Ai.RIM – Resource and Inventory Management
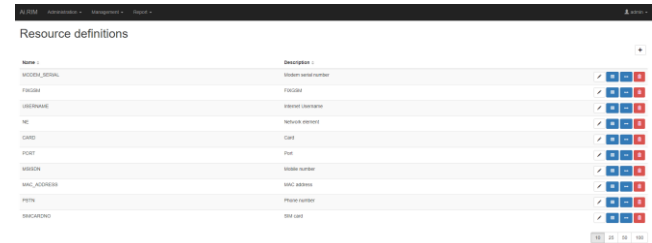
VERSO
ALTIMA

Contents

ERSO
ALTIMA

What is Ai.RIM?

- **Platform for management of physical and logical resources**

- **Resource is generic; can be anything that requires management of state and ownership**

    - **Typical resources: SIM cards, mobile and fix phone numbers, modems, smart cards, usernames, IP addresses…**

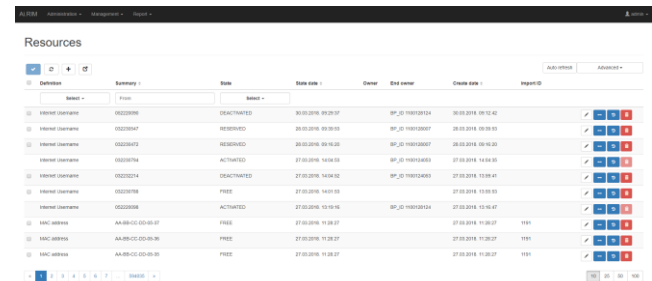- **Exposes web service interfaces (SOAP and REST) towards other systems (such as CRM)**

CRM → HTTP → **Ai.RIM**

VERSO
ALTIMA
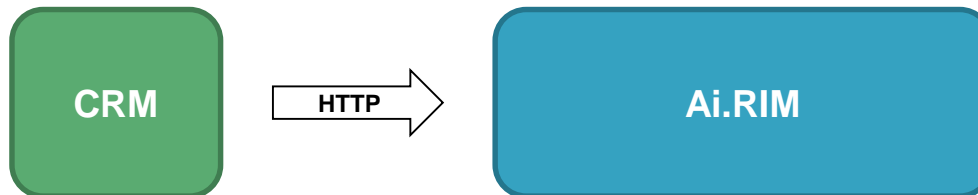
Resource Definitions

- **Resource Definition defines type of resource**

- **Each Resource Definitions can be assigned multiple parameters that further describe that resource**

- **Once resource is defined, an instance of Resource Definition is a concrete resource**

    - **For example, for a Resource Definition called MODEM, we define parameter MAC_ADDRESS. A concrete instance of such resource is a modem having MAC address aa:bb:cc:dd:ee:ff.**

ALTIMA

Resource Relations

- **On Resource Definitions, we may define allowed relationships between Resource Definitions**

- **Like resources, relations can be logical or physical**

  - **For example, we may define a pairing relation between a SIM card and an MSISDN, or between a phone number and a port on a switch**

- **Resource relations may also define how different components of a managed resource are connected (a switch may have a slot, which may have a port, etc.)**

Resource Queries

- **Resource Queries are executable Groovy scripts used to extend Ai.RIM**

- **When Ai.RIM is invoked by external system, a resource query is executed**

  - **It may perform actions such as returning a list of resources, reserving a resource for a customer, changing relations between resources, etc.**

Edit resource query

Name

GET_PORT_CONNECTED

Script

```
    for (ResourceDto resource : resources) {
        threadLog.log(String.format("Resource %s", resource));

        portResource = resourceApi.findFirstResource(ResourceCondition.whereResourceIs("PORT")
            .andParentResourceExists('CONNECTED', resource.getId()));

        threadLog.log(String.format("Connected port %s", portResource));

        if (portResource != null) {
            response.addResource(portResource);
        }
        return response;
    }

    if (portResource != null) {
        return response;
    }
    }
}
```

Save   Cancel

Bulk Actions

- **Several actions can be performed in bulk:**
  - **Massive imports of resources through CSV files**
  - **Bulk querying of multiple resources by various filters**
  - **Bulk reservation, freeing, or updating resources**

VERSO
ALTIMA

Architecture

- **Running on Apache Tomcat, with a web GUI**

- **May be running on multiple nodes, where each node is stateless**

  - **This allows horizontal scalability with little effort – there is no limit on the number of nodes**

- **Load balancing and failover is done through hardware load balancers or software load balancers (nginx)**

- **Database abstraction layer allows supporting multiple relational database vendors (running on Oracle by default)**

HTTP → LB → HTTP → **Ai.RIM #1**
              → HTTP → **Ai.RIM #2**
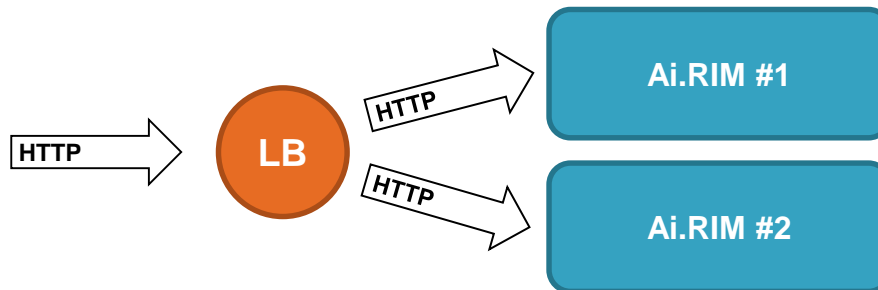
VERSO
ALTIMA

Thank you for your attention!

VERSO
ALTIMA