

Architecture and Open Queue Model

Thesis

Silvio Dei Giudici, Marco Morella, Mattia Nicoella

26 September 2020

1 Introduction

This section will deal with the assumptions we have made so far and why we've made them. Alternatives will be described for the various possibilities.

1.1 Assumptions

1. **Assumption 1:** The only entity needing a storage system is the Central Node, in particular we are assuming that all other nodes receive information that can be processed, stored and aggregated in the RAM which will never be full.
2. **Assumption 2:** more of a design choice for the moment, each node is modeled as a simple queue composed of CPU+RAM together, in future they will be extended and may need more component and a review of the entire queue model.
3. **Assumption 3:** In this first stage we won't be holding account of the fault tolerance of the system, in particular we are not considering duplication of the data not in storing neither in multiple link sending.

2 System functioning and message flow

The system is composed by:

- Central Node: in our model we have one central. It manages all the regional nodes. The central aggregates the telemetry messages received by the regionals before saving them in the disk and save all the transitions messages that it receives. Also the central may generate a command to the actuators in response to a transition message that it receives, but not all the times. All transition-command couples (i.e. what we called the batch data as we'll show in the next sections, that are basically a log of each command sent by central, regional and local nodes to the actuators) arrive here to be logged in the disk.
- Regional Node: each regional node manages a subset of all local nodes, disjoint from any other regional node domain. Telemetries will flow through here to be aggregated and sent to the central node. Also transitions flow through here and may generate a command to the actuators in the area controlled by the regional. Every batch data are forwarded to the central node.
- Local Node: each local node manages a subset of all actuators and sensors, disjoint from any other local node. The telemetry messages are aggregated and sent to the regional assign to the local node. Also the transitions received by the local are forwarded to its regional, but also they may generate a command for its actuators. As before, the batch data are forwarded to its regional to reach the central.
- Actuator: commands will reach the actuators for a state change in the system, they can also send transitions to the higher levels when their state is changed without receiving a command message .
- Sensor: periodically will generate telemetry messages and send them to the local level to be forwarded up to the central node, also will generate and send transition messages when their state is changed. They only send, never receive.

Both local and regional nodes need an acknowledgement reply for every telemetry message from the higher level, respectively regional and central nodes.

All aggregations we talked about are determined by parameters that will be given.

There is no duplication in sent messages.

All message types will be thoroughly analyzed in the Classes section.

3 Structural scheme

This section will explain how the structural scheme for the system is made and its project's choices.

In the model we are going to show, we used as connection between the upper level node and the lower level ones a WAN connecting the upper level router to all lower level ones.

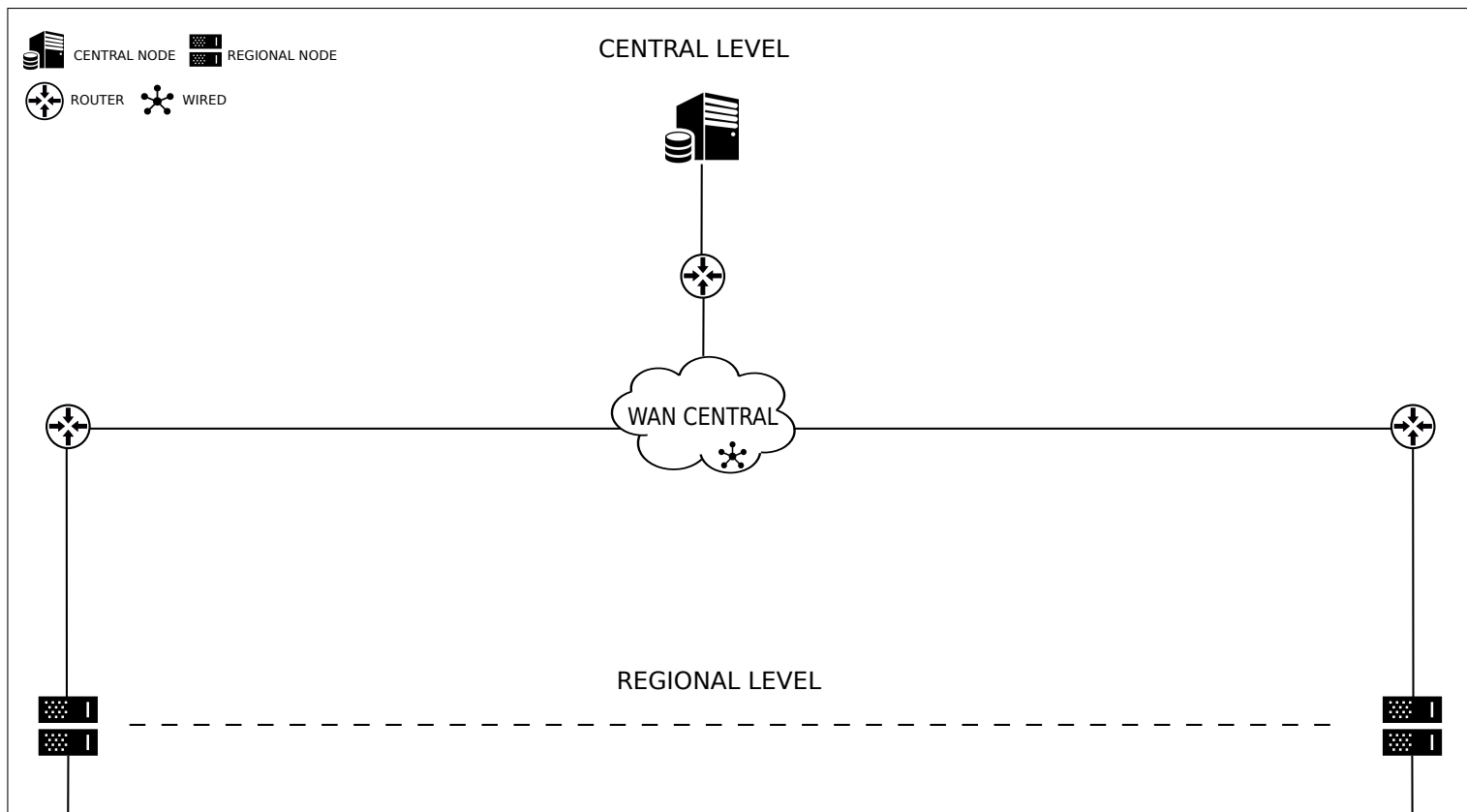


Figure 1: Focus on Central and Regional nodes

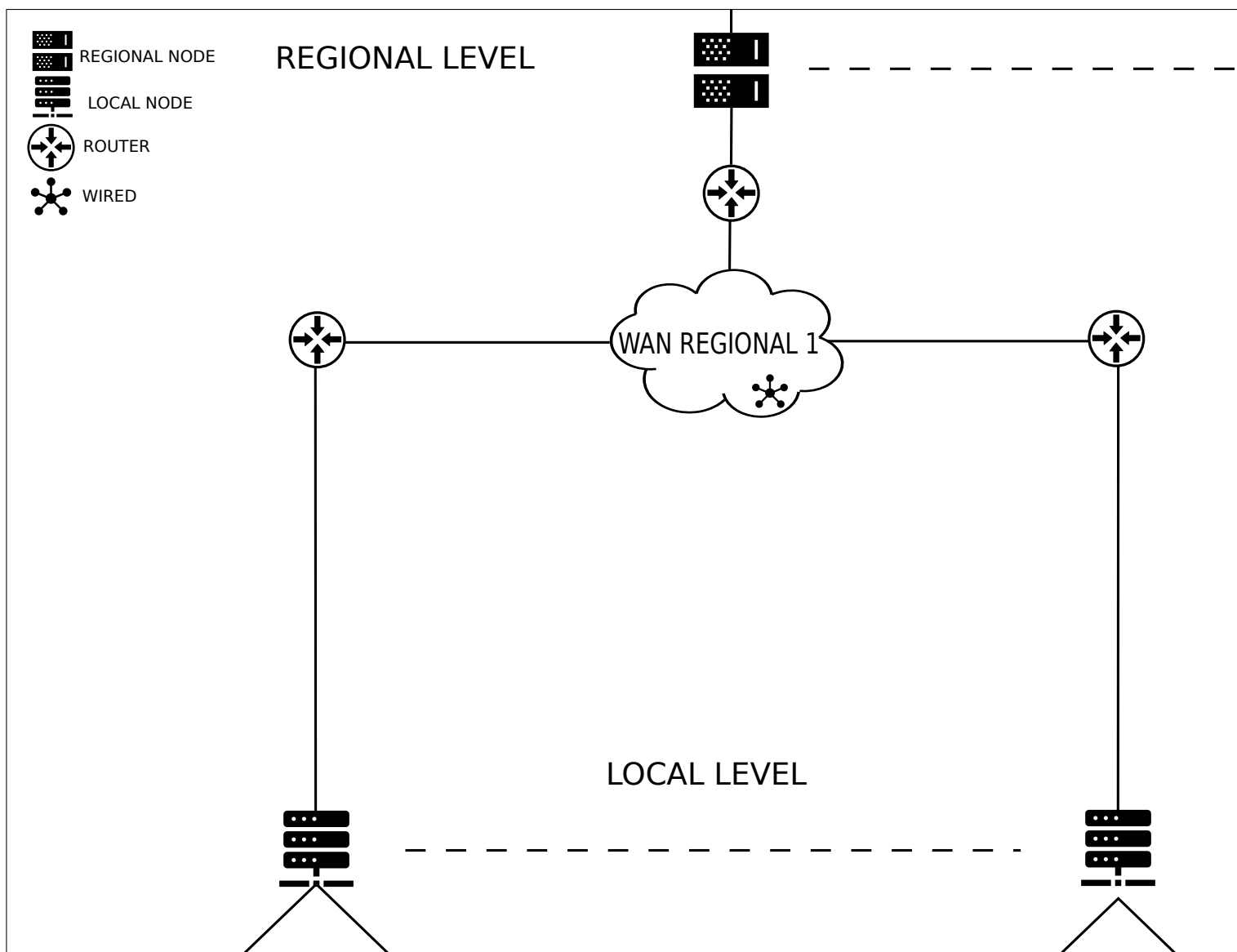


Figure 2: Focus on Regional and Local nodes

Figure 1 represents the connection between Central and Regional nodes, while Figure 2 shows the connection between Regional and Local. We can see that lower level router is used as an interface for lower level nodes to receive and send messages to the upper level WAN which then forwards to the node.

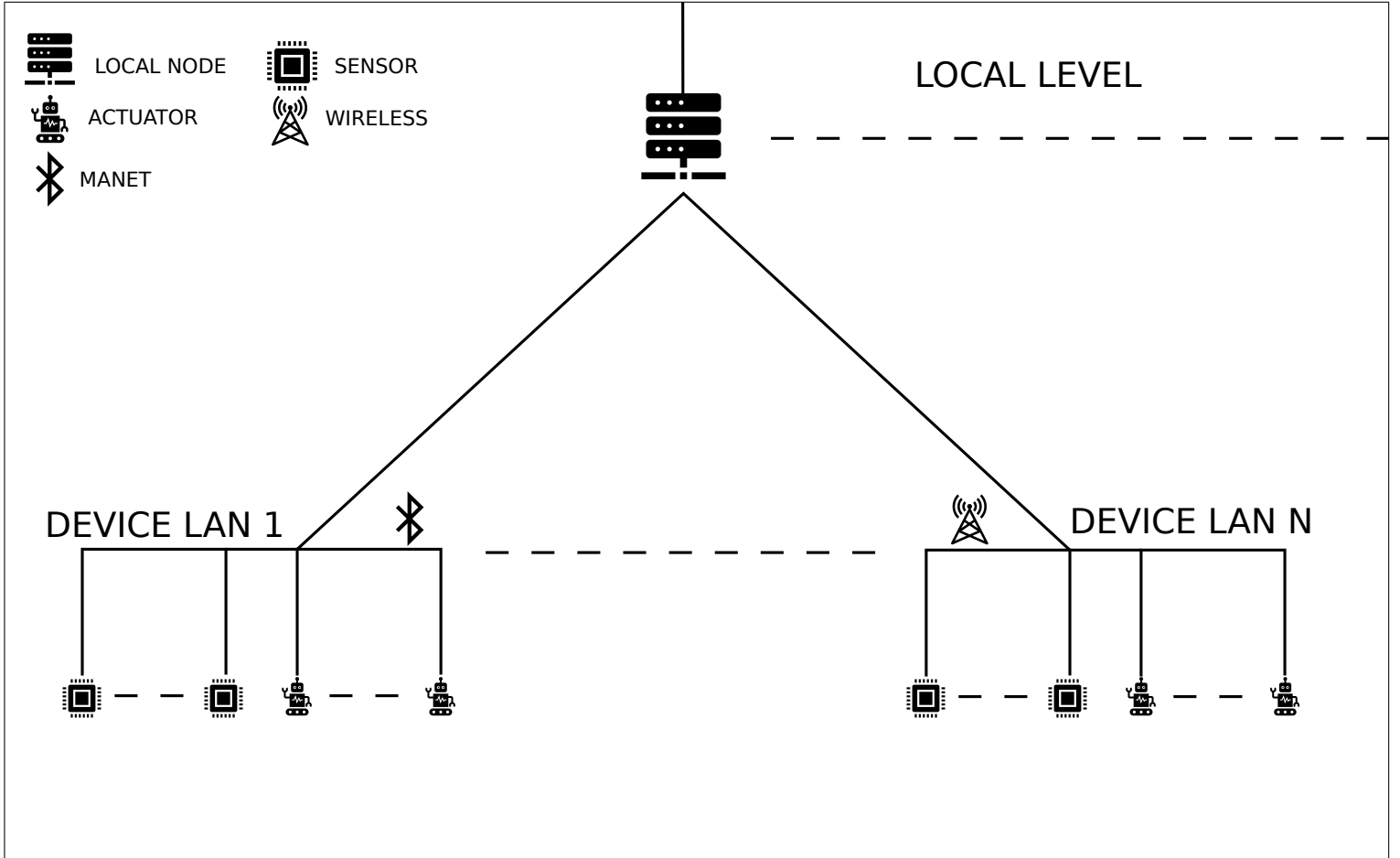
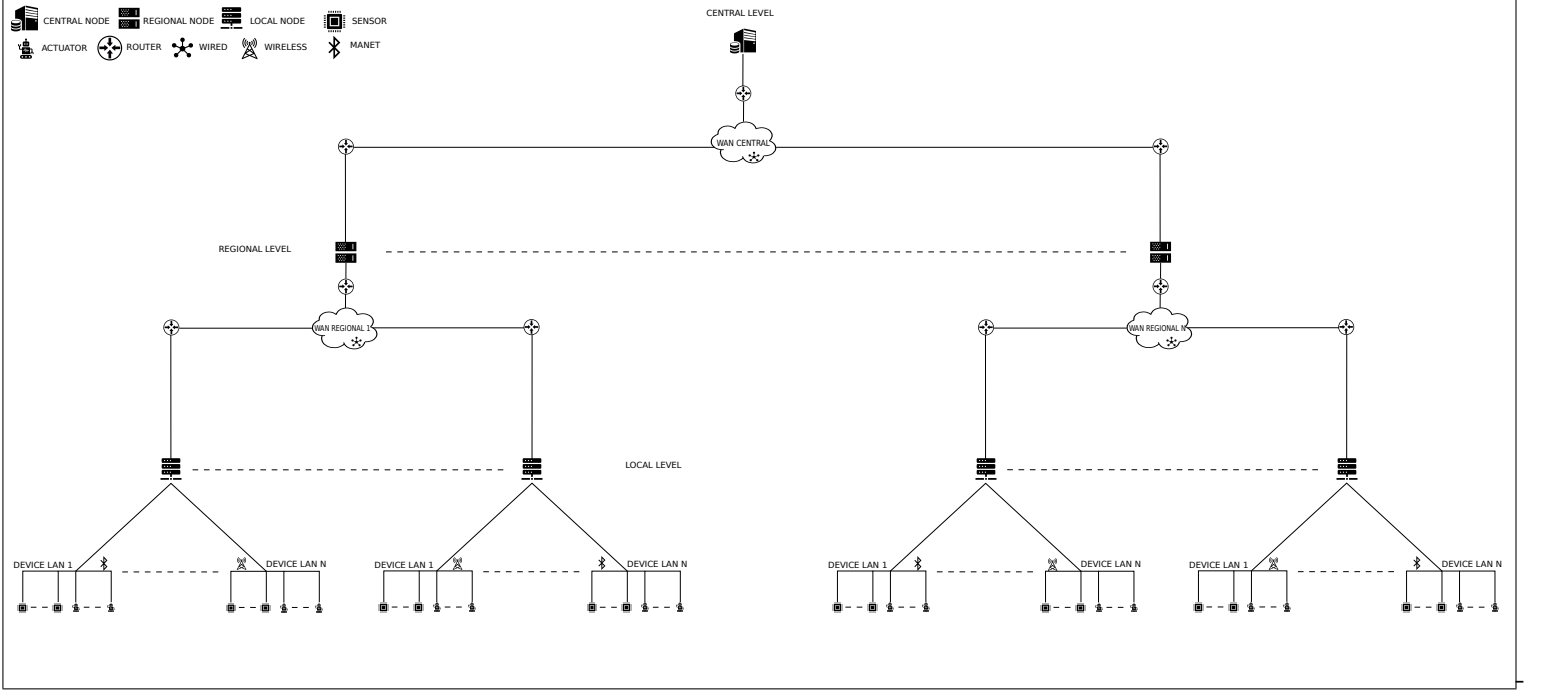


Figure 3: Focus on Local nodes and end points

Instead in figure 3, which displays the connection between Local nodes and the end points of the edge system, namely actuators and sensors, one LAN is used for each type of connection technology, all spanning between end points and their respective upper level node.

The whole scheme:



shell-escape

Figure 4: Structural scheme

4 Queues Scheme and Formulas

The following open queue scheme will be divided in segments during the analysis of each queue. To make the notation less heavy in the graphical scheme we will use some variables which will be reused with a different value for all the subtrees of a given level:

Local LAN subtree :

- T_a, T_s : number of actuator and sensor types, respectively, used in the LAN.
- a_i, s_i : number of actuators and sensors, respectively, of a type i .

Local node subtree :

- L : Number of different local LANs connected to the local node.

Regional node subtree :

- N : Number of local nodes connected to the regional nodes.

It's fundamental to note that the variables described are valid only for the queue network scheme, during the queue analysis we will use the ones defined in the Notation section.

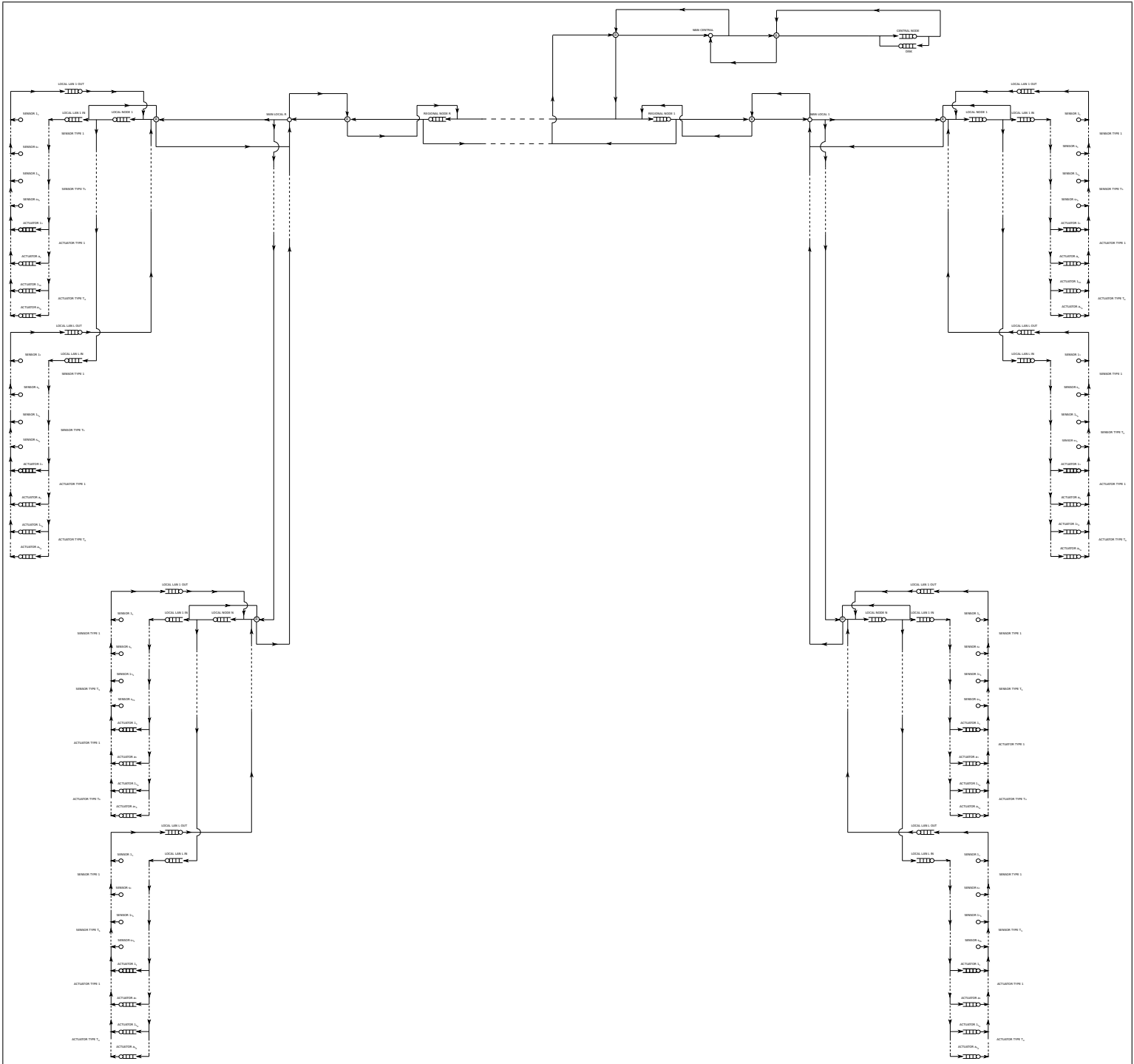


Figure 5: The open queue scheme

4.1 Classes

Based on the specifics and the architecture, we opted for a multi-class open queueing model, in particular at the moment we have four different classes going through the system:

- t: telemetry events, where a sensors sends to its superior the data it has periodically, this data then will be aggregated at each level up until the central node.
- e: transition events, which are sent by actuators or sensors when their state or the physical network changes.
- c: command events, generated by local, regional or central nodes, in response to a transition message which reached it from an actuator or sensor.
- b: batch send, generated by the central, regional and local nodes periodically to send to the central's disk data about the commands that were generated in the network.

Another type of message flowing through the network is a reply message but since they are triggered only by transition events that are sent from a node to another node; we model them with the average number of visits equal to two when traversing nodes. Due to the nature of this kind of open queueing model, the formulas regarding utilization factor, system demand and response time are very similar between the components.

4.2 Notation

We will define the notation in the following formulas: Since the structure of the network is very different between the branches of the tree, we need a notation that express how many (and which) nodes, sensors and actuators we have in each branch:

- R , is the set of regional nodes in the network. Every element in the set is express through an unique number.

- $L^{\{i\}}$, is the set of local nodes that have in common the same regional node with identifier i .
- $LAN^{\{i,j\}}$, is the set of lan's types that the local node j (which belongs to the branch of regional node i) is connected to.
- $S^{\{i,j,k\}}$, is the set of sensors' types that are placed in the LAN k (which belongs to the j local node in the i regional node).
- $A^{\{i,j,k\}}$, is the set of actuators' types that are placed in the LAN k (which belongs to the j local node in the i regional node).
- $S_t^{\{i,j,k\}}$, is the set of sensors which are of type t , placed in the LAN k (which belongs to the j local node in the i regional node).
- $A_t^{\{i,j,k\}}$, is the set of actuators which are of type t , placed in the LAN k (which belongs to the j local node in the i regional node).

Now for the parameters we are going to use:

- $\lambda_t, \lambda_e, \lambda_c, \lambda_b$: λ is the arrival rate, respectively for the data type telemetry, transition, command and batch.
- V : mean number of visits.
- S : service time.
- D : service demand.
- U : utilization factor.
- R : response time.

The elements we are going to analyze are:

- cn : central node.
- cns : central node storage
- rn : regional node.

- ln : local node.
- li : lan in.
- lo : lan out.

We will navigate the queue network starting from the sensors and actuators until reaching the central node.

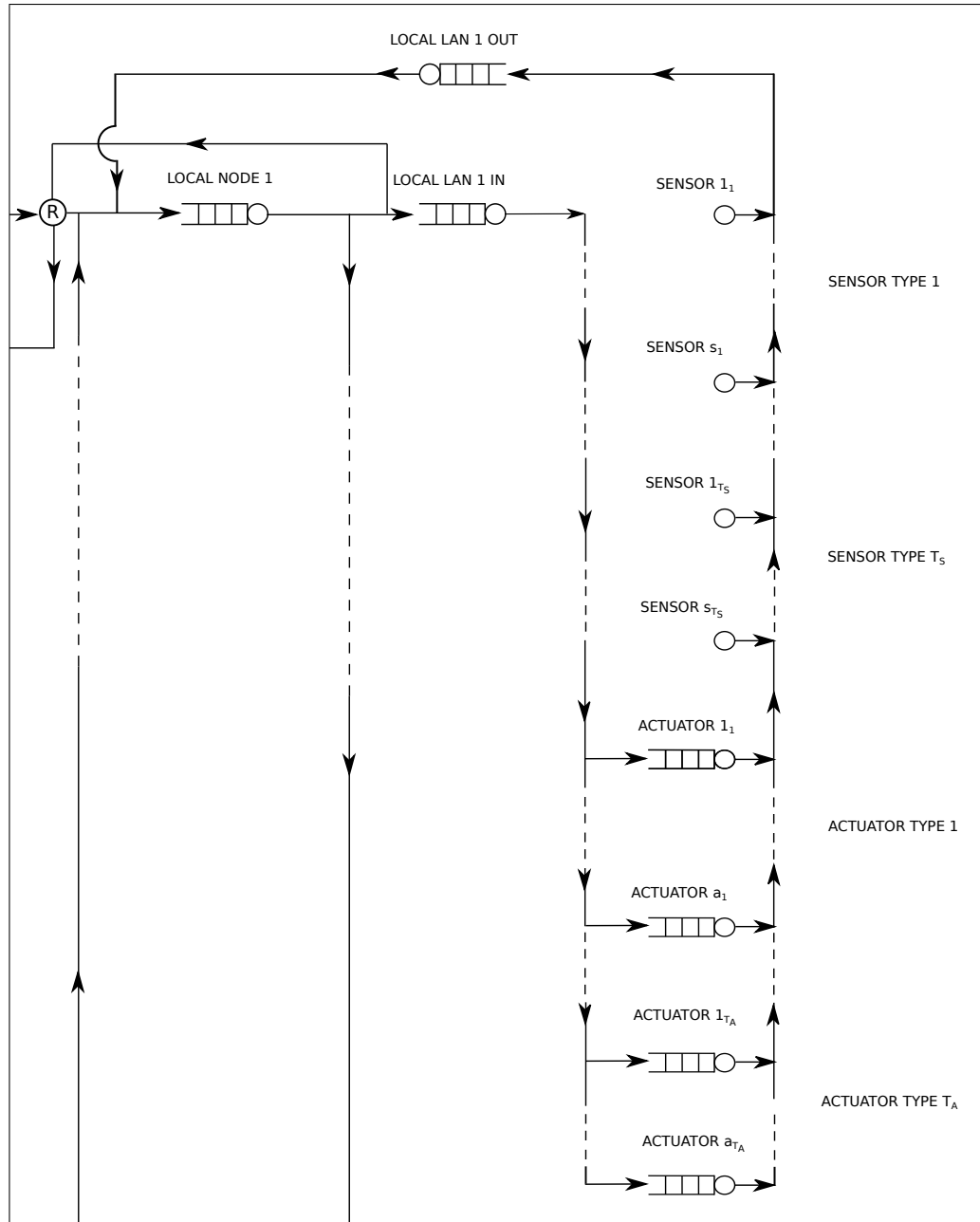


Figure 6: Focus on actuators, sensors, local nodes, and local LANs

4.3 Actuators and Sensors

Actuators are modeled as queues since they can receive command from the central, regional and local nodes (to which they are connected) in response to a transition event.

Regarding the sensors, they only send data, so we don't need to model them as queues. $\lambda_{s,t}^{\{w\}}$ and $\lambda_{s,e}^{\{w\}}$ are the rates at which the sensor sends a telemetry or a transition message respectively, in the case the sensor is a type w of sensor. The same is valid for the transition events rate of the actuators with $\lambda_{a,e}^{\{w\}}$.

Before going to the main formula we need to do some pre-computation. Let's suppose that we are in an actuators that belongs to the regional i , in the local j of this regional, and in the lan k of the local that we are considering.

Firstly we need some sets of actuators' types in various places in the network:

- T' : the set of all actuators in all the system (so managed by the central node). Analytically, it can be expressed by the following:

$$T' = \{t \mid (\forall m, n, s \mid m \in R \wedge n \in L^{\{m\}} \wedge s \in LAN^{\{m,n\}} \wedge t \in A^{\{m,n,s\}})\} \quad (1)$$

- T'' : the set of all actuators managed by the regional node i that our local node refers to. Analytically:

$$T'' = \{t \mid (\forall n, s \mid n \in L^{\{i\}} \wedge s \in LAN^{\{i,n\}} \wedge t \in A^{\{i,n,s\}})\} \quad (2)$$

- T''' : the set of all actuators managed by our local node j (that belongs to regional i). So:

$$T''' = \{t \mid (\forall s \mid s \in LAN^{\{i,j\}} \wedge t \in A^{\{i,j,s\}})\} \quad (3)$$

Note that in the next sections we are not going to specify how T' , T'' , T''' are characterized analytically, but the reasoning is the same as we did in

the last lines.

Then let's compute the number of actuators in all the system's branches for each type $t \in T'$ of actuator in the system:

$$a'_t = \sum_{\forall m,n,s \mid m \in R \wedge n \in L^{\{m\}} \wedge s \in LAN^{\{m,n\}}} |A_t^{\{m,n,s\}}| \quad (4)$$

And let's do the same for $t \in T''$ (which are the number of actuators for each type seen only in the branch of the regional i):

$$a''_t = \sum_{\forall n,s \mid n \in L^{\{i\}} \wedge s \in LAN^{\{i,n\}}} |A_t^{\{i,n,s\}}| \quad (5)$$

And for $t \in T'''$ (which are the number of actuators for each type seen only in the branch of the local j that belongs to the regional i):

$$a'''_t = \sum_{\forall s \mid s \in LAN^{\{i,j\}}} |A_t^{\{i,j,s\}}| \quad (6)$$

As before, note that in the next sections it's not necessary to specify how to characterized a'_t , a''_t , a'''_t because the reasoning is the same as in the last lines.

Finally we need to denote the probability of each type of actuator to receive a command. Indeed, the probability to receive a command it's not evenly distributed over the actuators' types, so there may be the case that a type of actuator receives more commands with respect to another type. For this reason we denote p_t as the probability associated to the t type of actuator. Since when a command is generated it must be delivered to at least an actuator, so we need to specify that the sum of all type's probabilities is 1:

$$\sum_{t \in T'} p_t = 1 \quad (7)$$

Now we are ready to do the real computation. Denoting $w \in T'$ as the type of the actuator that we are considering in our branch (i, j, k) , we can write:

$$\begin{aligned}
V_{a,c} &= 1 \\
\lambda_{a,c} &= \lambda_{ln,c} * \frac{p_w}{\sum_{t \in T'''} a_t''' * p_t}
\end{aligned} \tag{8}$$

The command event formula will be thoroughly explained in the next section, as a brief explanation it is composed by the rate of command messages sent by the local node, multiplied the weighted ratio between the single actuator's probability of receiving and all actuators in the same LAN. From the previous formulas and the open queue model we can compute the Service Demand, Utilization factor and Response time.

$$\begin{aligned}
D_{a,c} &= V_{a,c} * S_{a,c} \\
U_{a,c} &= \lambda_{a,c} * D_{a,c} \\
R_{a,c} &= \frac{D_{a,c}}{1 - U_{a,c}}
\end{aligned} \tag{9}$$

4.4 LANs

This part will deal with all kind of LANs available in the lower level, the formulas are similar so we will show a general one.

As before, let's suppose that we are in a branch of the system in which we belong to the i regional node, j local node and LAN k .

The definitions of T' , T'' , T''' are the same as the previous section (same thing for a_t' , a_t'' , a_t''' and the probability of each type p_t) but now we also need of the set of types on this branch (that in this case is the LAN k):

$$T = A^{\{i,j,k\}} \tag{10}$$

Consequently let's compute the number of actuators in this branch for each type $t \in T$ of actuator:

$$a_t = |A_t^{\{i,j,k\}}| \tag{11}$$

Now, denoting with $\lambda_{s,t}^{\{w\}}$ the arrival rate from the sensor of type $w \in T'$, we can compute the main formulas for the model. Dividing the LAN in Lan

In and Out as we modeled them, we see Lan In having:

$$\begin{aligned} V_{li,c} &= 1 \\ \lambda_{li,c} &= \lambda_{ln,c} * \frac{\sum_{t \in T} a_t * p_t}{\sum_{t \in T'''} a_t''' * p_t} \end{aligned} \quad (12)$$

Indeed the Lan In receives only command events directed to its actuators and sensors.

Now Lan Out, which will send both transitions and telemetries:

$$\begin{aligned} V_{lo,t} &= 1 \\ V_{lo,e} &= 1 \\ \lambda_{lo,t} &= \sum_{w \in T} |S_w^{\{i,j,k\}}| * \lambda_{s,t}^{\{w\}} \\ \lambda_{lo,e} &= \sum_{w \in T} |S_w^{\{i,j,k\}}| * \lambda_{s,e}^{\{w\}} + \sum_{w \in T} |A_w^{\{i,j,k\}}| * \lambda_{a,e}^{\{w\}} \end{aligned} \quad (13)$$

We will explain the lambda formulas since from now on we will see some iterations of them that are simpler so explaining these ones should suffice.

The telemetries (in the LAN OUT) are straightforward, fixed our lan k, the local node j and the regional node i, the total rate over the network is given by the composition of all sensors sending data over this lan thus we multiply the number of each type of sensors with their rate.

The transition events (only in LAN OUT) are given by the sum of two elements, the first one representing the rate of all sensors in this lan and the second the rate of all actuators.

By the approach we are using on command events (in the LAN IN), the rate of command events over the lan is given by the probability of a command event being generated from the local node, i.e. the rate of generation of those events multiplied by the ratio of the number of actuators in this lan over all actuators connected to the local node, which are the ones that can be targeted with a command event.

The ratio is weighted by the type of actuator since each type has a different probability of receiving a command message.

From the previous formulas and the open queue model we can compute the Service Demand, Utilization factor and Response time for the LAN IN:

$$\begin{aligned}
D_{li,e} &= V_{li,e} * S_{li,e} \\
D_{li,c} &= V_{li,c} * S_{li,c} \\
U_{li,e} &= \lambda_{li,e} * D_{li,e} \\
U_{li,c} &= \lambda_{li,c} * D_{li,c} \\
U_{li} &= U_{li,e} + U_{li,c} \\
R_{li,e} &= \frac{D_{li,e}}{1-U_{li}} \\
R_{li,c} &= \frac{D_{li,c}}{1-U_{li}}
\end{aligned} \tag{14}$$

And for LAN OUT:

$$\begin{aligned}
D_{lo,t} &= V_{lo,t} * S_{lo,t} \\
D_{lo,c} &= V_{lo,c} * S_{lo,c} \\
U_{lo,t} &= \lambda_{lo,t} * D_{lo,t} \\
U_{lo,c} &= \lambda_{lo,c} * D_{lo,c} \\
U_{lo} &= U_{lo,t} + U_{lo,c} \\
R_{lo,t} &= \frac{D_{lo,t}}{1-U_{lo}} \\
R_{lo,c} &= \frac{D_{lo,c}}{1-U_{lo}}
\end{aligned} \tag{15}$$

4.5 Local Node

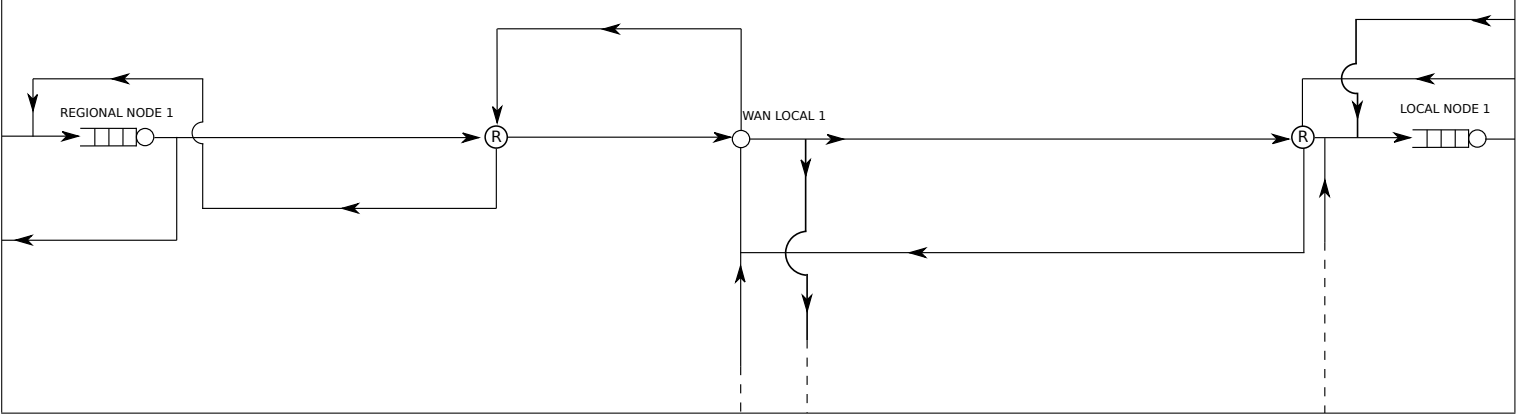


Figure 7: Focus on local nodes, and local WANs and regional nodes

Taken a local node j , and its upper level regional node i , remembering the definition of T' , T'' and T''' from the previous section and modifying the definition of T by amplifying the branch to include all LANs managed by this local node i we will define the new formulas. Note that at this point $T = T'''$, because our branch now is a local node.

So, denoting $p_{cmd,ln}$ as the probability (or percentage) that a transition, arriving in the local node, generates a command and $\lambda_{lo,t}^{\{s\}}$ as the rate of the telemetry messages by the LAN s , we can write:

$$\begin{aligned}
V_{ln,t} &= 1 \\
V_{ln,e} &= 2 \\
V_{ln,c} &= 1 \\
V_{ln,b} &= 1 \\
\lambda_{ln,t} &= \sum_{s \in LAN^{i,j}} \lambda_{lo,t}^{\{s\}} \\
\lambda_{ln,e} &= \sum_{s \in LAN^{i,j}} \lambda_{lo,e}^{\{s\}} \\
\lambda_{ln,c} &= \lambda_{rn,c} * \frac{\sum_{t \in T} a_t * p_t}{\sum_{t \in T''} a_t'' * p_t} + p_{cmd,ln} * \lambda_{ln,e} \\
\lambda_{ln,b} &= p_{cmd,ln} * \lambda_{ln,e}
\end{aligned} \tag{16}$$

The main difference here is that we don't have a fixed LAN so we need to span across all LANs of the local node, so, for the case of transitions and telemetry, we sum all the rates of each LAN that are in the branch of the local node j .

From the previous formulas and the open queue model we can compute the Service Demand, Utilization factor and Response time.

$$\begin{aligned}
D_{ln,t} &= V_{ln,t} * S_{ln,t} \\
D_{ln,e} &= V_{ln,e} * S_{ln,e} \\
D_{ln,c} &= V_{ln,c} * S_{ln,c} \\
D_{ln,b} &= V_{ln,b} * S_{ln,b} \\
U_{ln,t} &= \lambda_{ln,t} * D_{ln,t} \\
U_{ln,e} &= \lambda_{ln,e} * D_{ln,e} \\
U_{ln,c} &= \lambda_{ln,c} * D_{ln,c} \\
U_{ln,b} &= \lambda_{ln,b} * D_{ln,b} \\
U_{ln} &= U_{ln,t} + U_{ln,e} + U_{ln,c} + U_{ln,b} \\
R_{ln,t} &= \frac{D_{ln,t}}{1 - U_{ln}} \\
R_{ln,e} &= \frac{D_{ln,e}}{1 - U_{ln}} \\
R_{ln,c} &= \frac{D_{ln,c}}{1 - U_{ln}} \\
R_{ln,b} &= \frac{D_{ln,b}}{1 - U_{ln}}
\end{aligned} \tag{17}$$

4.6 WAN

Since the WAN is an external object that we decided to model as a delay, we will consider it in the response time functions but it won't have formulas of its own. In particular the value of the delay is equal to the round trip time divided by two of the specific link.

4.7 Regional Node

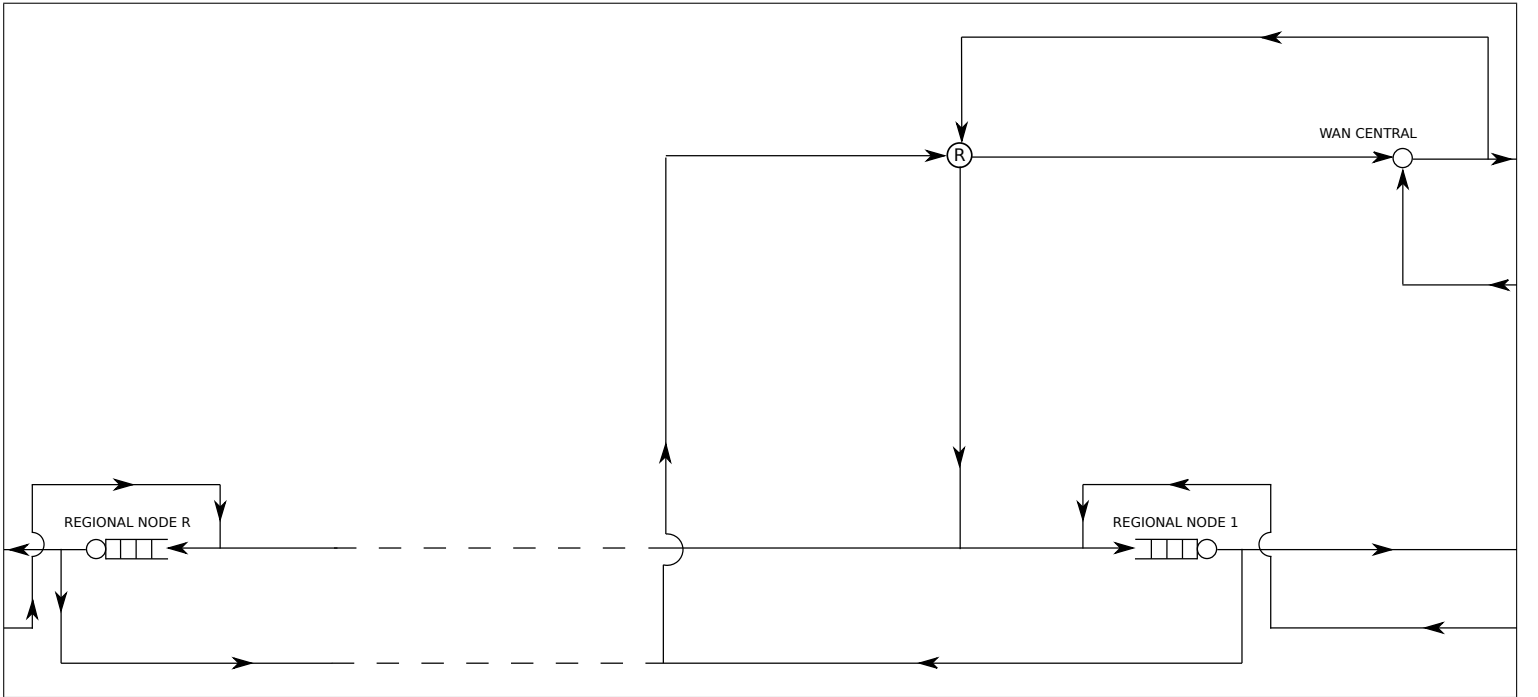


Figure 8: Focus on regional nodes and central WAN

As for the local node, we fix the regional node i , but we iterate over all the local nodes of the regional i . We will extend again the definition of T . Regarding telemetry and transitions, we will also need to iterate over all local nodes below the regional node i .

The main difference in the command rate will be that we won't notice a

local level fix in the command arrival rate.

$$\begin{aligned}
V_{rn,t} &= 1 \\
V_{rn,e} &= 2 \\
V_{rn,c} &= 1 \\
V_{rn,b} &= 1 \\
\lambda_{rn,t} &= \sum_{n \in L^{\{i\}}} \frac{\lambda_{ln,t}^{\{n\}}}{aggr_{ln}^{\{n\}}} \\
\lambda_{rn,e} &= \sum_{n \in L^{\{i\}}} \lambda_{ln,e}^{\{n\}} \\
\lambda_{rn,c} &= \lambda_{cn,c} * \frac{\sum_{t \in T} a_t * p_t}{\sum_{t \in T'} a'_t * p_t} + p_{cmd,rn} * \lambda_{rn,e} \\
\lambda_{rn,b} &= \sum_{n \in L^{\{i\}}} \lambda_{ln,b}^{\{n\}} + p_{cmd,rn} * \lambda_{rn,e}
\end{aligned} \tag{18}$$

where $\lambda_{ln,b}^{\{n\}}$ denotes the rate of n -th local node of the regional that we are considering, i.e. i -th regional. Also $aggr_{ln}^{\{n\}}$ denotes the aggregation done by the local node n and $p_{cmd,rn}$ is the probability that a transition generates a command in the regional node.

From the previous formulas and the open queue model we can compute the Service Demand, Utilization factor and Response time.

$$\begin{aligned}
D_{rn,t} &= V_{rn,t} * S_{rn,t} \\
D_{rn,e} &= V_{rn,e} * S_{rn,e} \\
D_{rn,c} &= V_{rn,c} * S_{rn,c} \\
D_{rn,b} &= V_{rn,b} * S_{rn,b} \\
U_{rn,t} &= \lambda_{rn,t} * D_{rn,t} \\
U_{rn,e} &= \lambda_{rn,e} * D_{rn,e} \\
U_{rn,c} &= \lambda_{rn,c} * D_{rn,c} \\
U_{rn,b} &= \lambda_{rn,b} * D_{rn,b} \\
U_{rn} &= U_{rn,t} + U_{rn,e} + U_{rn,c} + U_{rn,b} \\
R_{rn,t} &= \frac{D_{rn,t}}{1 - U_{rn}} \\
R_{rn,e} &= \frac{D_{rn,e}}{1 - U_{rn}} \\
R_{rn,c} &= \frac{D_{rn,c}}{1 - U_{rn}} \\
R_{rn,b} &= \frac{D_{rn,b}}{1 - U_{rn}}
\end{aligned} \tag{19}$$

4.8 Central Node

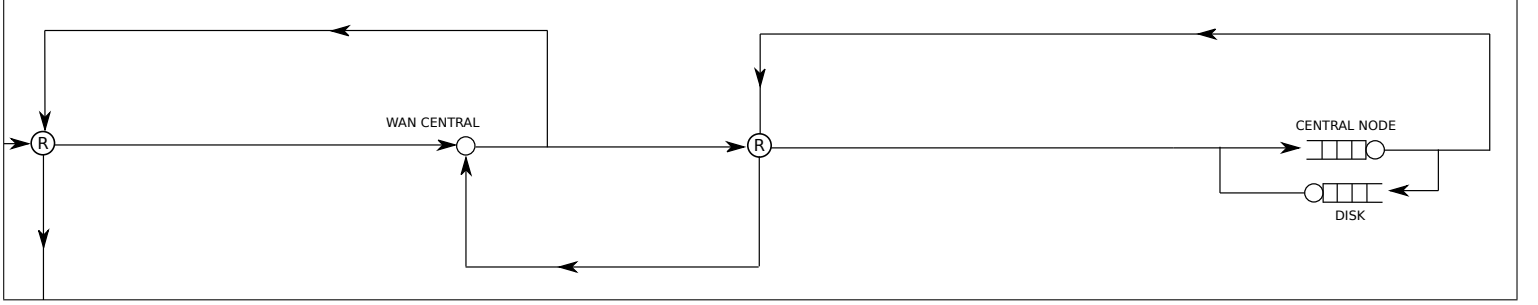


Figure 9: Focus on regional nodes and central node

In the highest level, we will need to work over all the regional nodes in R . Central node command rate is oblivious to the regional node command events, thus the arrival rate is composed of only one element.

$$\begin{aligned}
 V_{cn,t} &= 1 \\
 V_{cn,e} &= 2 \\
 V_{cn,c} &= 1 \\
 V_{cn,b} &= 1 \\
 \lambda_{cn,t} &= \sum_{m \in R} \frac{\lambda_{rn,t}^{\{m\}}}{aggr_{rn}^{\{m\}}} \\
 \lambda_{cn,e} &= \sum_{m \in R} \lambda_{rn,e}^{\{m\}} \\
 \lambda_{cn,c} &= p_{cmd,cn} * \lambda_{cn,e} \\
 \lambda_{cn,b} &= \sum_{m \in R} \lambda_{rn,b}^{\{m\}} + p_{cmd,cn} * \lambda_{cn,e}
 \end{aligned} \tag{20}$$

where $\lambda_{rn,b}^{\{m\}}$ is the rate of the batch data coming from the m -th regional. Obviously the ratio is not present in this case since it would be 1, since $T = T'$.

From the previous formulas and the open queue model we can compute the

Service Demand, Utilization factor and Response time.

$$\begin{aligned}
D_{cn,t} &= V_{cn,t} * S_{cn,t} \\
D_{cn,e} &= V_{cn,e} * S_{cn,e} \\
D_{cn,c} &= V_{cn,c} * S_{cn,c} \\
D_{cn,b} &= V_{cn,b} * S_{cn,b} \\
U_{cn,t} &= \lambda_{cn,t} * D_{cn,t} \\
U_{cn,e} &= \lambda_{cn,e} * D_{cn,e} \\
U_{cn,c} &= \lambda_{cn,c} * D_{cn,c} \\
U_{cn,b} &= \lambda_{cn,b} * D_{cn,b} \\
U_{cn} &= U_{cn,t} + U_{cn,e} + U_{cn,c} + U_{cn,b} \\
R_{cn,t} &= \frac{D_{cn,t}}{1-U_{cn}} \\
R_{cn,e} &= \frac{D_{cn,e}}{1-U_{cn}} \\
R_{cn,c} &= \frac{D_{cn,c}}{1-U_{cn}} \\
R_{cn,b} &= \frac{D_{cn,b}}{1-U_{cn}}
\end{aligned} \tag{21}$$

4.8.1 Central node storage system

In this first draft, the central node storage system is not characterized with a specific system(RAID 1-5, etc) and it is modeled as a simple queue filled by the central node.

Since the central node will do some aggregation and computations on the telemetry messages received by the regional nodes, the rate arriving on the disk will be divided by the aggregation parameter of the central node.

$$\begin{aligned}
V_{cns,t} &= 1 \\
V_{cns,e} &= 1 \\
\lambda_{cns,t} &= \frac{\lambda_{cn,t}}{aggr_{cn}} \\
\lambda_{cns,e} &= \lambda_{cn,e} \\
\lambda_{cns,b} &= \lambda_{cn,b}
\end{aligned} \tag{22}$$

From the previous formulas and the open queue model we can compute the Service Demand, Utilization factor and Response time.

$$\begin{aligned}
D_{cns,t} &= V_{cns,t} * S_{cns,t} \\
D_{cns,e} &= V_{cns,e} * S_{cns,e} \\
D_{cns,b} &= V_{cns,b} * S_{cns,b} \\
U_{cns,t} &= \lambda_{cns,t} * D_{cns,t} \\
U_{cns,e} &= \lambda_{cns,e} * D_{cns,e} \\
U_{cns,b} &= \lambda_{cns,b} * D_{cns,b} \\
U_{cns} &= U_{cns,t} + U_{cns,e} + U_{cns,b} \\
R_{cns,t} &= \frac{D_{cns,t}}{1-U_{cns}} \\
R_{cns,e} &= \frac{D_{cns,e}}{1-U_{cns}} \\
R_{cns,b} &= \frac{D_{cns,b}}{1-U_{cns}}
\end{aligned} \tag{23}$$