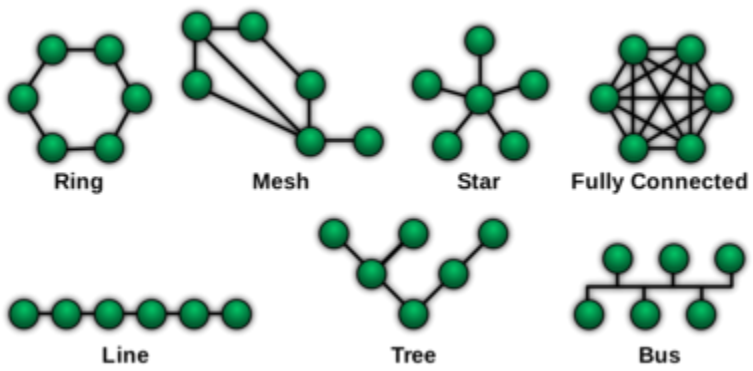
Spider-Web Network Topology



Today at our meeting, we discussed what our ideal AREDN topology should be. AREDN is supposed to be a mesh network, which means that ideally, little planning should be required. However, in practice, there's a trade off between redundancy and network stability:

> A physical topology that contains switching or bridge loops is attractive for redundancy reasons, yet a switched network must not have loops. The solution is to allow physical loops, but create a loop-free logical topology using link aggregation, shortest path bridging, spanning tree protocol or TRILL on the network switches. [1)]

Two key words in the quote above are “physical” (the way the nodes are physically connected), and “logical” (the paths that the data is allowed to use). This means that even if there are physical network loops, it's possible to use clever software protocols to avoid data to actually go round-and-round and flood the network. Which leads us to “Babel”, which is

> [...] a loop free protocol so, regardless of how the network is changing, routing loops will never form in the network.[2)]

It's already available in the nightly updates of AREDN, and should be available in the next stable version. Once most nodes have converted to Babel, I think we'll be able to start building more redundancy in the network by creating more physical loops.

Here's a technical paper I read a few years ago discussing a Spider-Web Topology. On page 36, they say that:

> [E]xperimental results shows the proposed topology is three times better than others compared typologies in performing tasks.

I have no idea if Babel is optimized or not for such a physical topology, but the idea is interesting.

1)
Switching Loop
2)
Adding Babel as an AREDN® Routing Protocol