



Development of the O2 prototype



Charis Kouzinopoulos

CERN

ALICE Offline week

Friday, 27 June 2014



Development of the O2 prototype

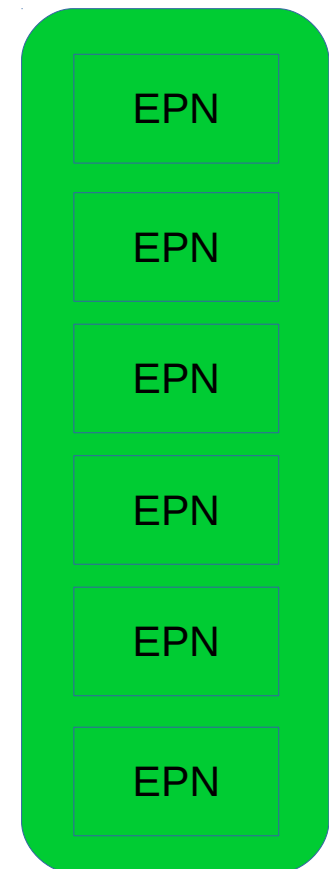
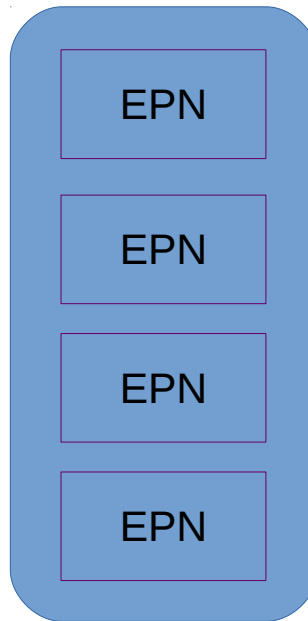
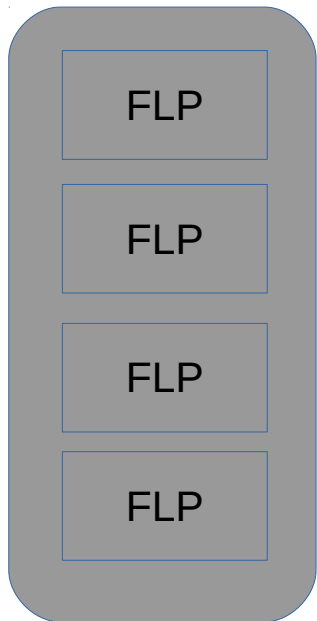
Current development status

- For the data transportation we have tested different implementations (details to follow)
- The code for the current implementation can be tracked at Github: <https://github.com/MohammadALTurany/FairRoot.git>
- Deployed at the DAQ test cluster
- Example scripts available:
/local/home/cwg13/new_test_21.05.2014/single/



Development of the O2 prototype

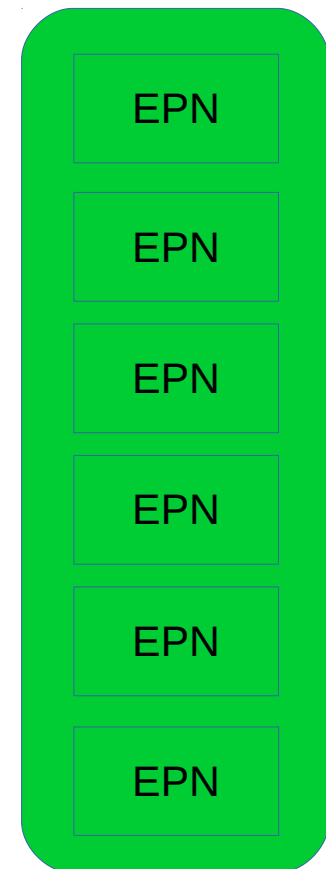
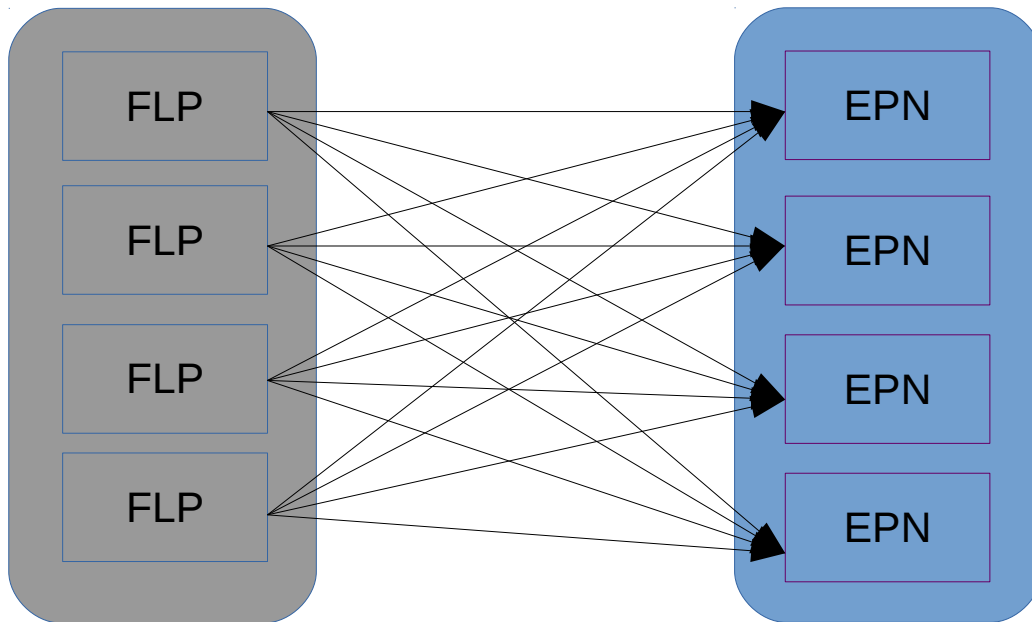
- For RUN3 there will be ~250 FLPs and ~2000 EPNs deployed
- There will be two levels of EPNs (~50 EPNs are enough for first level, ~100 for redundancy)





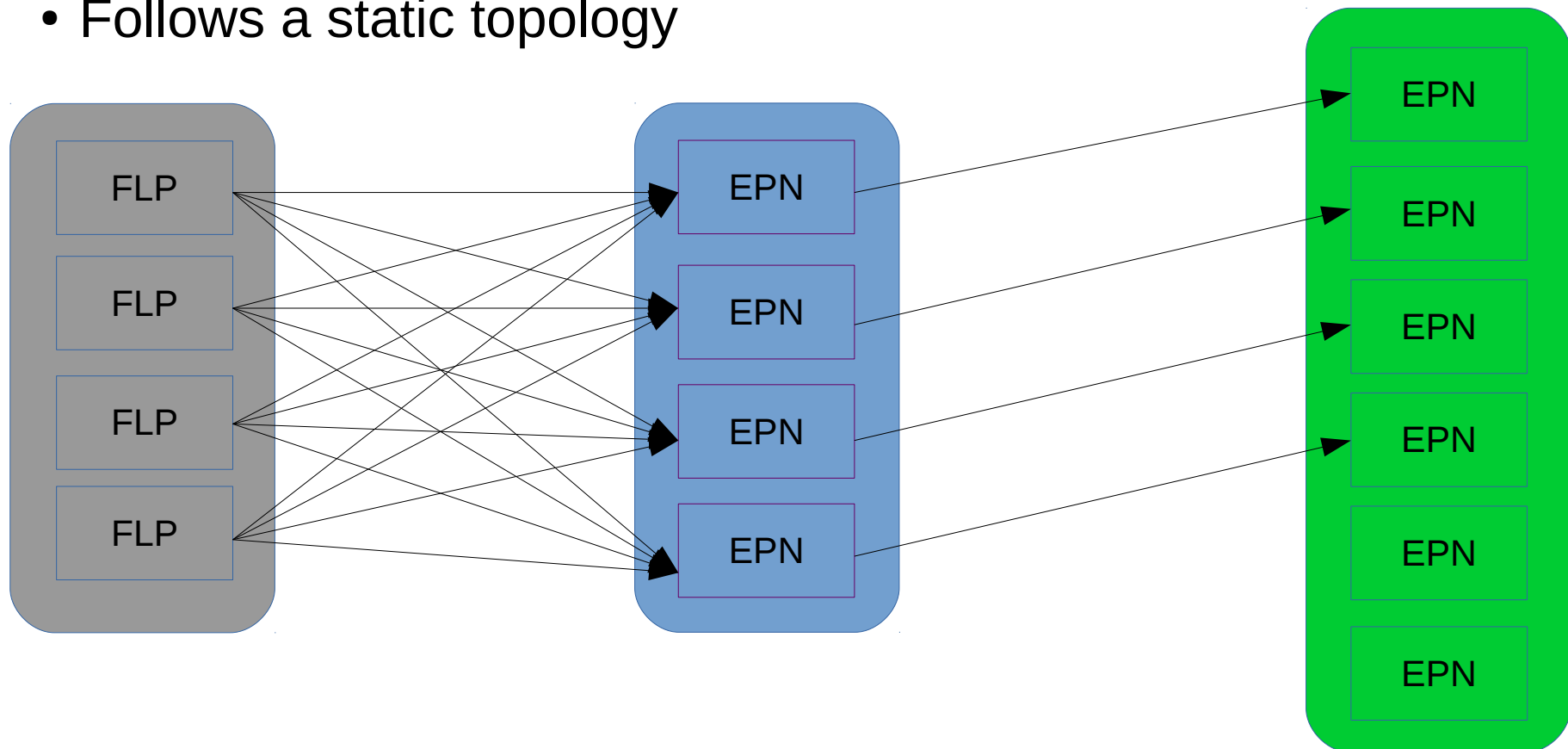
Development of the O2 prototype

- The ~100 first level EPNs connect to the binding FLPs
- FLPs push data frames to first-level EPNs on a round robin basis



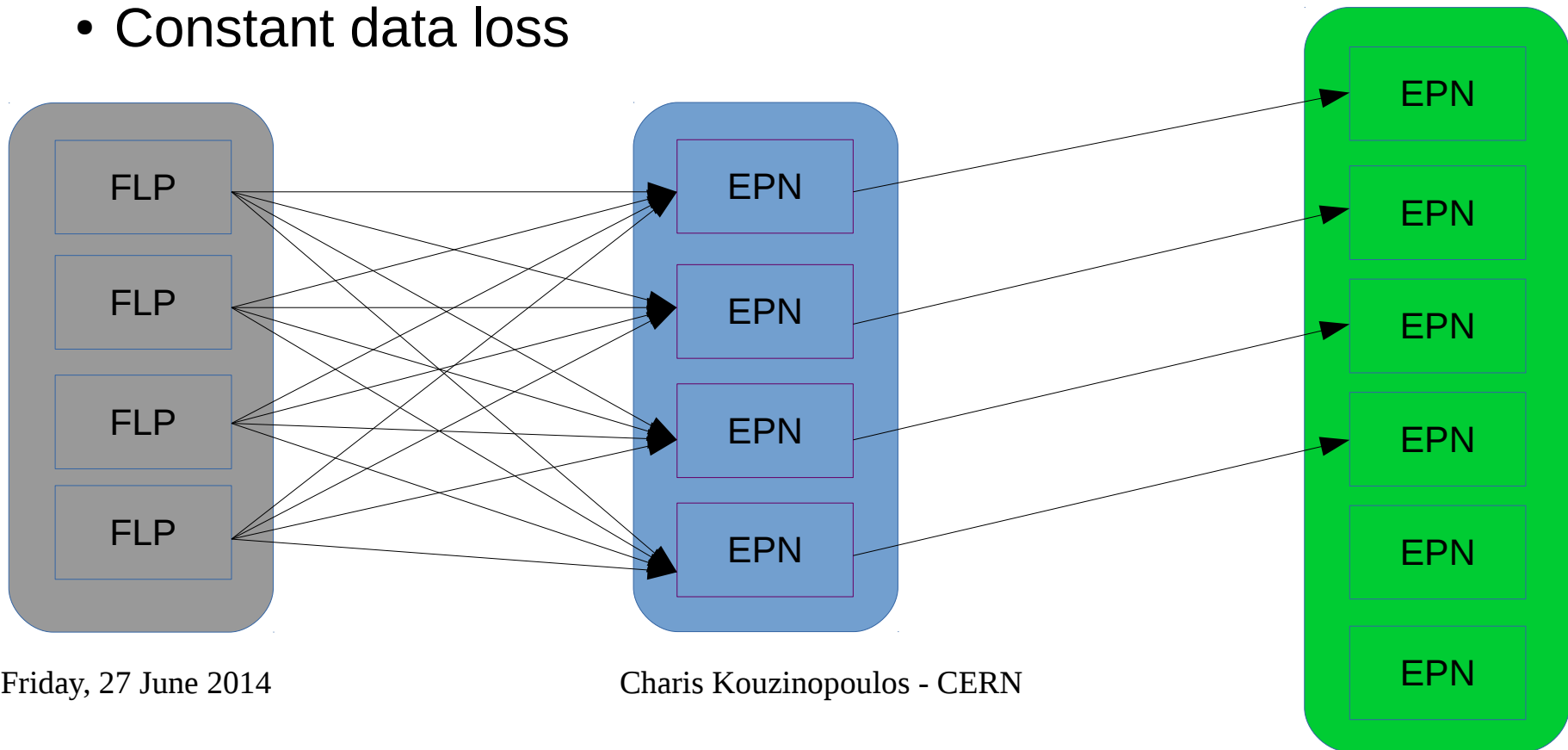
Development of the O2 prototype

- First-level EPNs merge the data as multi-part messages to avoid unnecessary copies and forward them to second-level EPNs
- Follows a static topology



Development of the O2 prototype

- **What happens if an EPN lags behind?**
 - Data will start queuing on the FLPs
- **What happens if an EPN dies?**
 - Constant data loss

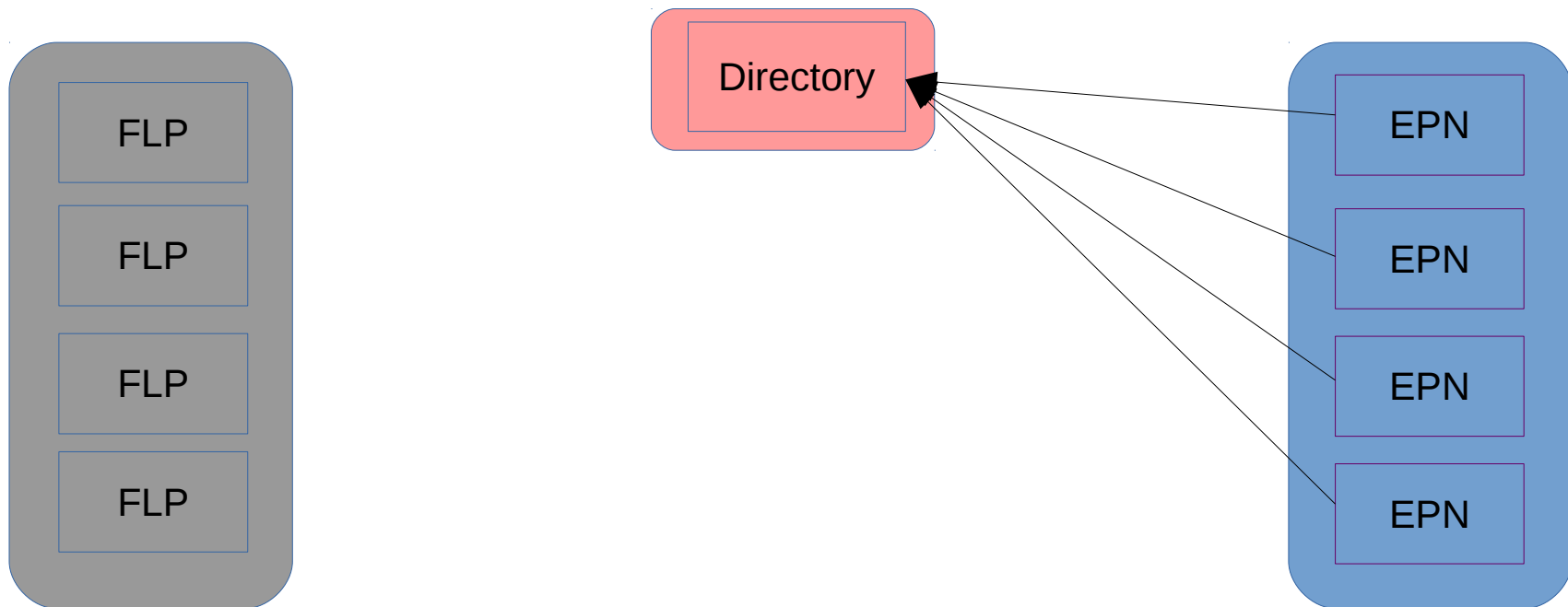




Development of the O2 prototype

A dynamic topology design was also considered:
(code at <https://github.com/kouzinopoulos/zmqproto.git>):

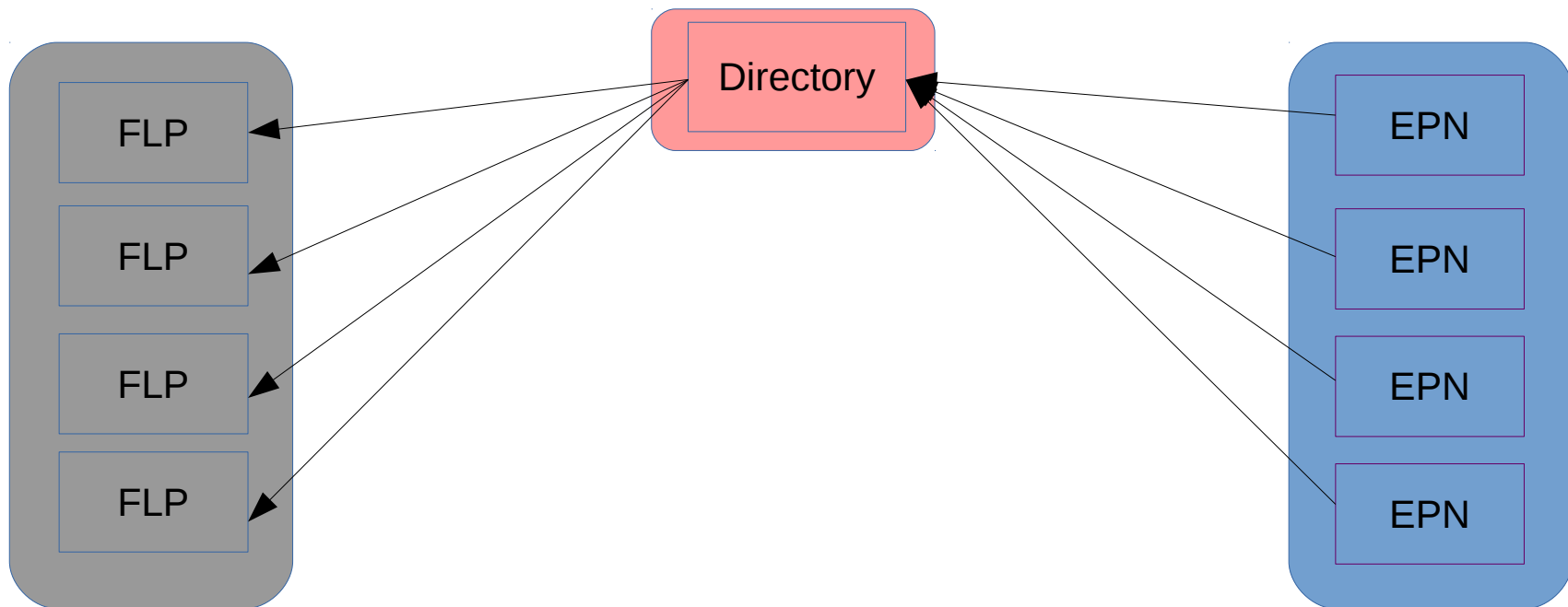
- The EPNs publish regularly an ID string to a directory node



Development of the O2 prototype

A dynamic topology design was also considered:

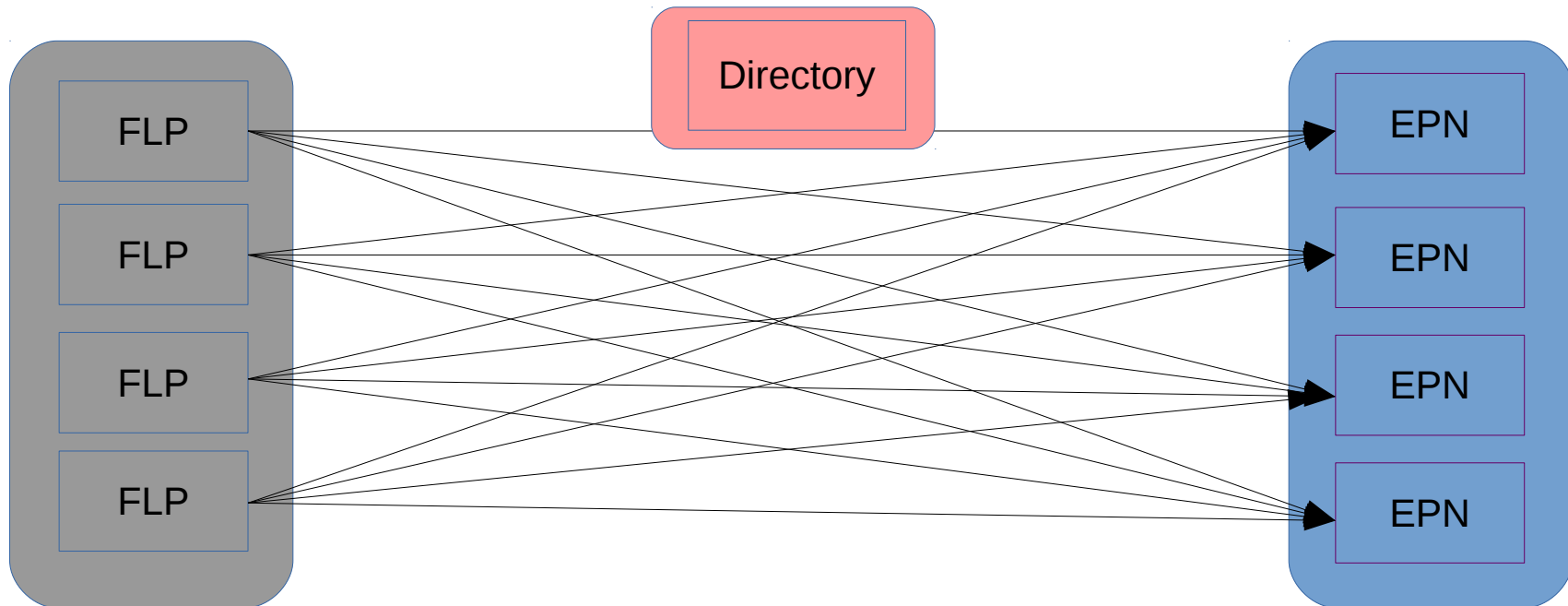
- The strings are stored in an ID vector that is then propagated to FLPs and cached locally



Development of the O2 prototype

A dynamic topology design was also considered:

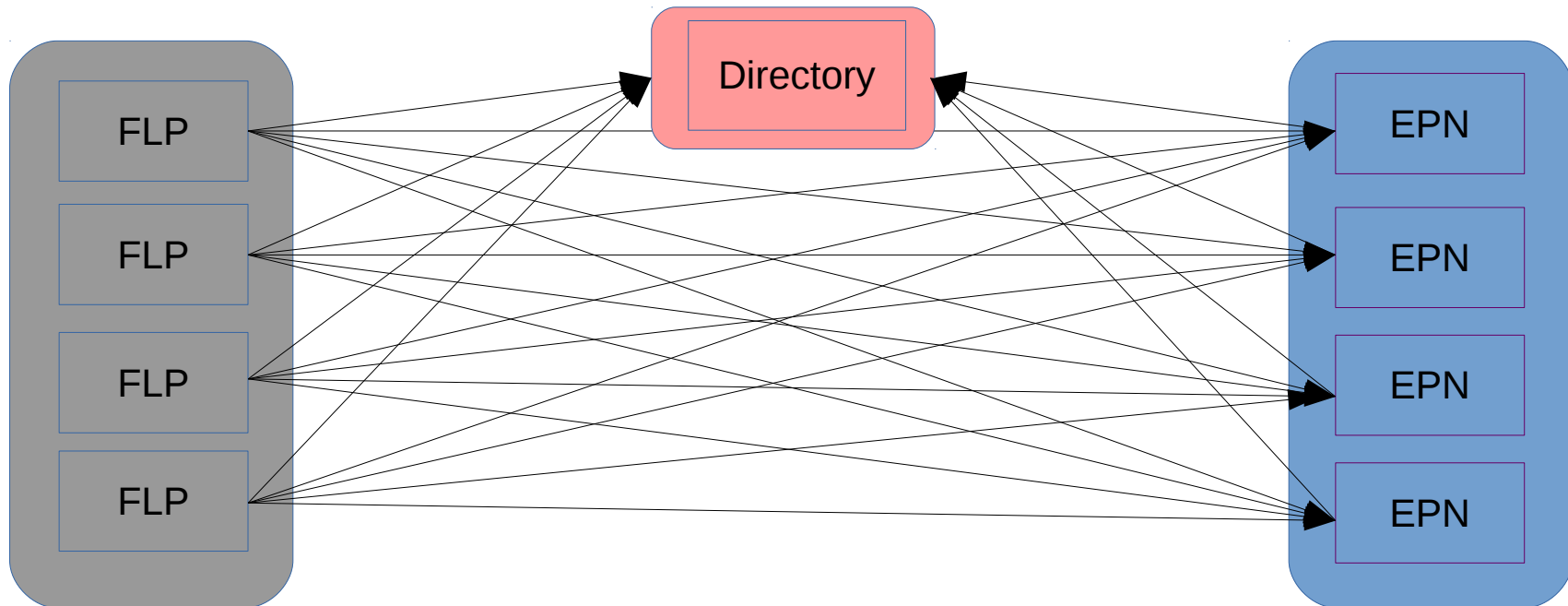
- The FLPs push data frames to the first-level EPNs



Development of the O2 prototype

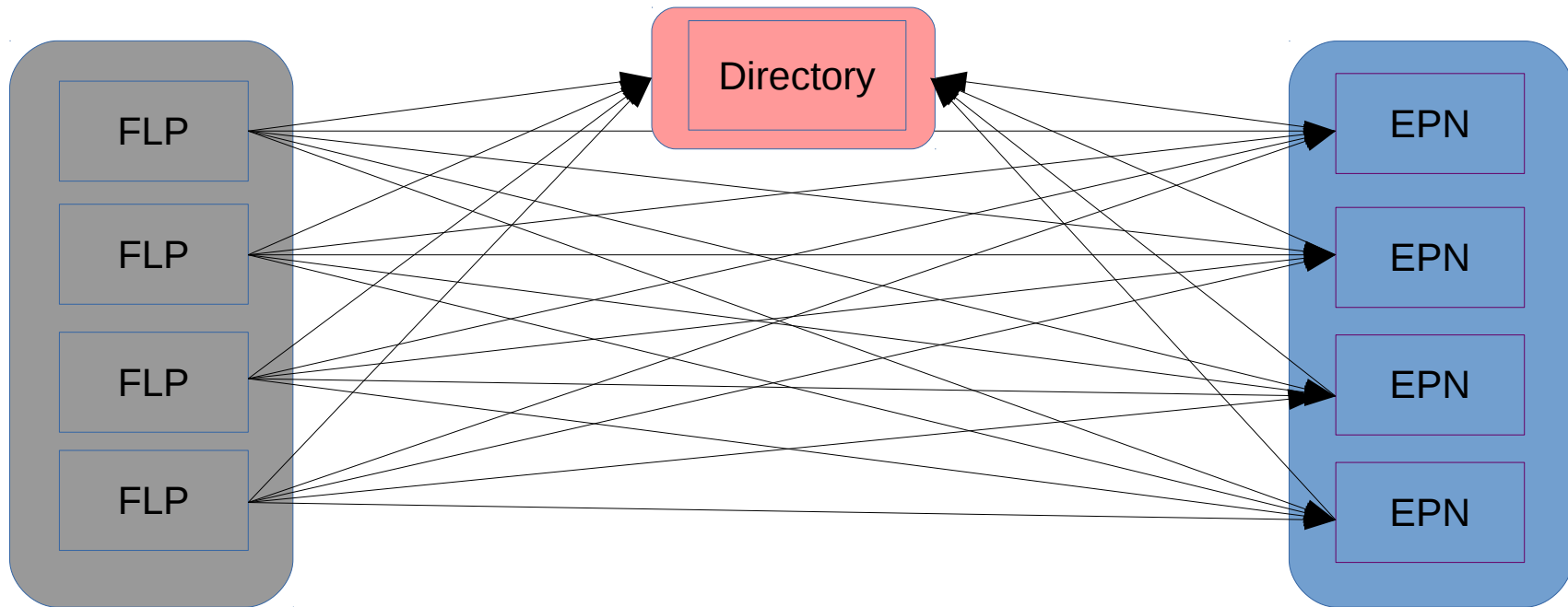
A dynamic topology design was also considered:

- The FLPs subscribe to the directory for vector updates every n secs



Development of the O2 prototype

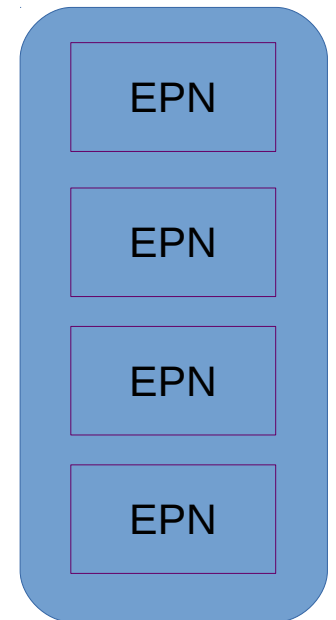
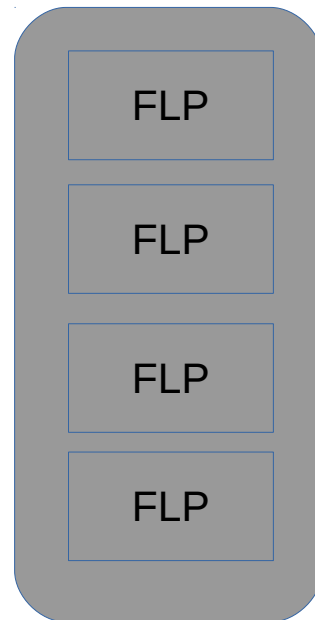
- What happens if the directory dies?
- The directory introduces a single point of failure to the network





Development of the O2 prototype

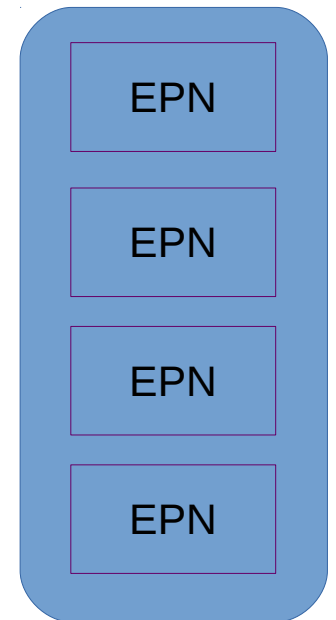
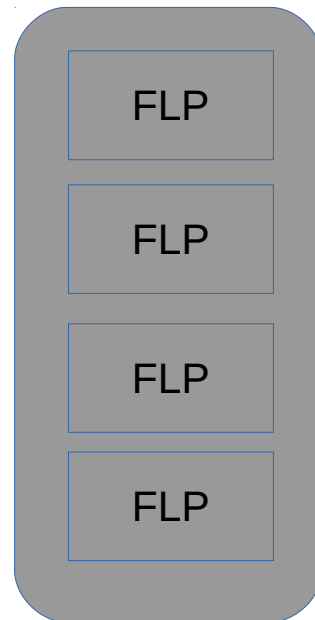
Current development





Development of the O2 prototype

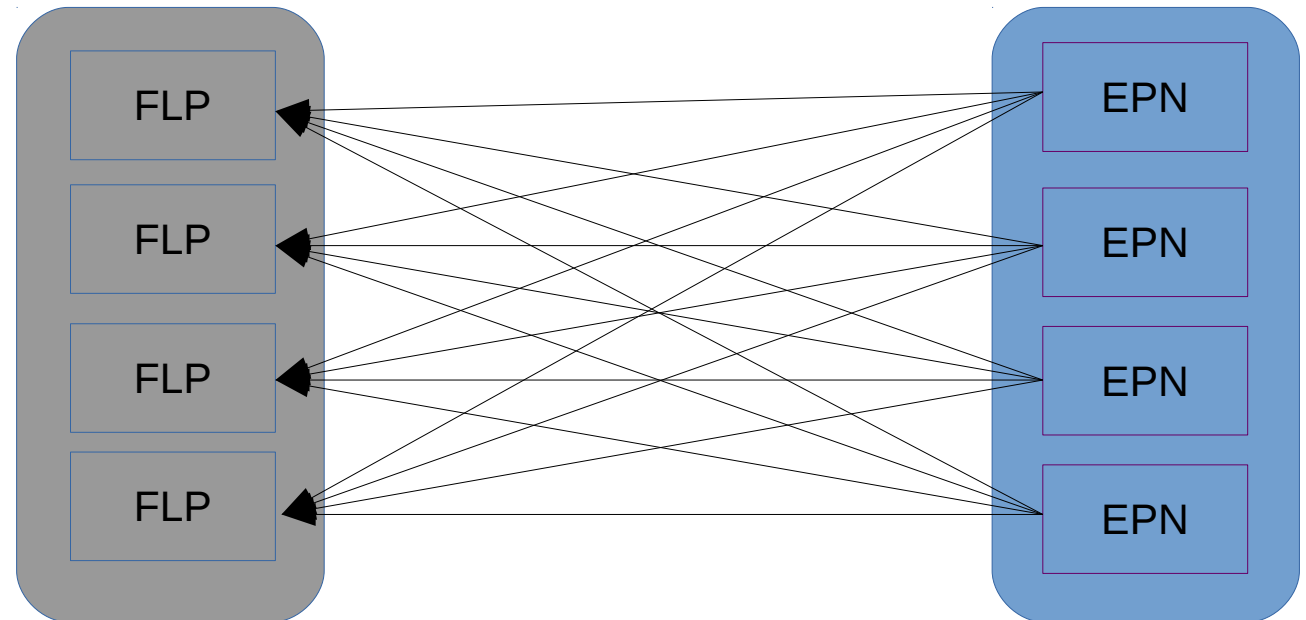
- The first-level EPNs publish an ID string in regular intervals





Development of the O2 prototype

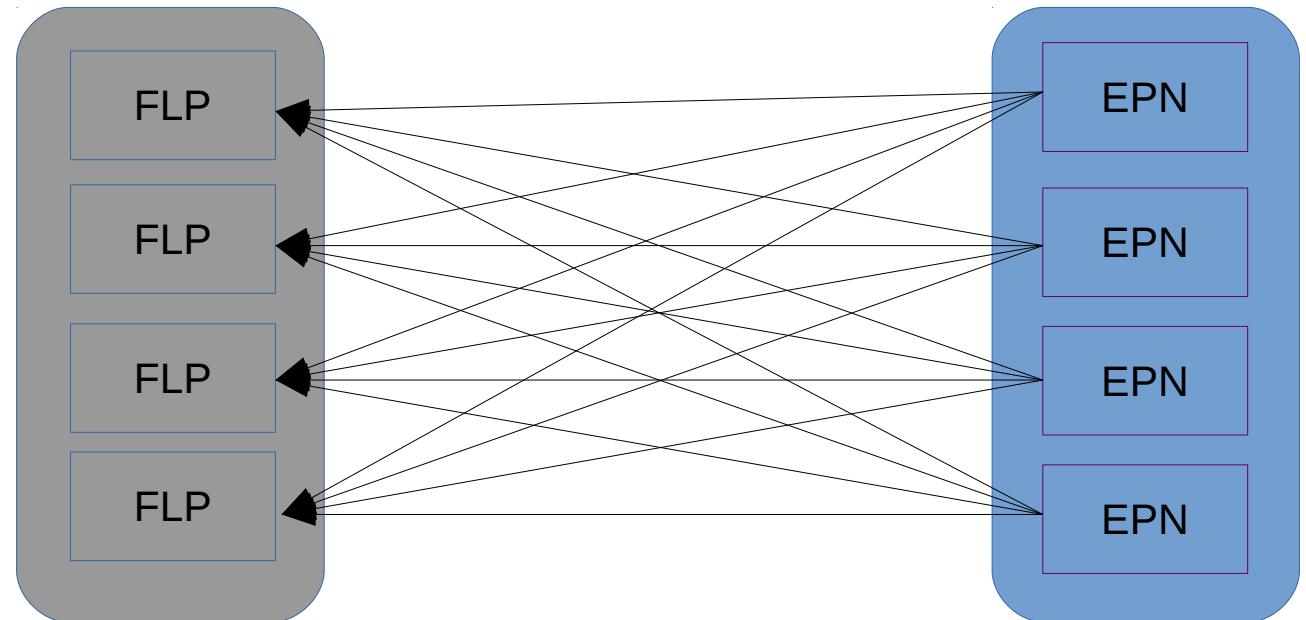
- The first-level EPNs publish an ID string in regular intervals
- The FLPs create an ID vector from the EPN IDs





Development of the O2 prototype

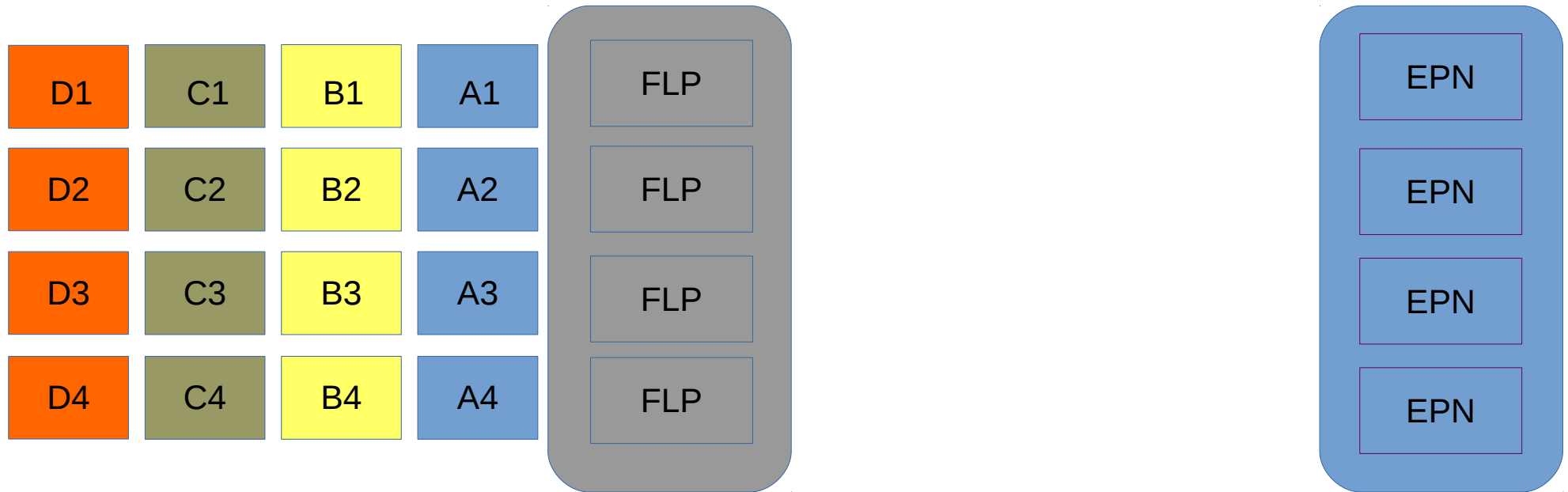
- The first-level EPNs publish an ID string in regular intervals
- The FLPs create an ID vector from the EPN IDs
- The FLPs open a socket per EPN and publish data frames based on the output of $F()$





Development of the O2 prototype

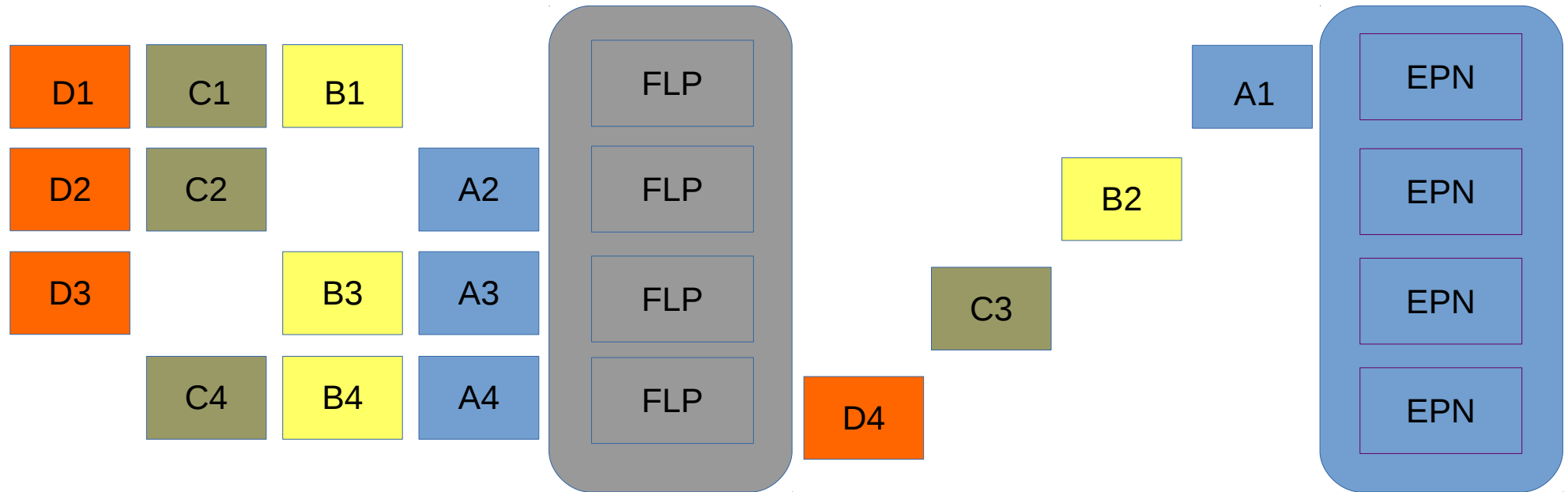
- To avoid network congestion, frames are cached on FLPs





Development of the O2 prototype

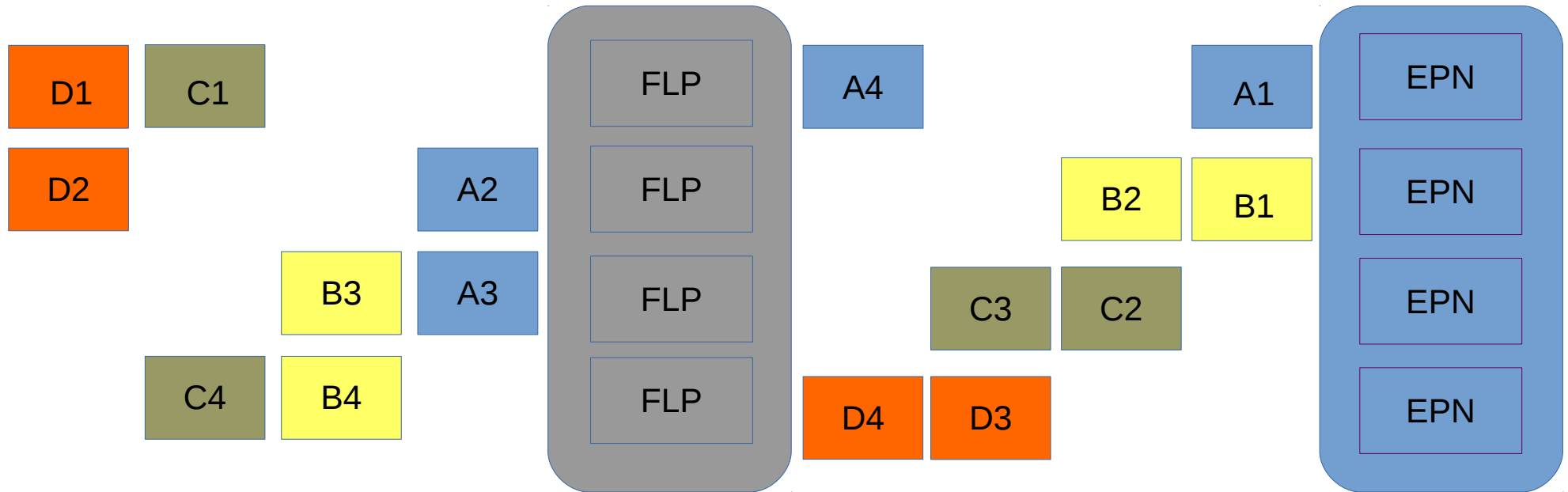
- To avoid network congestion, frames are cached on FLPs





Development of the O2 prototype

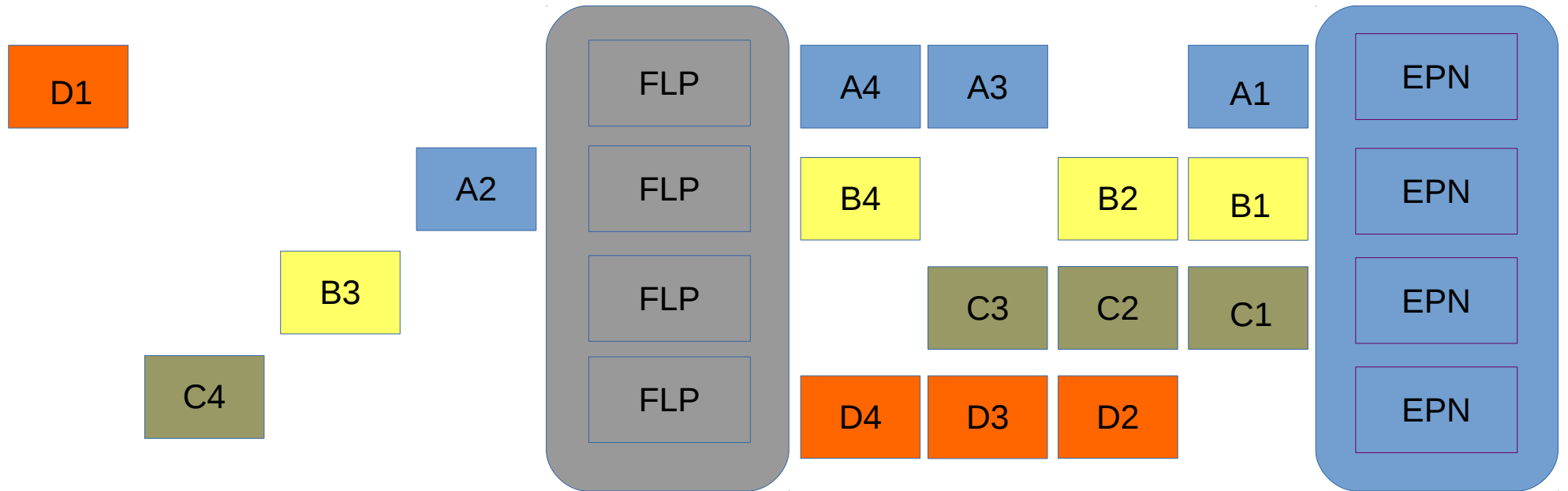
- To avoid network congestion, frames are cached on FLPs





Development of the O2 prototype

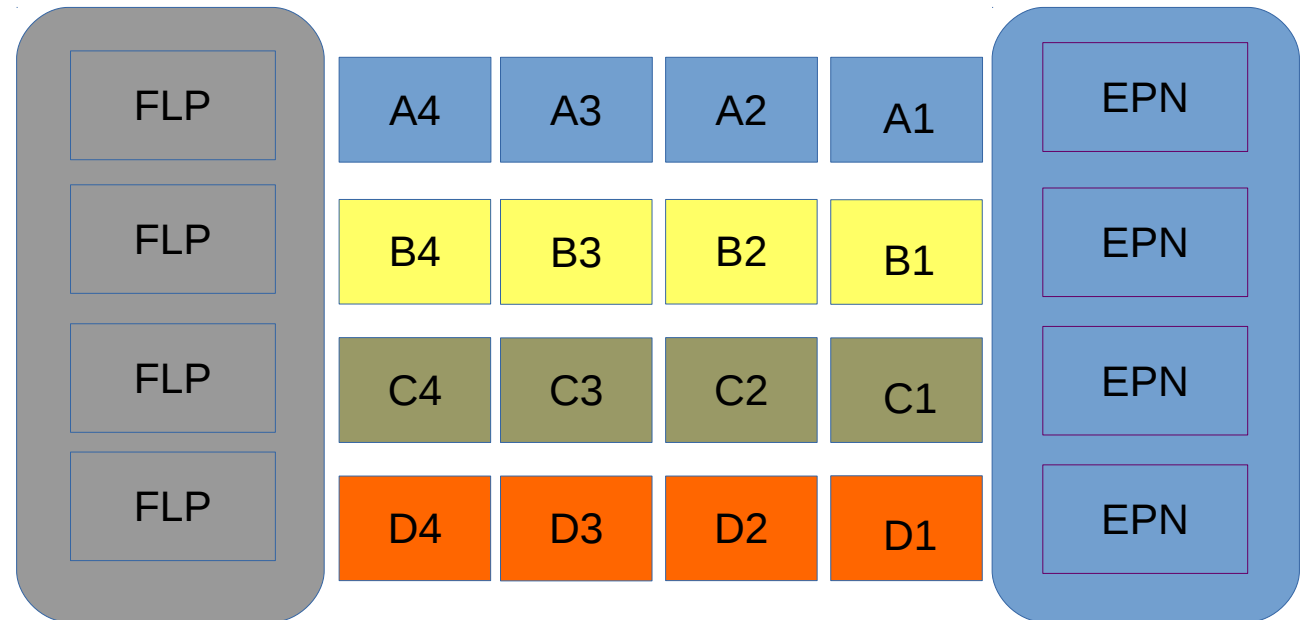
- To avoid network congestion, frames are cached on FLPs





Development of the O2 prototype

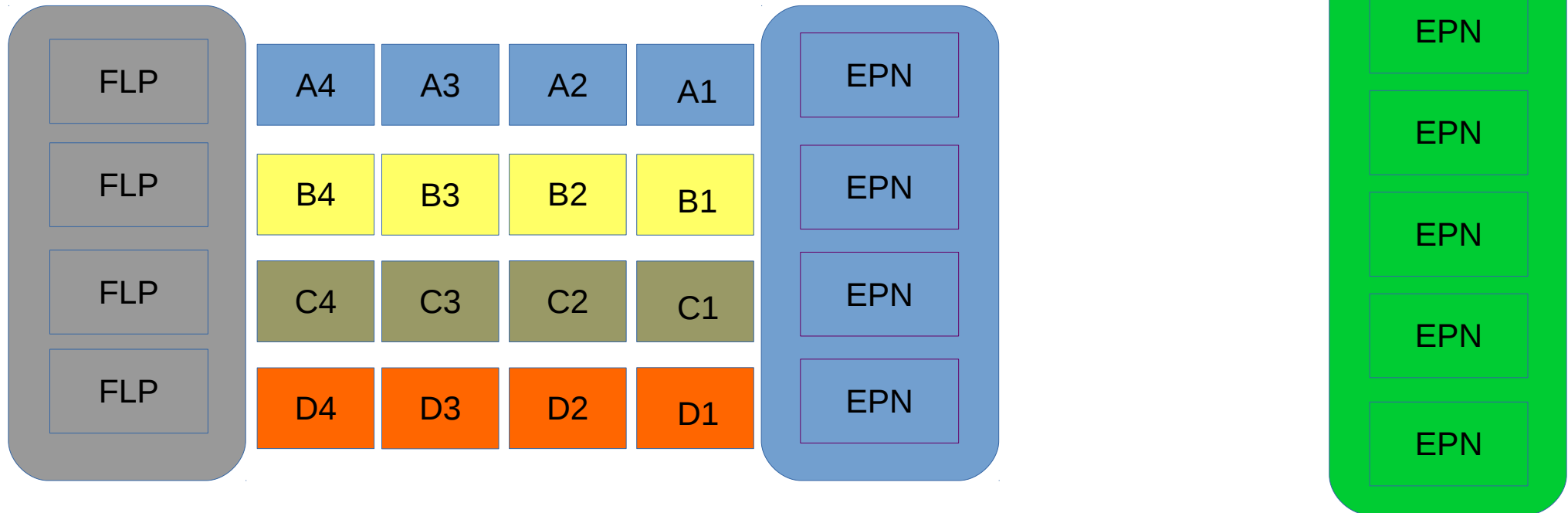
- To avoid network congestion, frames are cached on FLPs





Development of the O2 prototype

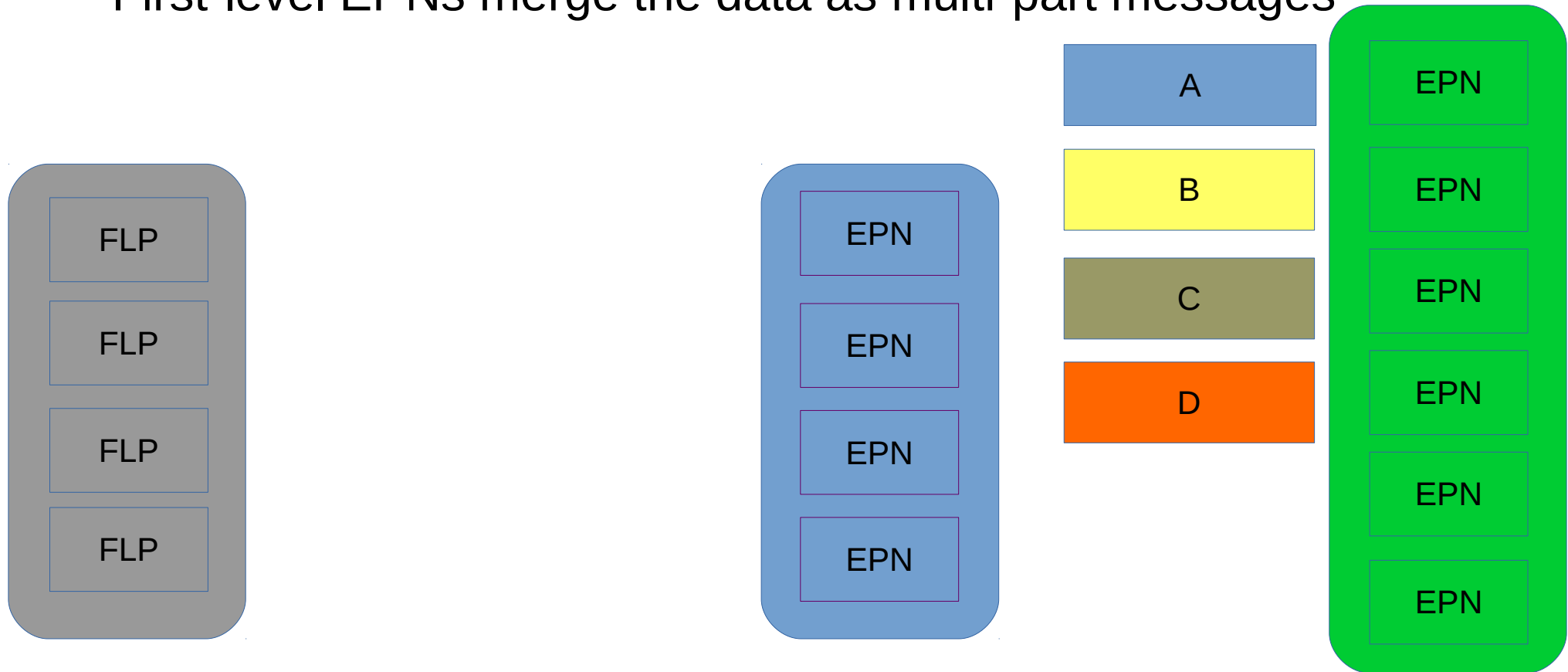
- Based on frame header, EPNs know when the last packet received
- First-level EPNs merge the data as multi-part messages





Development of the O2 prototype

- Based on frame header, EPNs know when the last packet received
- First-level EPNs merge the data as multi-part messages





Development of the O2 prototype

- What happens if a first-level EPN dies?
 - A temporary data loss
- What happens if a second-level EPN dies?

