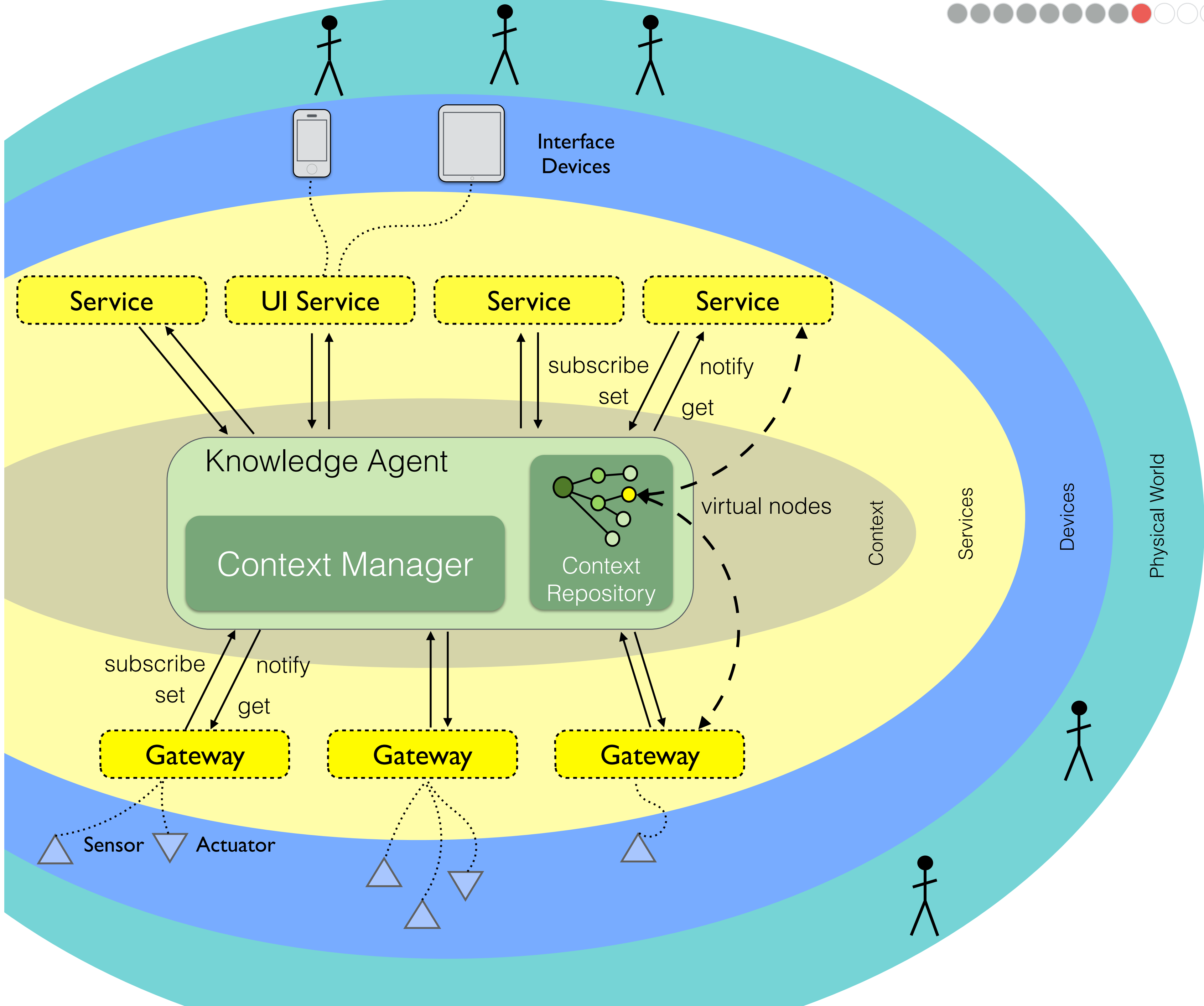
- Access Transparency
- Location Transparency
- Concurrency Transparency
- Migration Transparency
- Relocation Transparency
- Failure Transparency
- Persistence Transparency



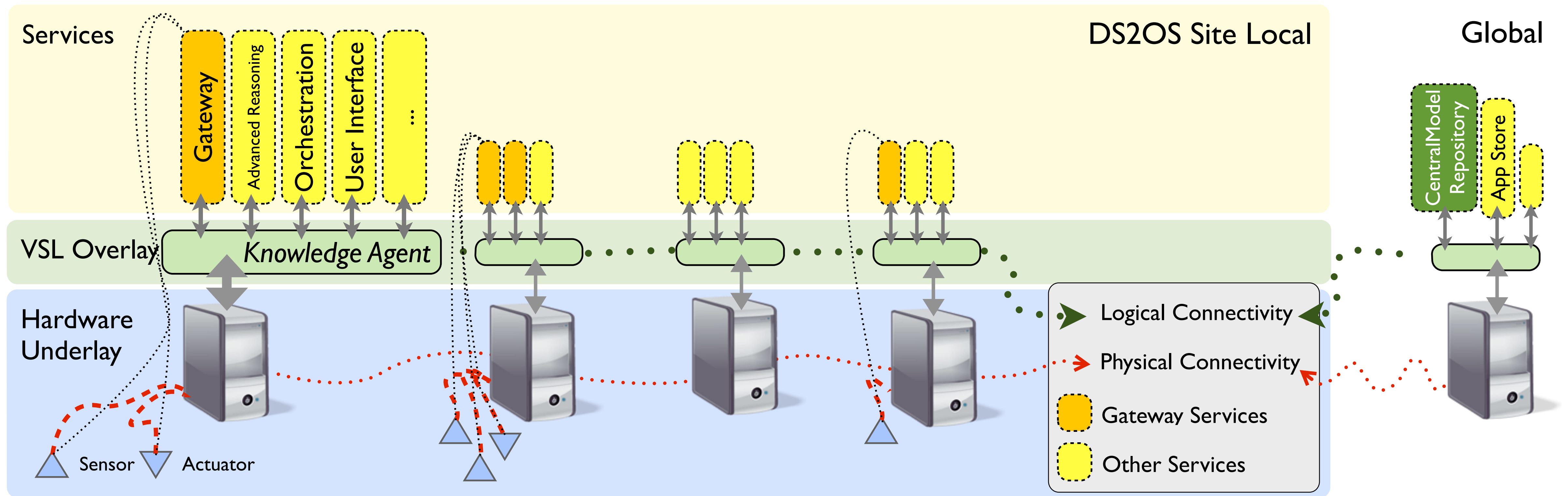
Adaptation

Level of Abstraction

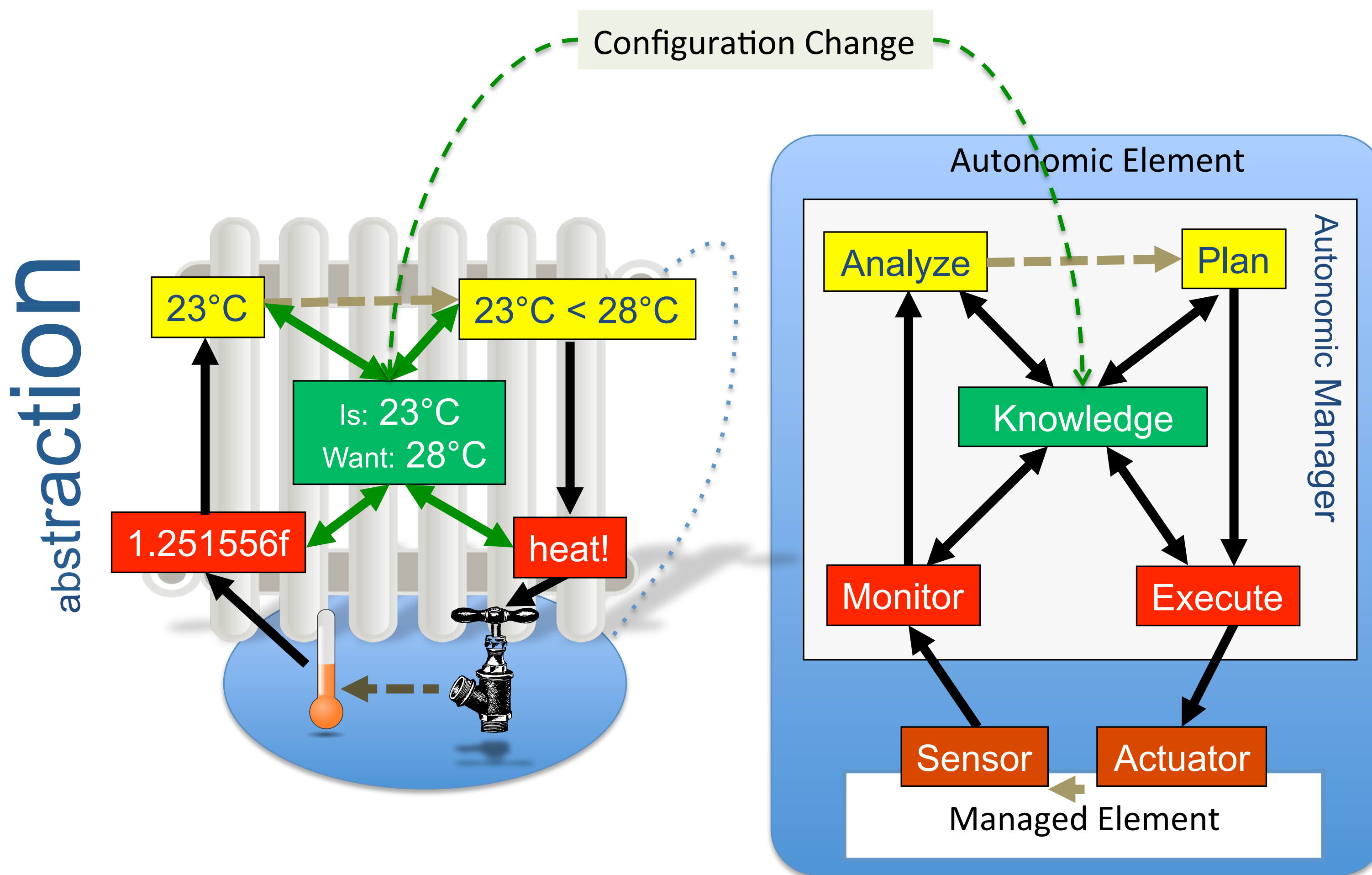
People
Interface Devices
Orchestration Workflows, etc.
Context Management
Bidirectional Adaptation
Heterogeneous Smart Devices
Physical World



Architecture

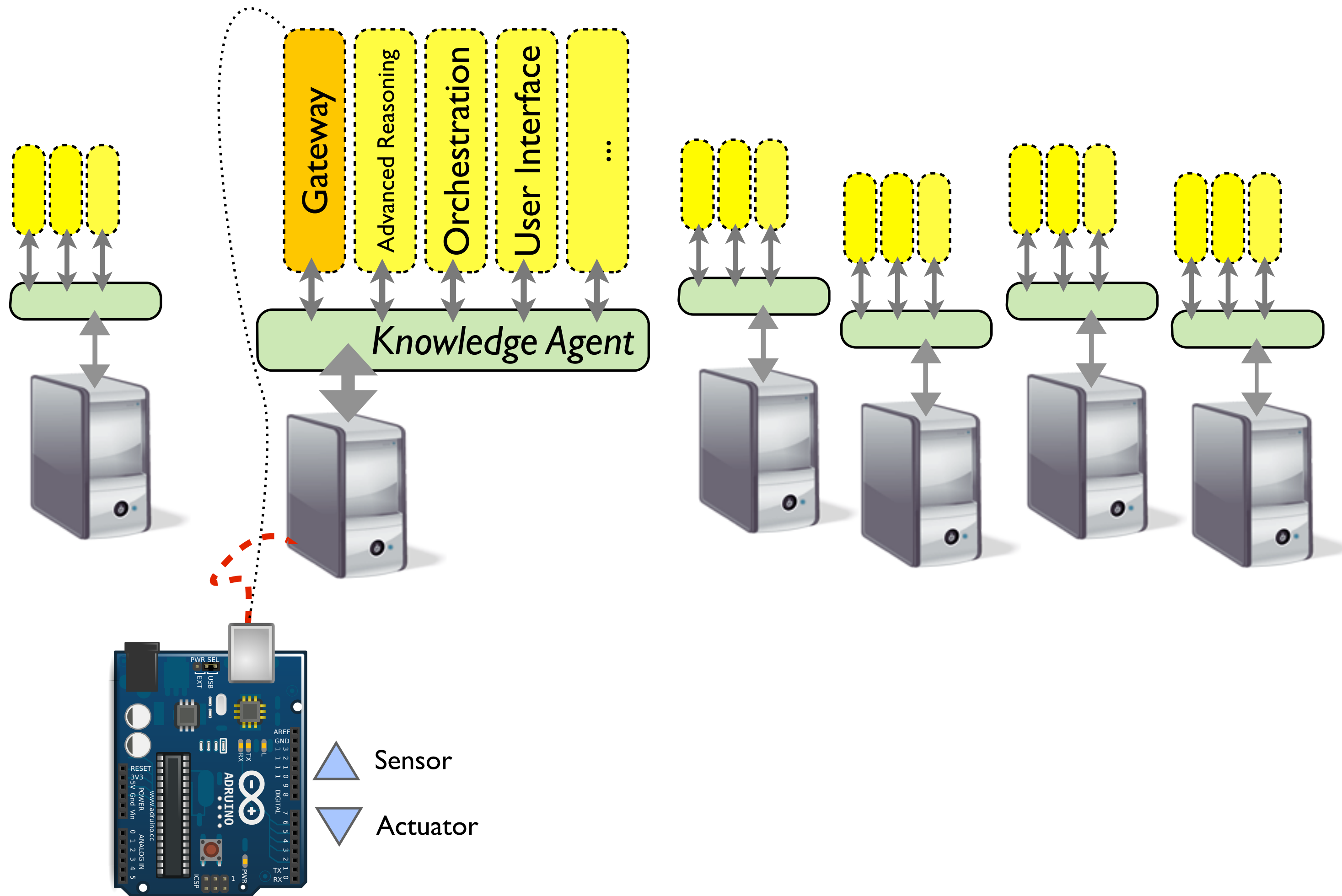


Autonomous Management



abstraction

Setup

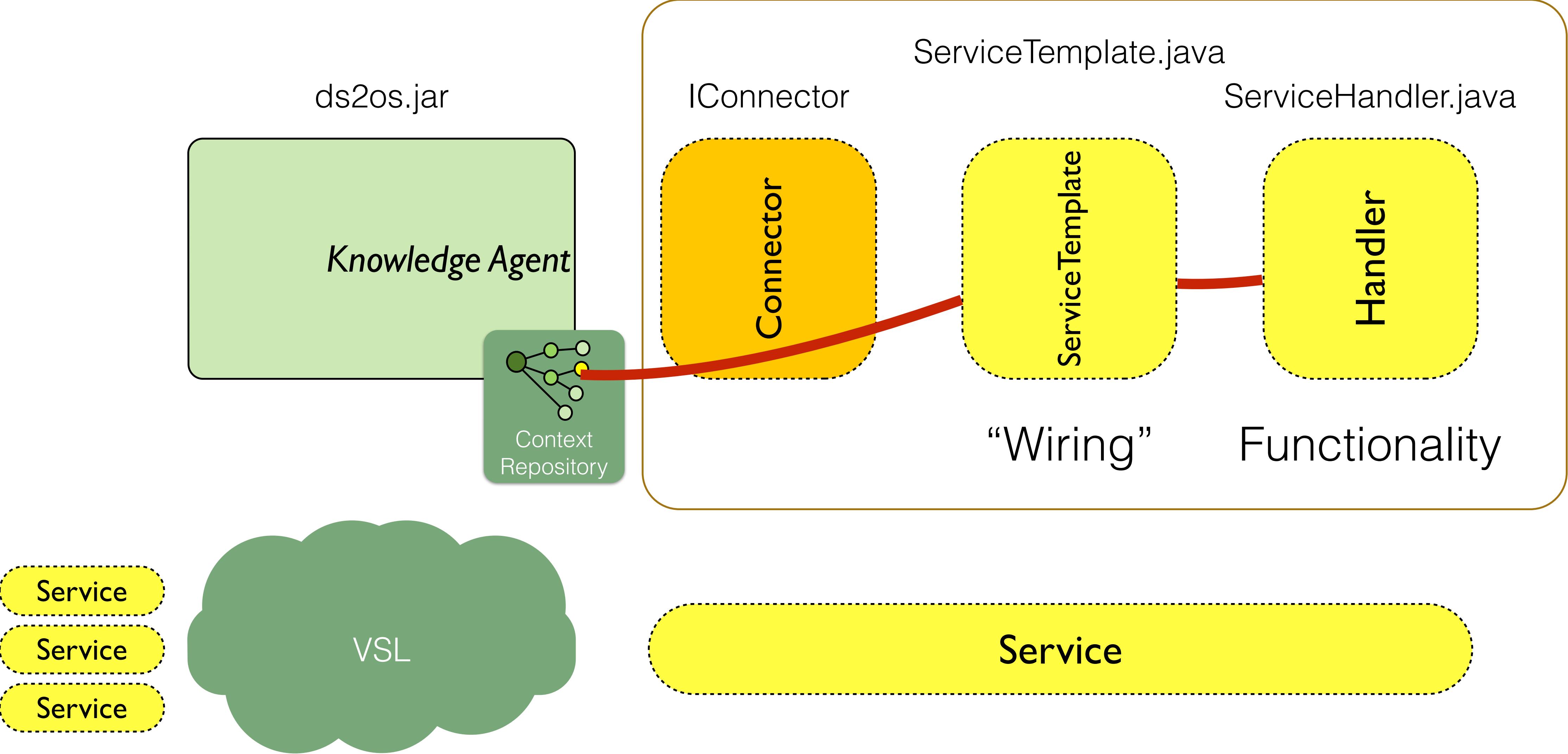


serviceTemplate.tar.gz
PlainOldDataObject

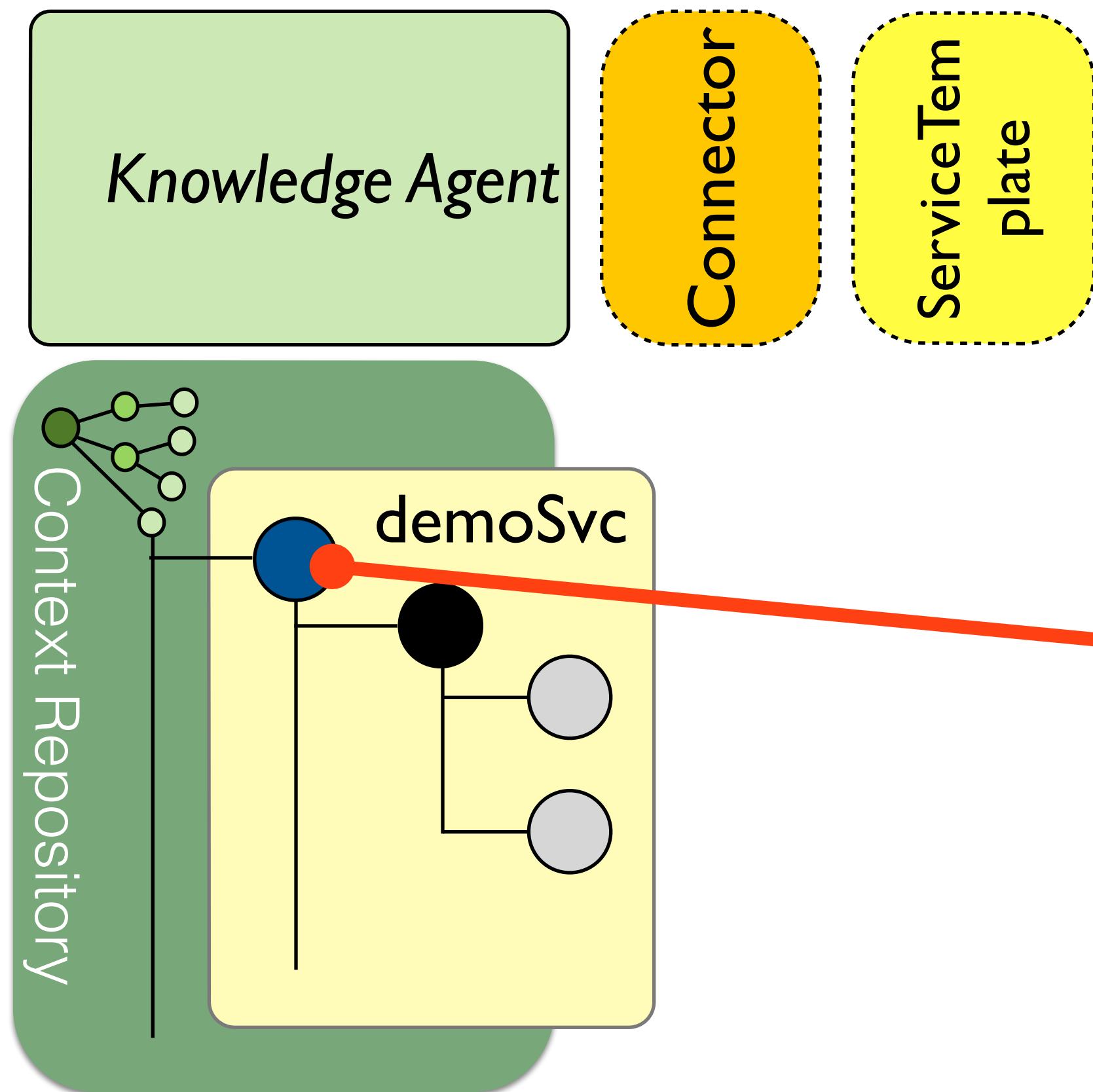
```
java -jar ds2os.jar --console
```



Service Template

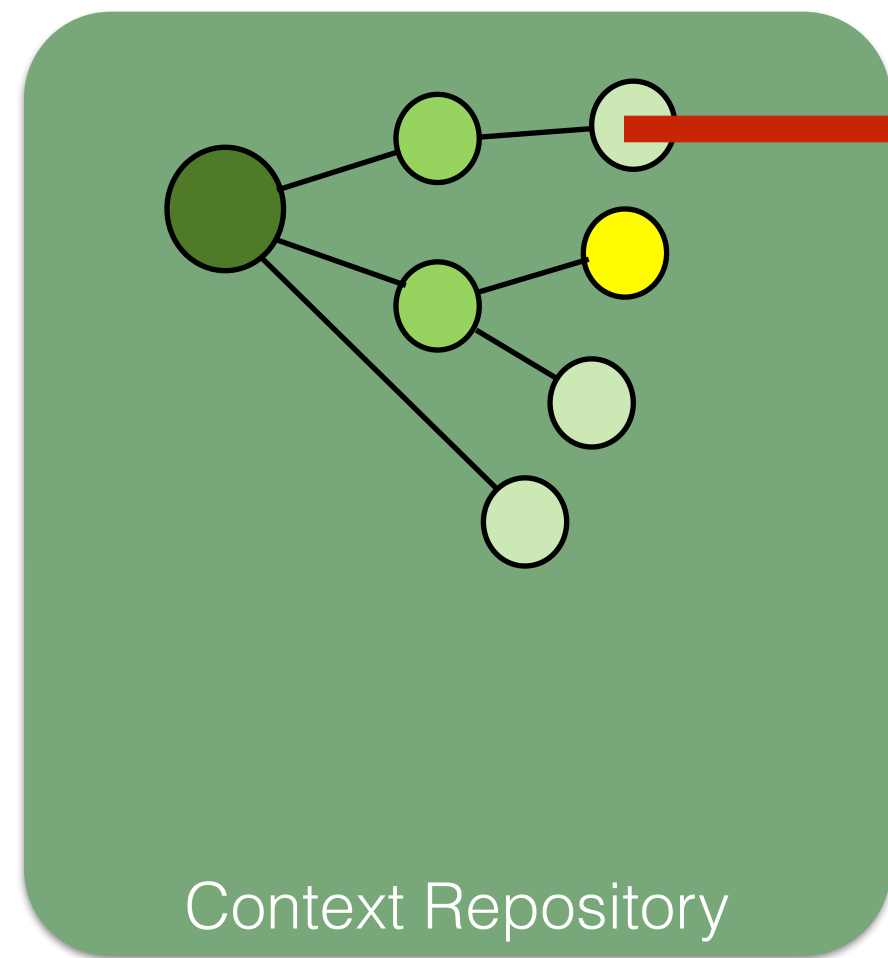


Connector Setup



```
public ServiceTemplate() {  
    try {  
        c = new Connector();  
        c.registerService(myServiceIdentifier, myServiceModelId);  
        myKnowledgeRoot = c.getKORSubtree();  
        h = new ServiceTemplateHandler(c, myKnowledgeRoot);  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```

Subscription

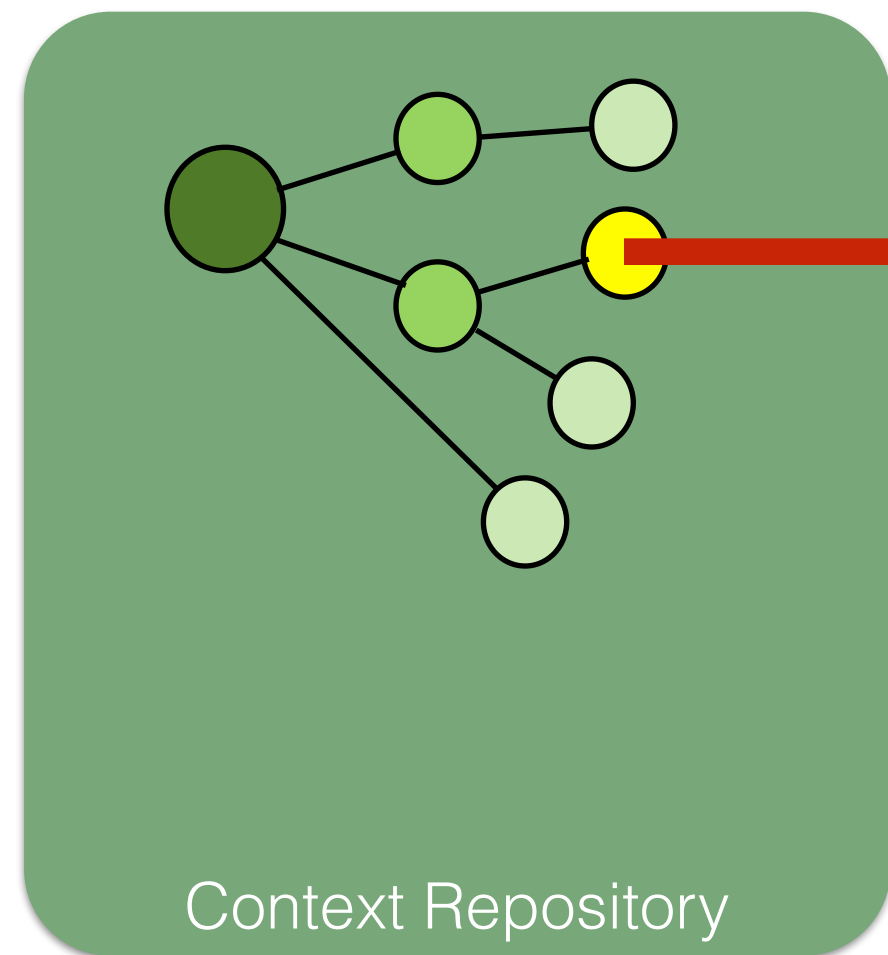


“/x/y/z changed”

```
c.subscribe(myKnowledgeRoot + "/tickle", new ISubscriber() {
    @Override
    public void notificationCallback(String address) {

        // read the node's data
        if (c.get(myKnowledgeRoot + "/tickle").equals("1")) {
            // Someone is tickling us
            h.tickleHandler();
        }
    }
})
```

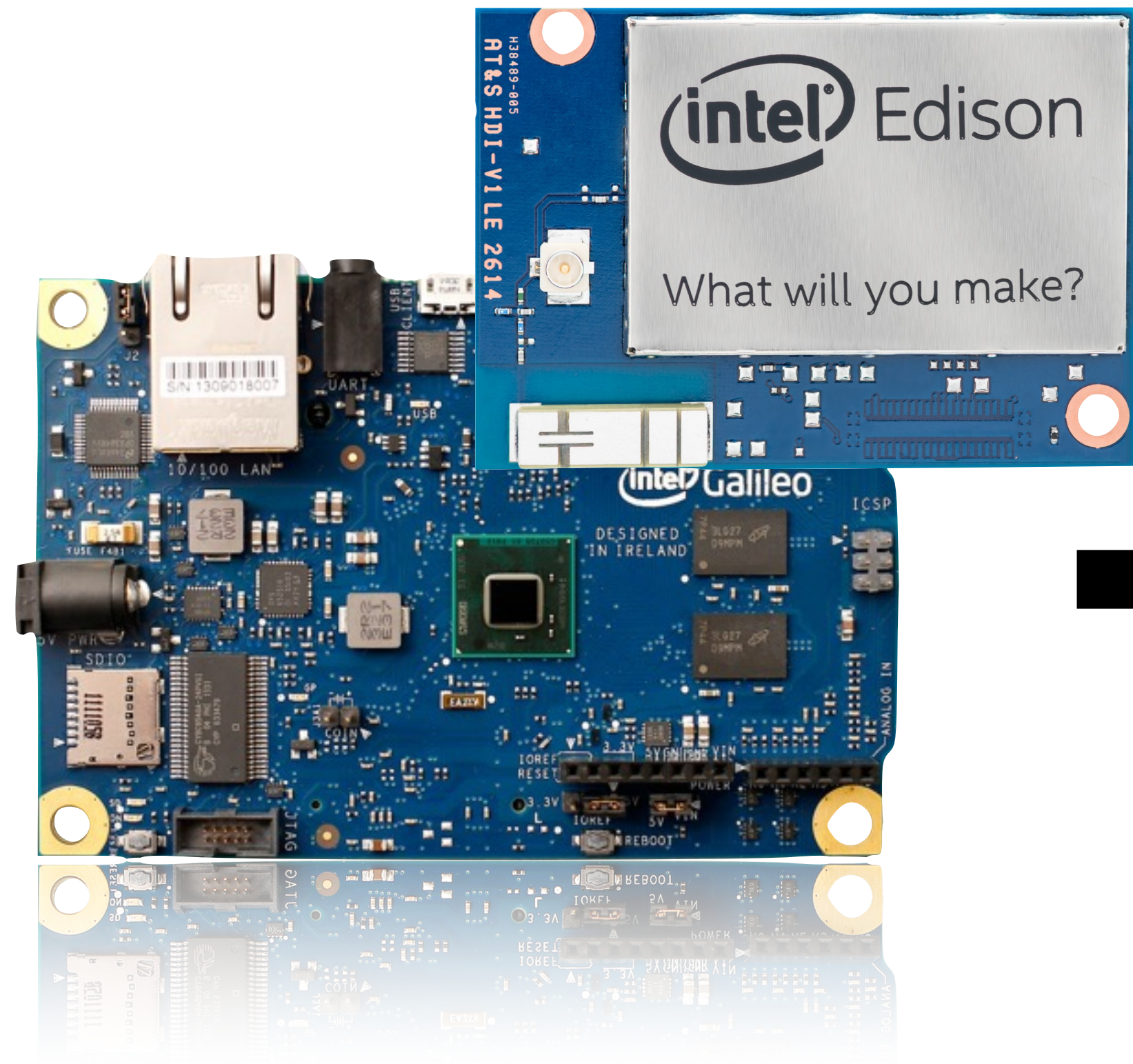

Virtual Node



“{get,set} /x/y/z/fooBar”

```
c.registerVirtualNode(myKnowledgeRoot + "/greet", new IVirtualNodeHandler() {  
    // Greet the posted name to the std::out. Ignore added suffixes.  
    @Override  
    public void set(String address, byte[] value, String writerID) {  
        h.greetConsole(value.toString() + "!");  
    }  
  
    // Greet the suffixed name by returning the greeting.  
    @Override  
    public byte[] get(String address, String readerID) {  
        String suffix = address.substring((myKnowledgeRoot + "/greet/").length());  
        return h.greetBack(suffix).getBytes();  
    }  
});
```

Why is DIY Software Making
Relevant?



+



**= Pervasive Computing
becomes Reality**

Hands-On GAME

Marc-Oliver Pahl, Philipp Härtinger, Felix Kuperjans, Andreas Hubel

