

Запрос: Gossip Request

- requester gossip version
- requester low gossip version
- requester clusterDataVersion

→

Ответ: Gossip Response

→ A) responder GV $\boxed{\text{not strictly } <}$ requester low GV

- Responder needs to reload Transaction log
(hope for version $\boxed{\text{strictly } \geq}$ requester low GV)

When reloading Transaction log,
only strictly outdated part
of HashGraph needs to be
purged from RAM

→ B) requester GV $\boxed{\text{not strictly } <}$ responder low GV

- Requester needs to reload Transaction log
(hope for version $\boxed{\text{strictly } \geq}$ responder low GV)

C) responder clusterDV $\boxed{\text{strictly } \geq}$ requester clusterDV

- Responder doesn't need updates

D) responder clusterDV $\boxed{\text{not strictly } <}$ requester clusterDV

- Responder needs gossip updates

(since responder gossip version to requester gossip version)

←

Payload: Gossip Update (D only)

- Send gossip Updates (list of Events) to Responder
 - since responder gossip version to requester gossip version

→

Post Gossip Update: update responder's gossip version
update responder's clusterDataVersion


1) To update gossip version

- find new top gossip graph version,
combining top gossip versions for each Node

2) To update multi-version / cluster version

- find new top data graph version,
combining top data versions for each Node
- traverse graph to find assumed / virtual versions for each new
top data events (BFS?)
 - if reached the end of graph - version didn't change

3) Round commencing

- 
- round is commenced $N > 5$ rounds after last event was decided
 - once a round is commenced, we remove it from RAM
 - round can be partially recreated in split brain situation,
but it will commence all the same - since its events will be
decided at a later round in this case

4) Startup flow

- We call gossip with our version
 - If our gossip version is behind we're asked to load
transaction log first. After that, we use transaction log
data version as our gossip version to join the gossip.