



# Smart IoT on OSGi with Apache OpenWhisk

David Bosschaert and Carsten Ziegeler



*I am doing IoT !*

*I am doing IoT !*



- Principal Scientist @ Adobe Research Switzerland
- Member of the Apache Software Foundation
- VP of Apache Felix and Sling
- OSGi Expert Groups and Board member





- R&D Adobe Ireland
- Co-chair OSGi Enterprise Expert Group
- Apache Felix, Aries PMC member and committer
- ... other opensource projects
- Cloud and embedded computing enthusiast



IoT



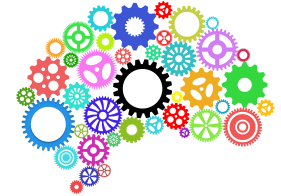
#AdobeRemix  
Hiroyuki Mitsume

# IoT Hello World!



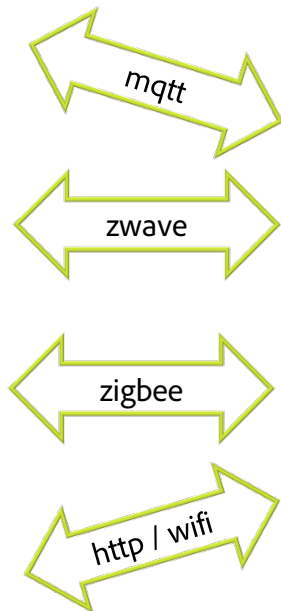
*I am doing IoT ?*

# IoT devices



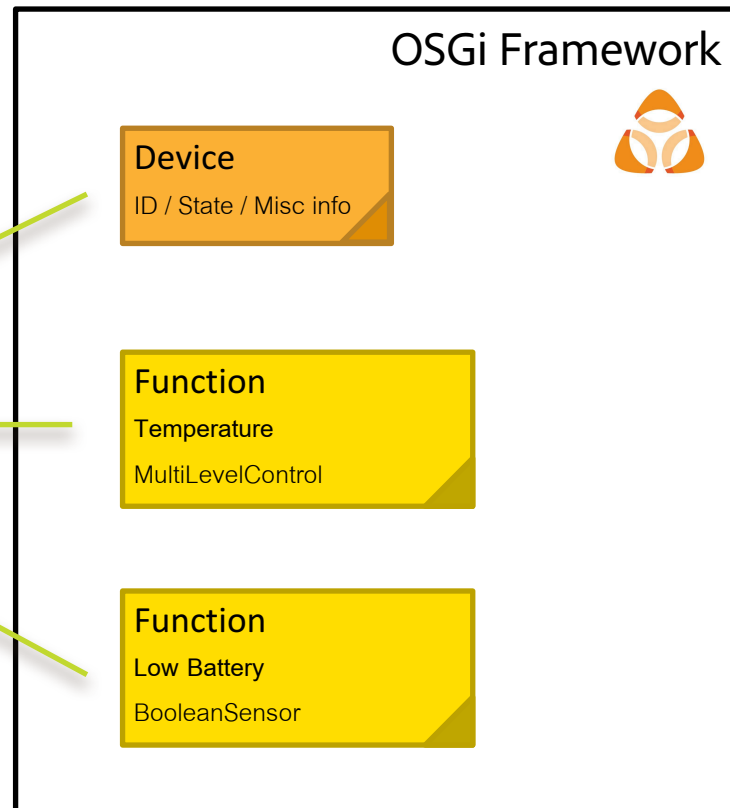


# IoT and OSGi



- Unified platform
- Service Registry integration point
- IoT APIs
- Share & Reuse
- Provisioning, tooling...
- Dynamically updatable

- Device and Function Services
- Protocol independent



## DataTypes

Alarm

BooleanControl

BooleanSensor

Keypad

Meter

MultiLevelControl

MultiLevelSensor

WakeUp

... custom types ...

## Function Types

COLD

CONTACT

DOOR

FIRE

FLOW

GAS

HEAT

HUMIDITY

LIGHT

LIQUID

MOTION

NOISINESS

OCCUPANCY

POWER

PRESSURE

RAIN

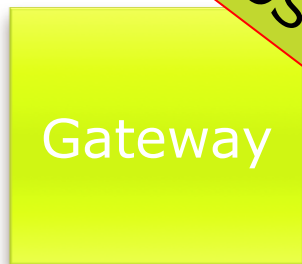
SMOKE

TEMPERATURE

WATER

WINDOW

# OSGi DAL demo



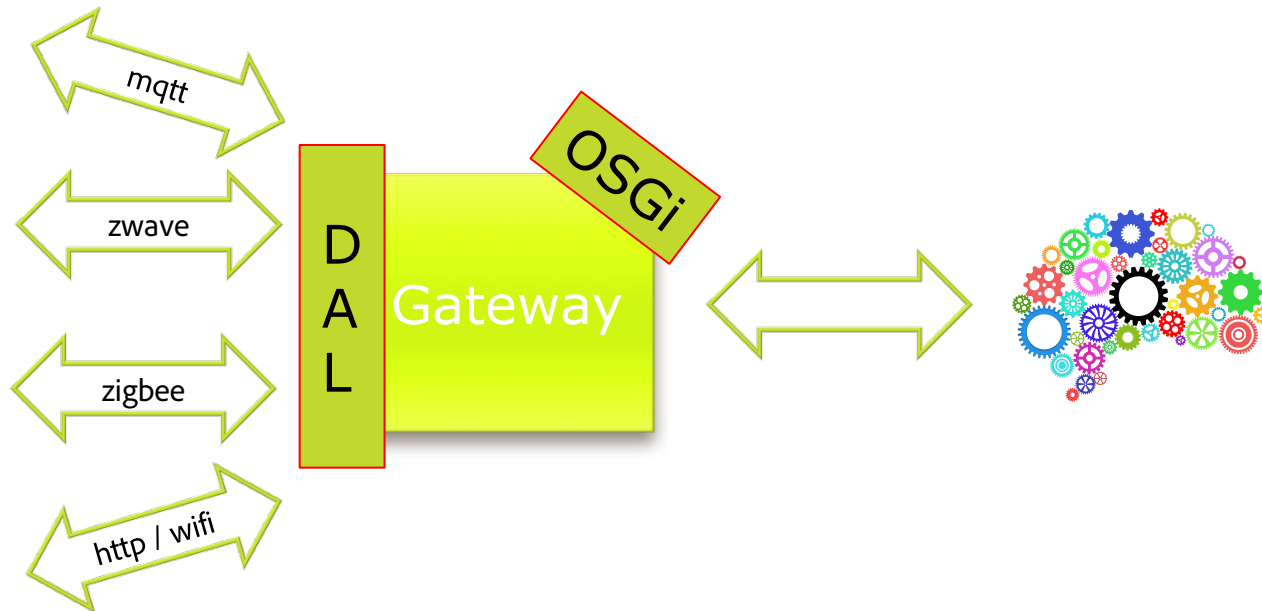


## DAL allows for purely functional code

```
@Component(  
    property = {  
        "event.topics=org/osgi/service/dal/FunctionEvent/PROPERTY_CHANGED",  
        "event.filter=(dal.function.UID=pir1:motion)"}  
public class RadiatorLogic implements EventHandler {  
  
    @Reference(target = "(dal.function.UID=rad1:setpoint)")  
    Function radiator;  
  
    @Override  
    public void handleEvent(Event event) {  
        MultiLevelControl radCtrl = getControl(radiator);  
  
        // Get the updated PIR sensor data  
        BooleanData data = (BooleanData) event.getProperty(FunctionEvent.PROPERTY_VALUE);  
  
        // Set the radiator temperature  
        radCtrl.setData(data.getValue() ? 21 : 7, "degrees");  
    }  
}
```



# Let's make it smarter!



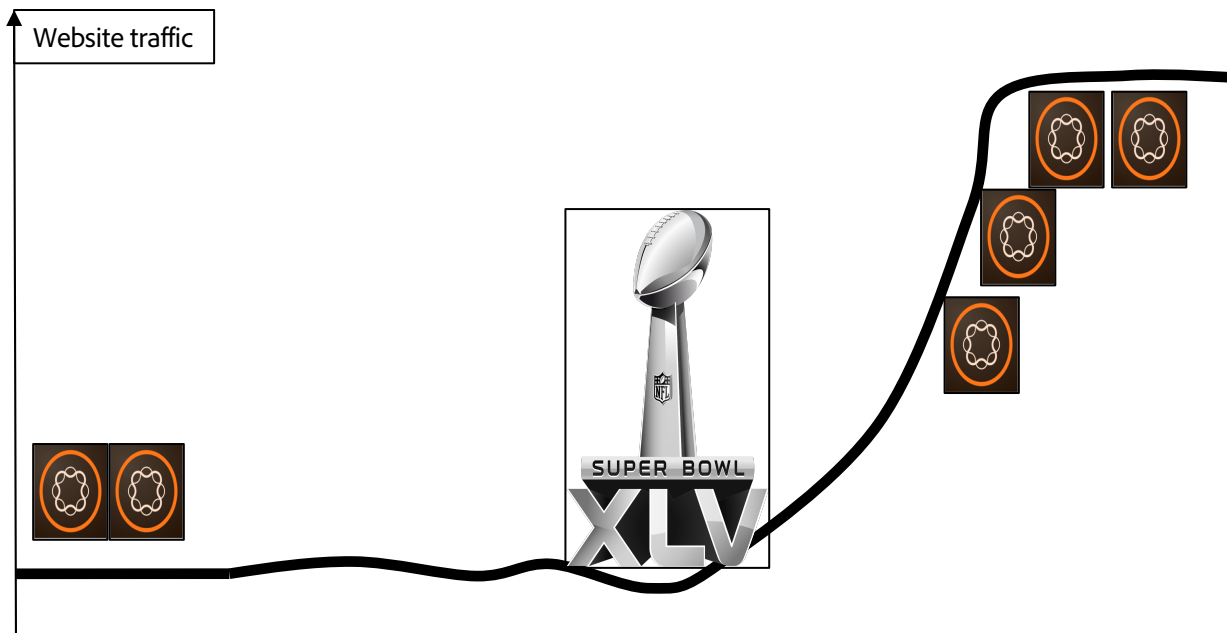


Less Server?



#AdobeRemix  
Hiroyuki Mitsume

# Surviving the Super Bowl





# A Short History of Infrastructure/Deployment



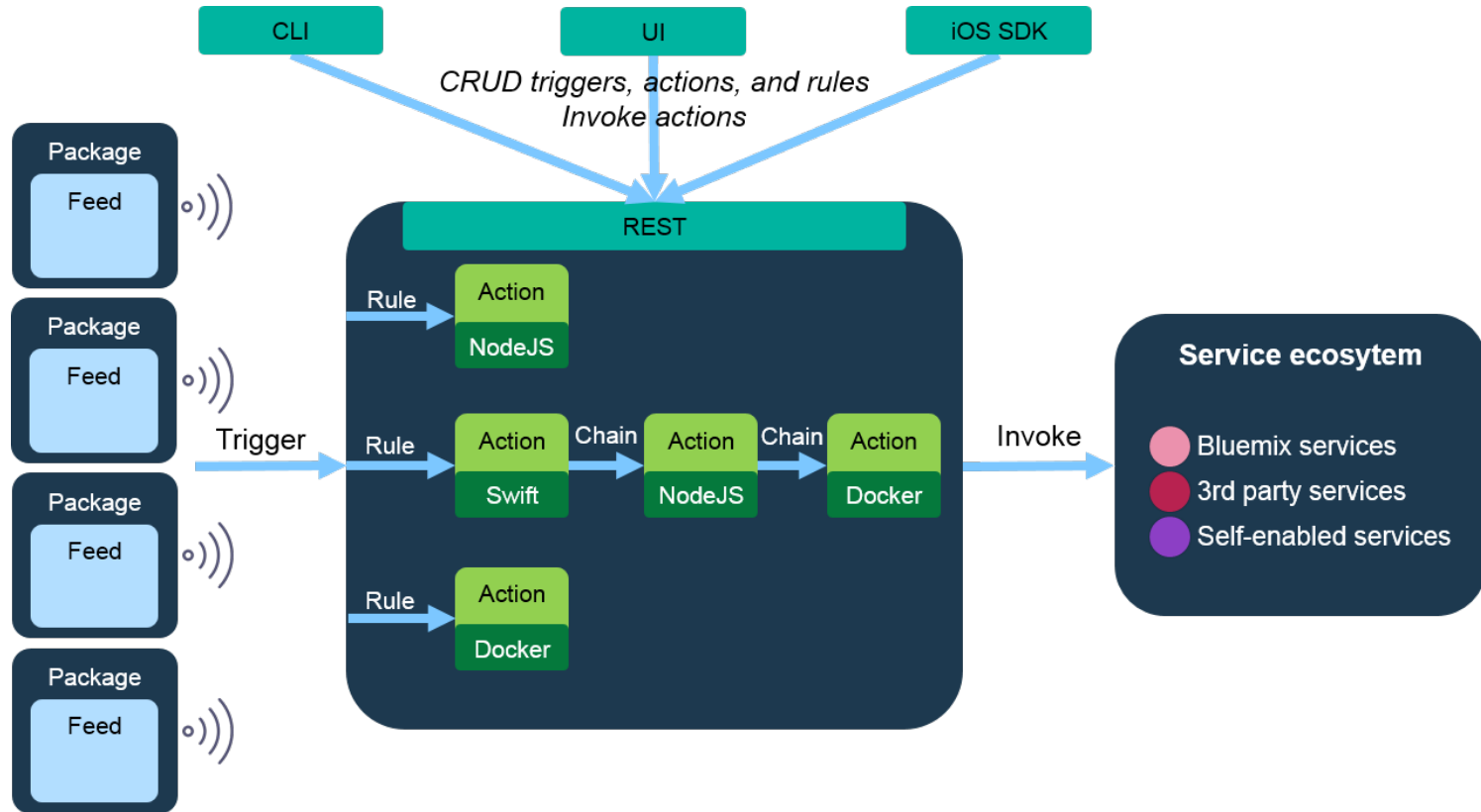
Microsoft Azure



Microsoft Azure



- Function as a Service (FaaS)
- Event triggering
- Scale as needed

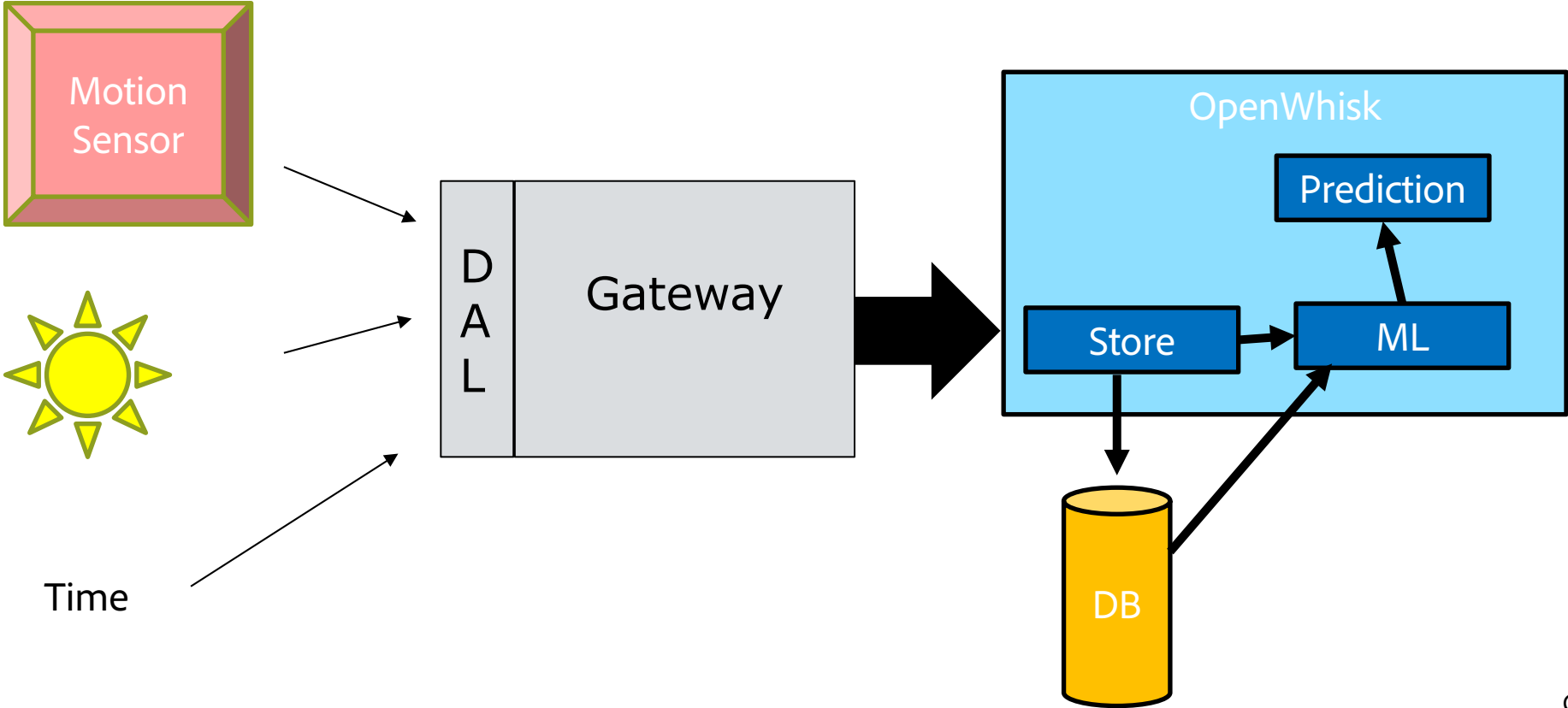


Source: <https://github.com/apache/incubator-openwhisk/blob/master/docs/about.md>

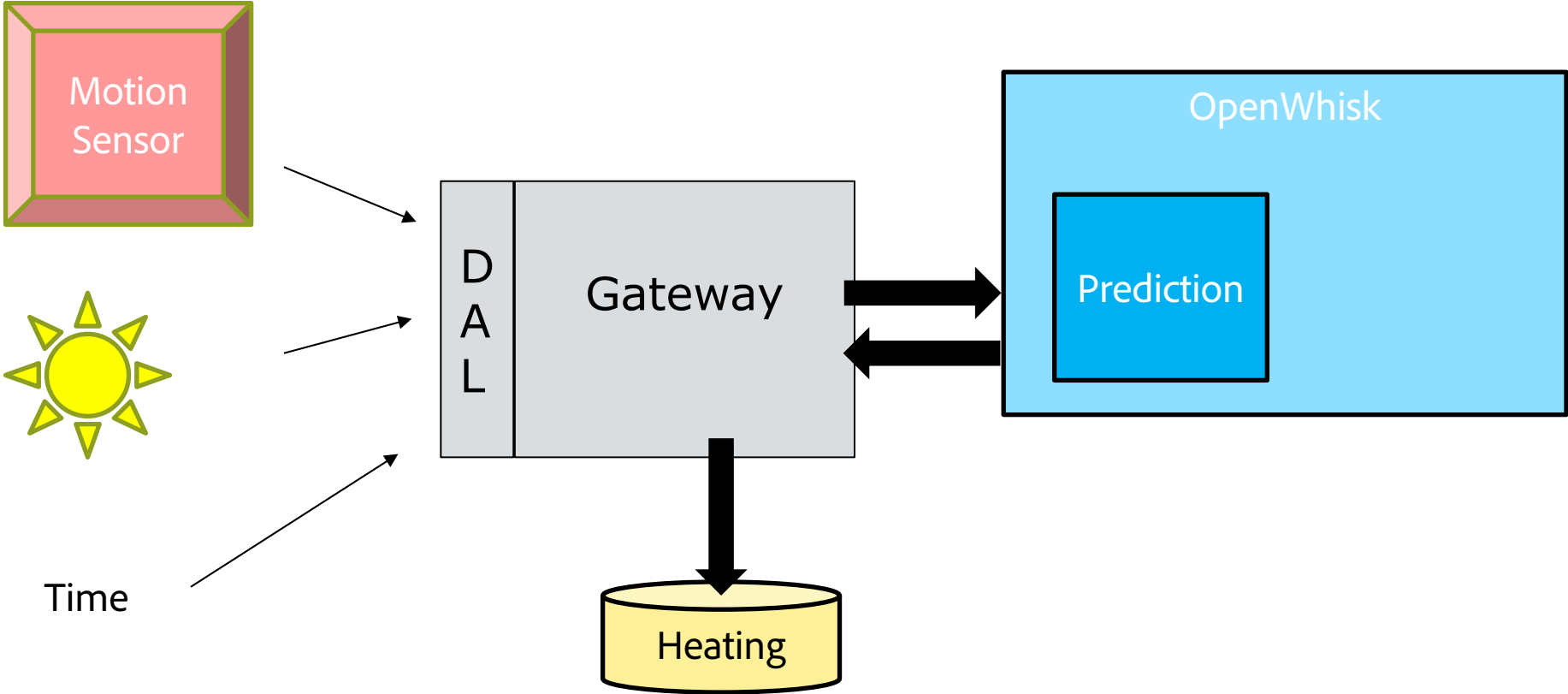


## Smart Home Logic

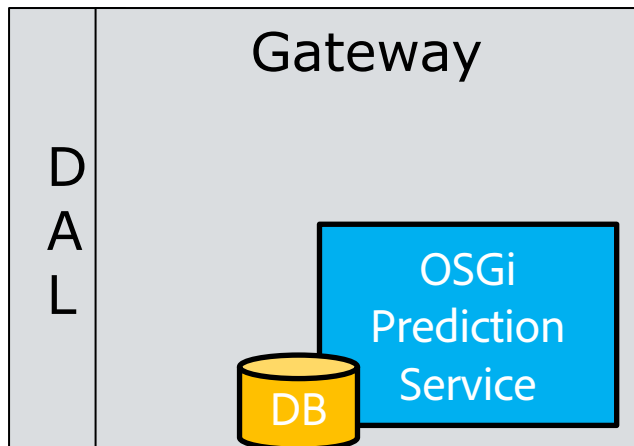




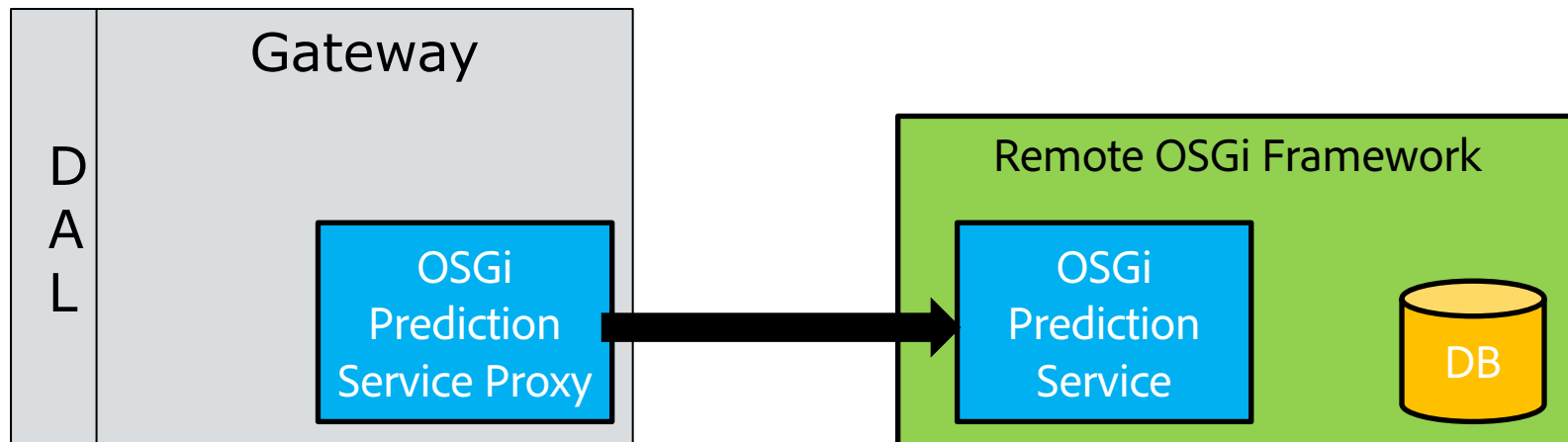




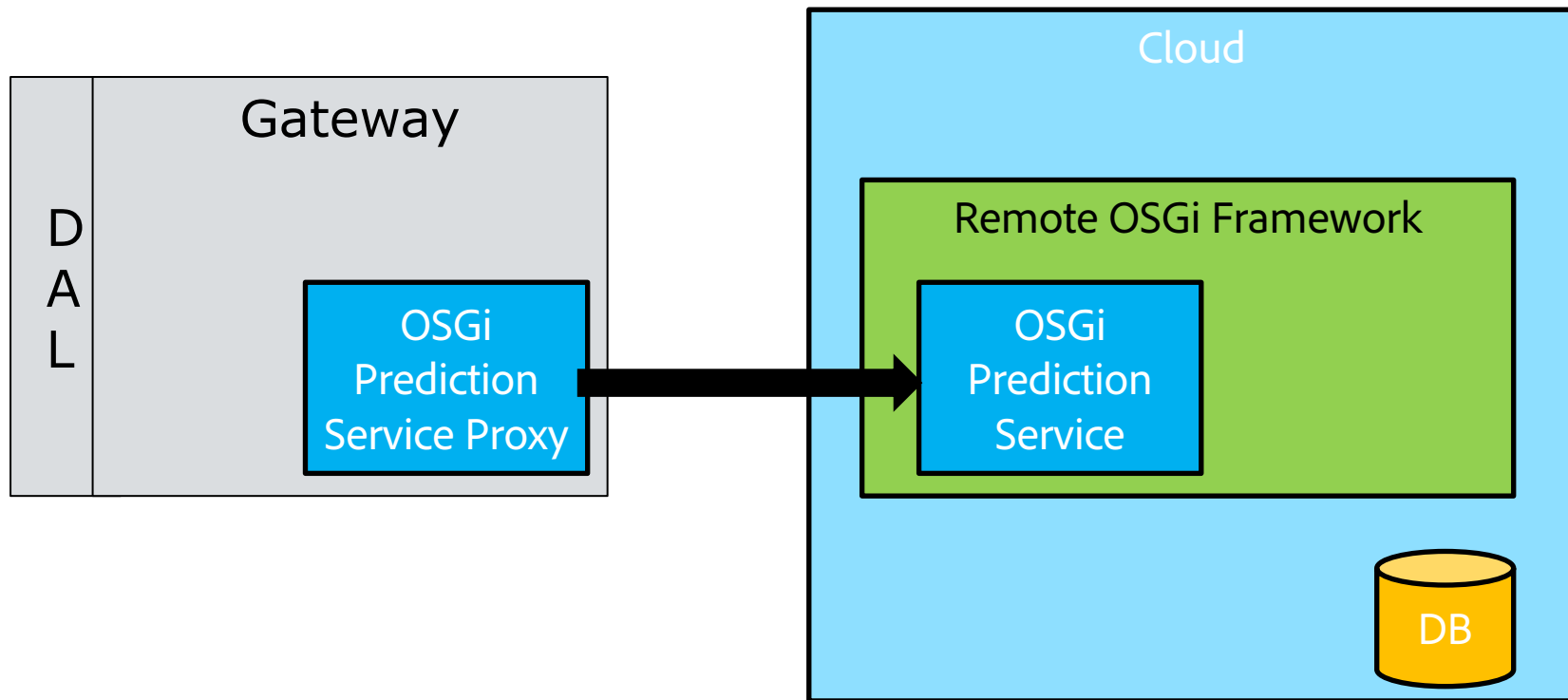
# Seamless Integration



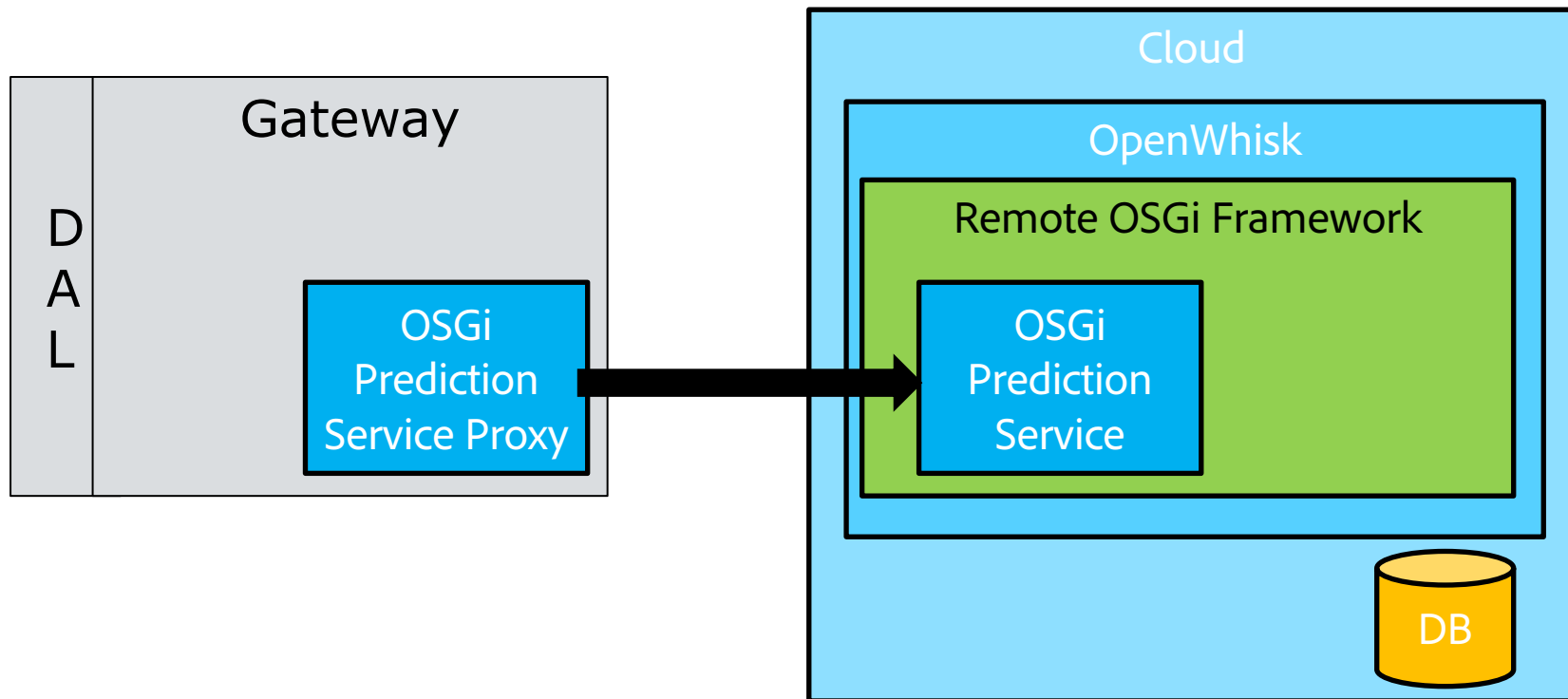
# Seamless Integration



# Seamless Integration



# Seamless Integration





# OSGi + OpenWhisk: the ideal base for IoT



THANK YOU.