Chapter 2

Problem Definition

In an intermittent connection environment, sometimes we need to make decisions based on the information we currently have. However, the information we base our decisions on change as soon as we reestablish the connection. Therefore, the decision we made before might need to be reevaluated. The goal of this research is to propose an architecture in the REST environment to keep track of the dependency between resources (information and resources), as well as alter the depended resources when a new state of a resource is discovered. This architecture will also run in the background as a service to support different platforms. To achieve the goal of this research, we need to answer the following questions:

- How to track the dependency between resources?
- How to inform the system when a resource state has changed?
- How to propagate resource state change to other resources?
- How to identify same resource?
- How to inform the resource in other devices when a resource it is dependent on has a new state?
- How to update a resource when not all its depended resources are reachable?

2.1 How to track the dependency between resources?

Keeping track of the dependency between resources is the first step we need to do to make our system work. Thus we need to find a way to let the system know which resource is dependent on which resource.

2.2 How to inform the system when a resource state had changed?

When a resource state had changed, how does the system know about it? A simple solution is to let the system keep checking for updates. However, as the number of resources increases, the cost of the update checking becomes more and more expansive. Is there a better way to solve this?

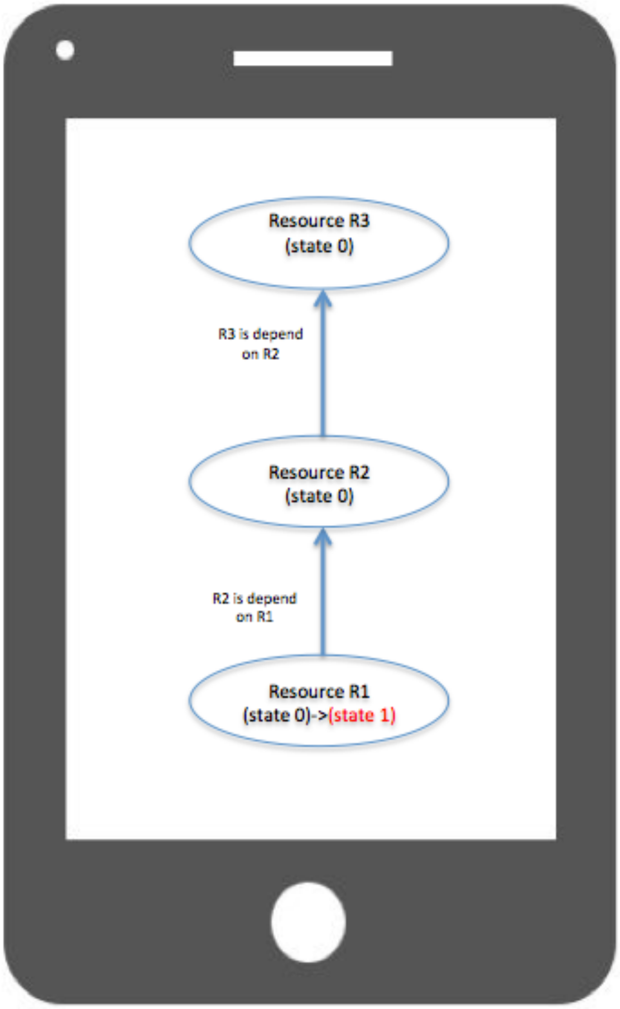

Figure 2.1 Dependent Resources

2.3 How to propagate resource state change to other resources?

After receiving a state change from a resource, how does the system tell the resources dependent on it to check their states and see if they need an update? In figure 2.1, Resource R1 told the system that its state just changes from state 0 to state 1. The system has to do something to let Resource R2, which is dependent on R1, know that the R1 state has changed and it might need to update its state as well. The system also needs to inform Resource R3, which is dependent on R2, that R2 is probably not up to date, so its state is probably out of date as well.

2.4 How to identify same resource?

Identifying resources is critical in this research; especially between devices. We need to find a way to let the system identify the same resource used in different device, so when devices exchange resource information, they know exactly which resource they are talking about.

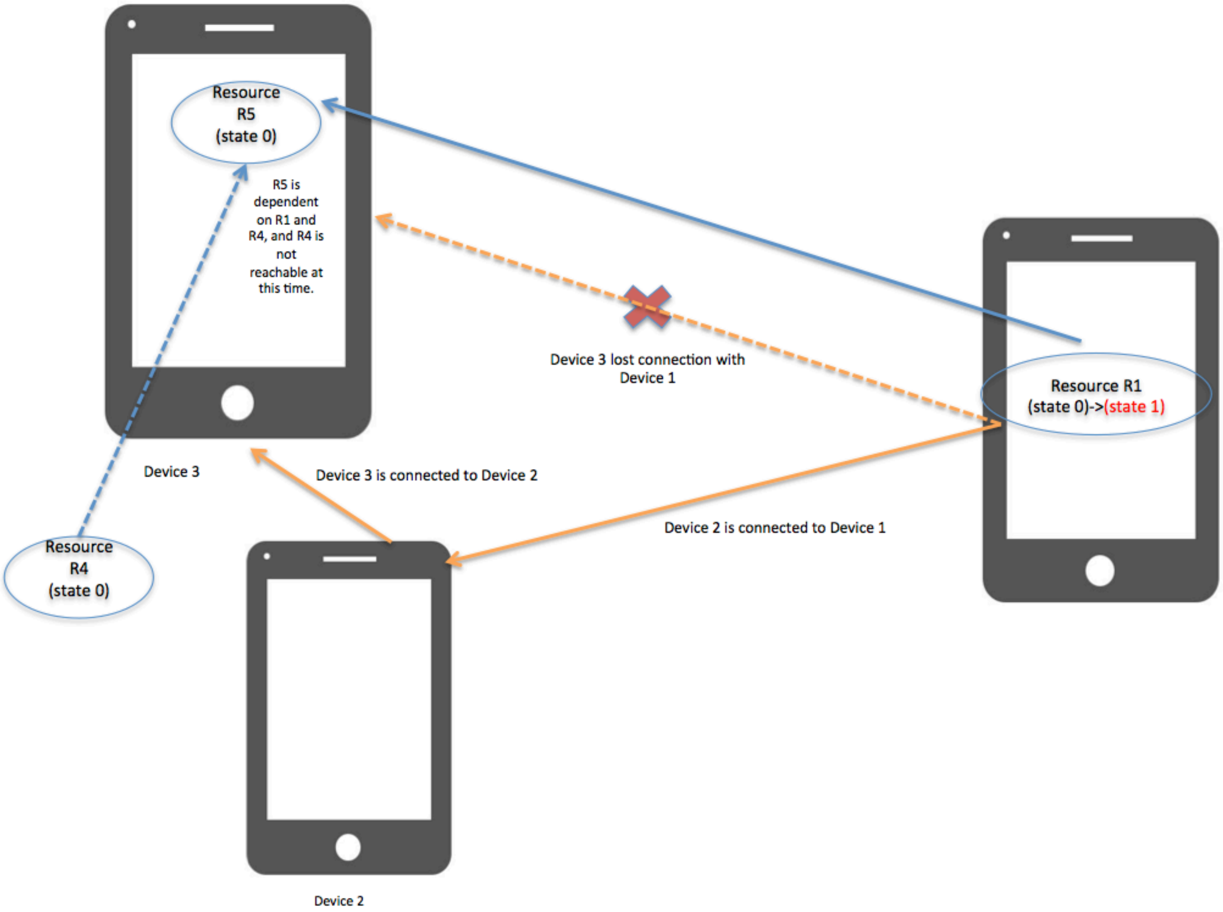
Resource
R5
(state 0)
R5 is dependent on R1 and R4, and R4 is not reachable at this time.
Device 3 lost connection with Device 1
Resource R1
(state 0)->(state 1)
Device 3
Device 3 is connected to Device 2
Device 2 is connected to Device 1
Resource
R4
(state 0)
Device 2

Figure 2.2: Resources between devices

2.5 How to inform the resource in other devices when a resource it is dependent on has a new state?

In figure 2.2, how to let resource R5 in device 3 knows that resource R1, which is one of the resources R5 is depended on, has a new state? This requires device 1 to send a signal to device 2 and then device 2 sent a signal to device 3. However, how does the system know that it can use device 2 as a medium to pass the update information?

2.6 How to update a resource when not all its dependent resources are reachable?

In figure 2.2, when the system informs resource R5 to check if it needs to change its state due to a fact that resource R1, which it depends on, has a new state, resource R5 might need to access both resource R1 and resource R4. However, R4 is not reachable for R5 because there is no connection. Therefore R5 might not be able to process its new state due to the fact that one of the resources it depends on, R4, is not available at this time.