Lightweight Network Network Interface

Using Link Aggregation

Rev. 3

Norman Finn

nfinn@cisco.com

ENNI: Heavy or light?

- There are at least two distinct methods we can pursue for defining an Ether NNI:
- **Heavy**:

  A Buffer Network is built along the lines suggested in new-nfinn-buffer-networks-0310-v01.pdf with an explicit data encapsulation.

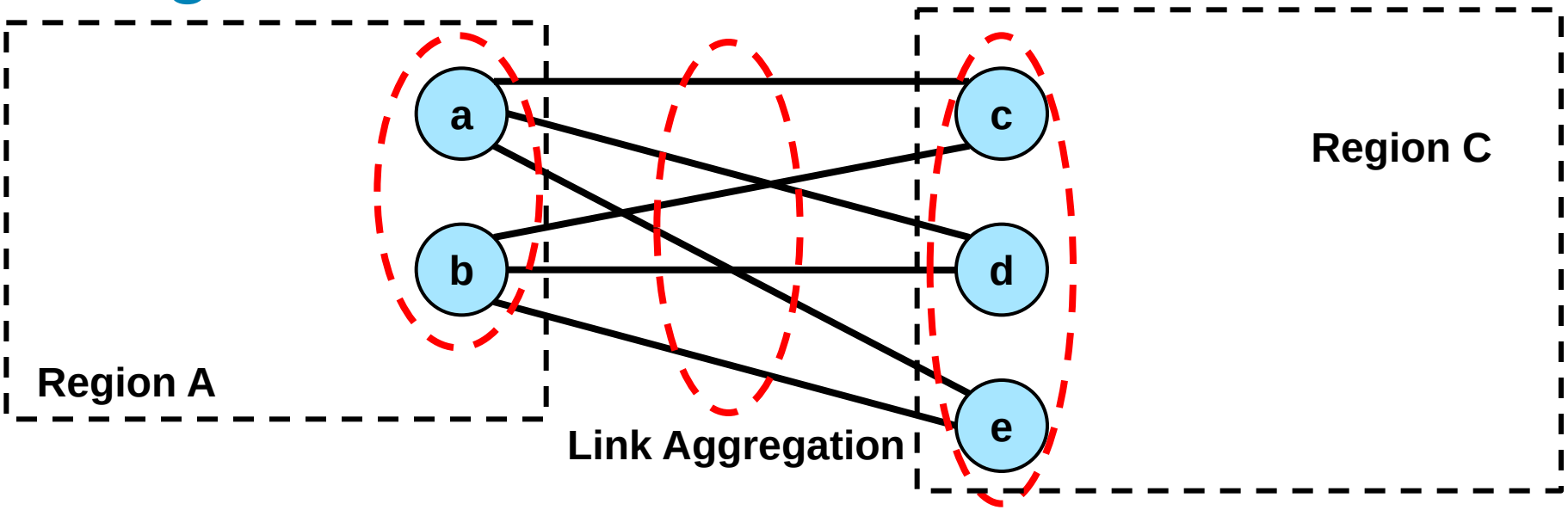
- **Light**:

  Buffer Network is built using “virtual nodes,” i.e. the multiple physical Nodes of each Portal cooperate to give the appearance of a Portal consisting of a single Node. This present document is new-nfinn-light-nni-0710-v03.pdf.

- Each method has its advantages and drawbacks, and all of the drawbacks can be addressed. This is a classic engineering decision.

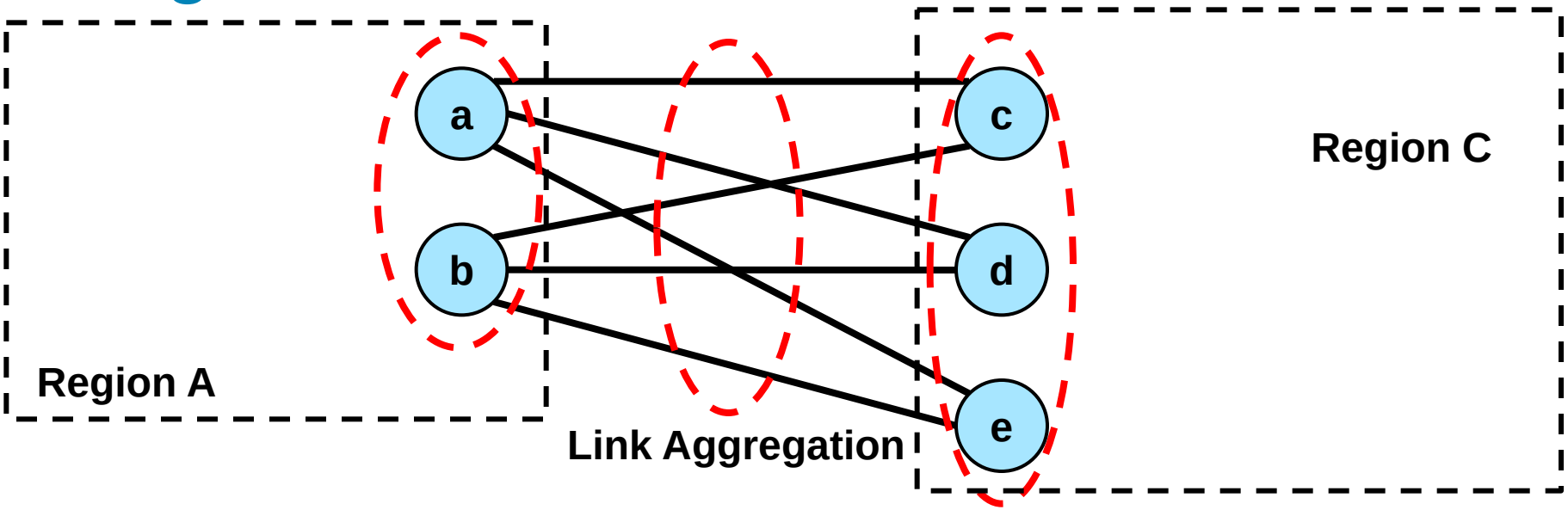
Light ENNI

Light ENNI



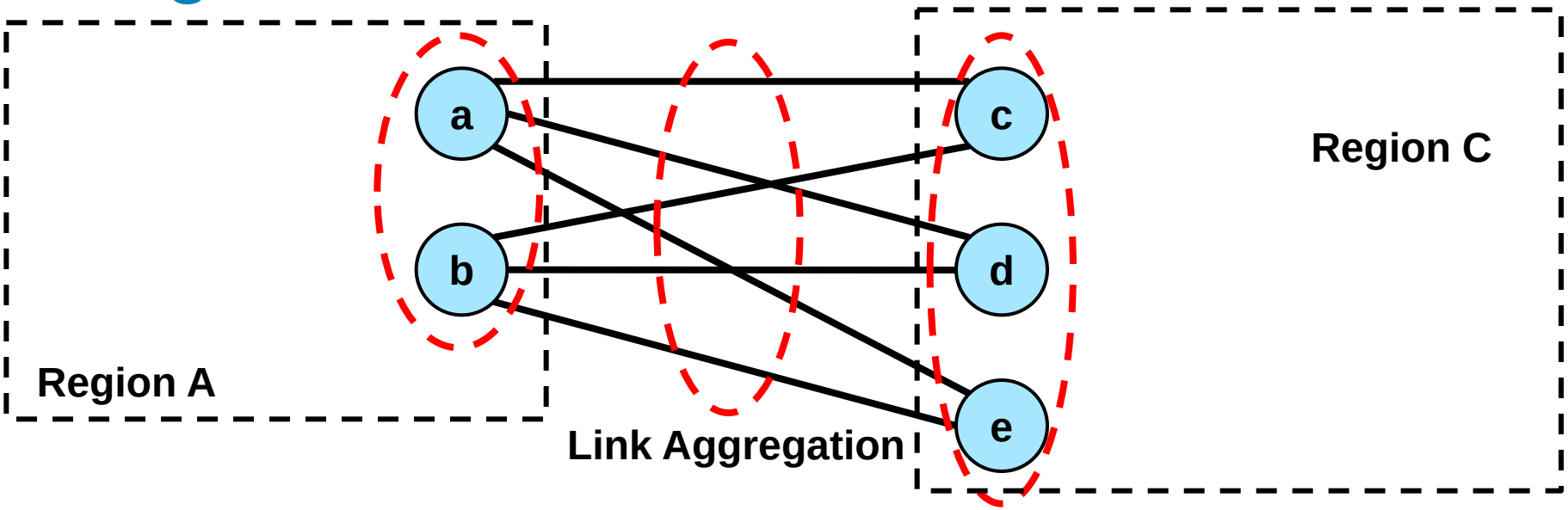
- The Terminal Nodes in each Region appear, to the other Region, to be a single Terminal Node (bridge, switch, or whatever).
- All of the inter-Region Links are combined into a single Aggregated Link using LACP.
- Links among Nodes in the same Region are invisible and irrelevant to the ENNI.

Light ENNI



- The means by which the Virtual Terminal Nodes are implemented does not need to be standardized; this author sees no requirement for **c**, **d**, and **e** to come from three different vendors.
- The choice of physical link is **always** up to the transmitting Virtual Terminal Node, and the receiving Virtual Terminal Node **must** live with the choice.

Light ENNI



- Physical level CFM can be used to improved failure detection time for the physical links; we do not have to depend on LACP's (slow) timeouts.
- Obviously, the two Regions have to agree on a data encapsulation, but a 1:1 service encapsulation translation can be performed at either (or both) ends, and **no encapsulation-dependent CFM is required**.

Service Assignment across NNI

Service to Link assignment

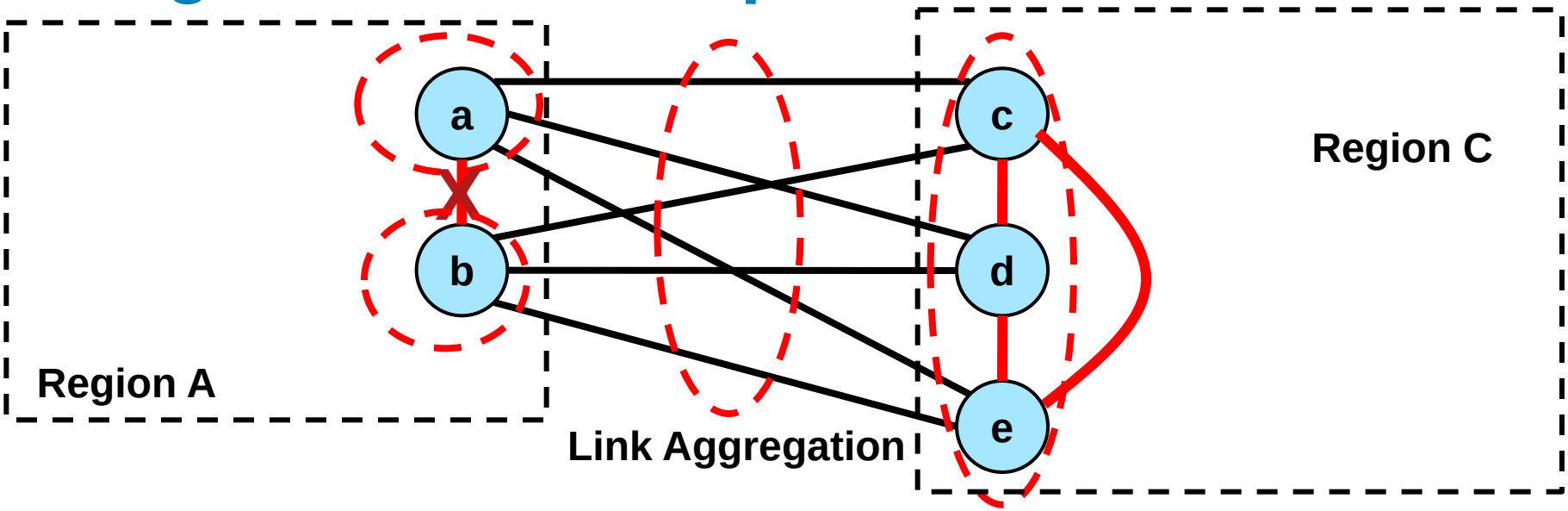
- We cannot express link preferences for thousands of services in an LACP or CCM PDU; some kind of “bundling” is necessary, **if** preferences are signaled explicitly.
- But, if we configure the list of Service-to-physical Link assignments for every possible combination of available physical Links, then no extra run-time protocol (other than configuration checksum comparison) is required across the NNI.
- This seems preferable to any dynamic algorithm, because it is amenable to human negotiation and judgment, and not subject to “priority assignment wars.”

Exchanging Link Assignment across NNI

- Just like MSTP, a Link Assignment Database has a “Link Assignment Identifier (LAID)” consisting of a name, a revision number, and a hash function, so that it is very unlikely to accidentally think you are in sync with your neighbor when you are not.
- Just like MSTP in BPDUs, the LAID is carried in every LACPDU.
- Perhaps, we carry both an old and a new LAID, so that a graceful transition can be made when the configuration changes. Details to be worked out.

Split brain detection = Graceful name change

Light ENNI – Intra-pair Link failure



- If the **Link** between two components of a Virtual Terminal Node (e.g. **a**-**b**) fails, both components can takeover the Node's identity, but act independently (the "**split brain**" scenario), with disastrous results.
- For this reason, "inter-VTN links" are made extra-reliable, and in some existing proprietary implementations, are assumed to be failure-proof.

Light ENNI – Split brain detection

- We cannot (in the author's opinion) design a network standard around "failure proof links".
- Since we are assuming that LACP is being used to establish Aggregated Links between Virtual Terminal Nodes, we could **enhance LACP** so that the devices connected to a Virtual Terminal Node can assist the VTN in detecting a "split brain" scenario.
- But, split brain detection is necessarily a hippity-hippity-hop operation, involving multiple Nodes; there is no equivalent to the Maintenance Associations described for the Heavy ENNI. Split brain detection may be **slower** than MA failure detection.

Light ENNI – Split brain recovery

- Recovery from the split brain is up to the implementation:

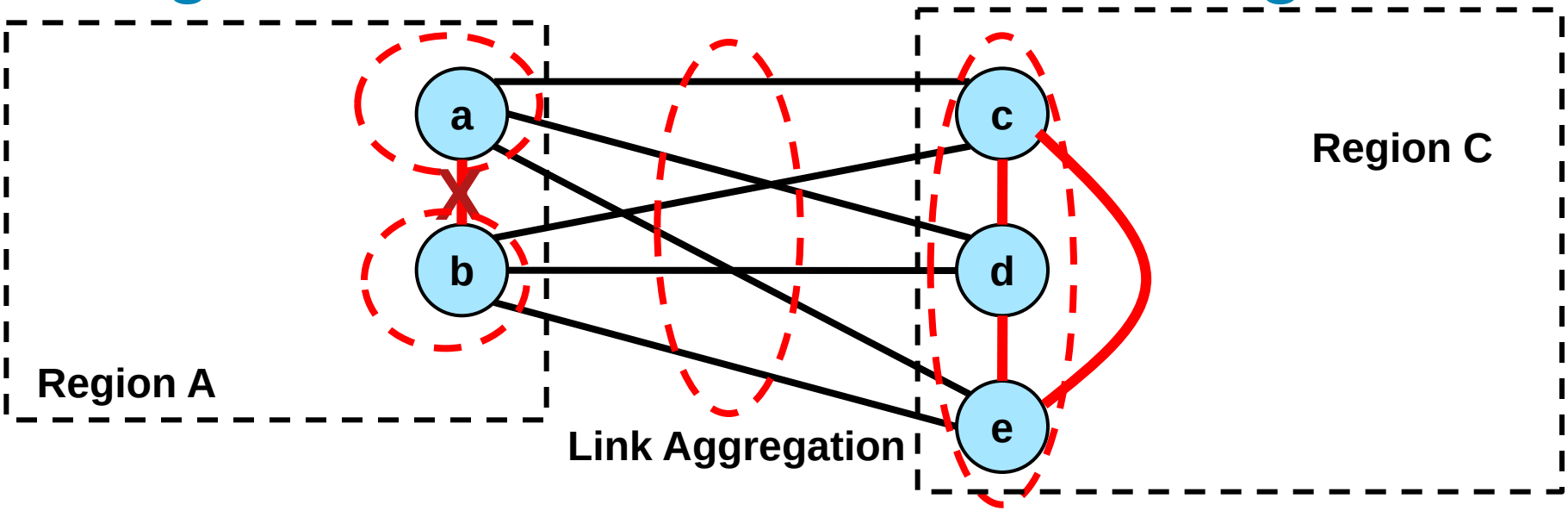
  Some implementations may have no issues with a split brain.

  Some implementations may shut down an isolated secondary component of the virtual node.

  Some implementations may change identities to become two separate devices (equivalent to shut down for the ENNI, since the “light” scheme requires a single virtual node).

- Signaling the recovery choice can be handled with current LACP, e.g., by removing Links to one of the physical Nodes from the aggregation.

Light ENNI – Graceful name change



- Let us suppose that, if the a-b Link fails, then device **a** continues to use the virtual device name as its Actor_System field, but device **b** changes its name to a new Actor_System field based on its own physical ID.
- If either **a** or **b** really failed, then **c**, **d**, and **e** will continue to use the link to the remaining system.
- If only Link a-b failed, then **c**, **d**, and **e** will each pick **a**.

Light ENNI – Graceful name change

- All that is needed is:
  - A means for **c**, **d**, and **e** to not disrupt the aggregation while **b** changes its name.
  - Assurance that **c**, **d**, and **e** will all pick the same Node (**a** or **b**) when Link a-b fails.
- The first can be accommodated by adding an “Old Actor_System_Priority” and “Old Actor_System” TLV to LACP. This allows a system to change its name without disrupting an ongoing aggregation.
- The second can be done by requiring **c**, **d**, and **e** to select the link with the lower numerical Actor_System_Priority and Actor_System to continue with the NNI.

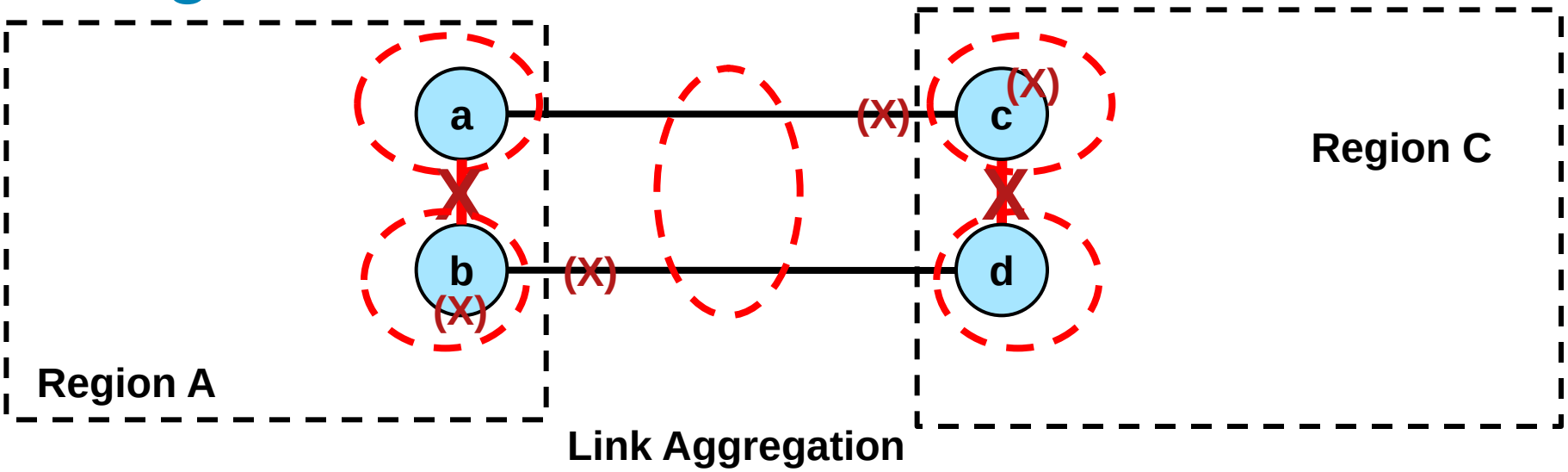
Light ENNI – Graceful name change

- Now, if Node **a** fails:
  - Nodes **c**, **d**, and **e**, all lose their Links to Node **a**, but continue to use the Links to Node **b**.
  - Node **b** changes its Actor_System name, but that causes no further disruption.
- If Node **b** fails:
  - Nodes **c**, **d**, and **e**, all lose their Links to Node **b**, but continue to use the Links to Node **a**.
- If Link a-b fails:
  - Node **b** changes its Actor_System name, and that causes Nodes **c**, **d**, and **e** to disaggregate from Node **b**.
  - Node **b** has no one to talk to.

Light ENNI – Graceful name change

- If Node **a** recovers:
  - Nodes **c**, **d**, and **e**, all switch over to using Node **a**.
  - Node **b** changes its Actor_System name to match Node **a**'s name, so all Links are back in use.
- If Node **b** recovers:
  - Nodes **c**, **d**, and **e**, regain their Links to Node **b**.
- If Link a-b recovers:
  - Node **b** changes its Actor_System name to match that of Node **a**.
  - Nodes **c**, **d**, and **e**, return their Links to Node **b** to the aggregation.

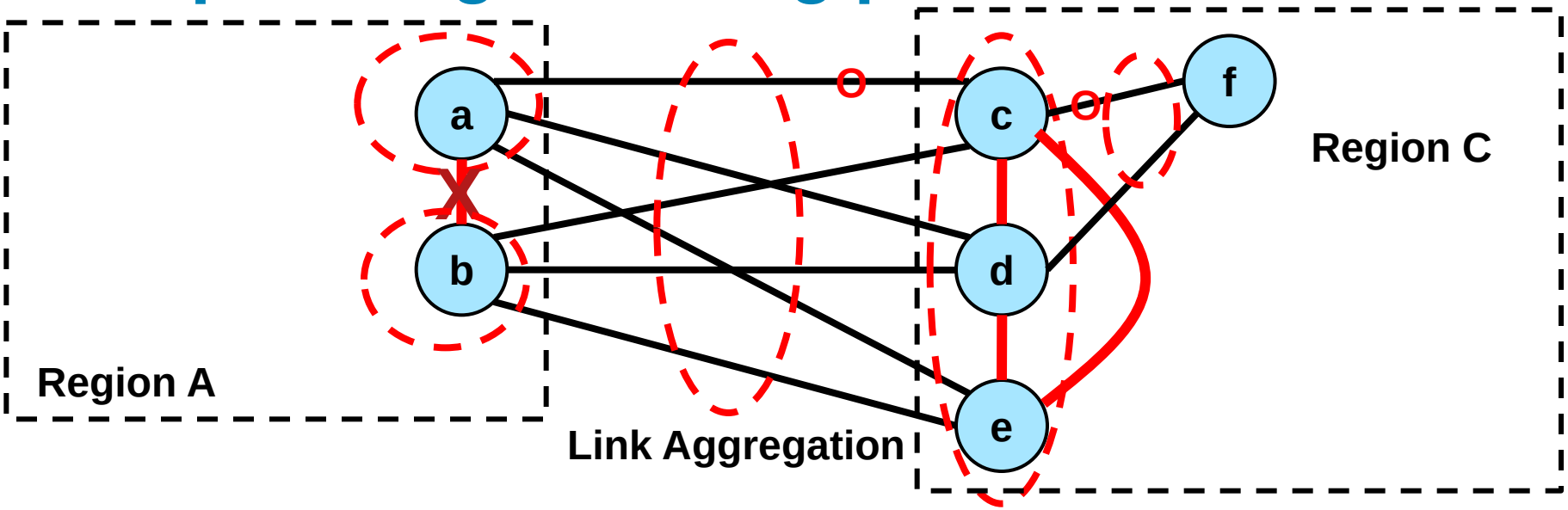
Light ENNI – bad choice of master brains



- Suppose the recovery method for “split brain” is that the secondary device shuts down or becomes unavailable.
- If **a** and **d**, above, are “master” nodes, then if both inter-VTN links fail (as shown), the ENNI would fail.
- Indicating in LACP which is the master node would enable the administrators to make **a** and **c** the master nodes, so that the **a—c** link would remain operational.

Bundling preferences

Expressing Bundling preferences



- Let us suppose that a particular Service uses Link **a**-**c**.
- It would be nice if **c** and **d** could tell f that they prefer that this Service use Link **c**-**f**.
- But, we would not want **f**'s preferences to drive the NNI, which drives Region A, which drives …
- Further thought is required on this subject.

NNI criteria list

How the NNI via LACP fares with criteria

All of the criteria are met, though some of them simply become requirements on the multi-chassis Bridge implementation. Some requirements need comment.

- Protect a single service (VLAN) or a group of services (VLAN).
  - We add Service-to-Link assignment configuration to LACPDU.
- Support interconnection between different network types (e.g. CN-PBN, PBN-PBN, PBN-PBBN, PBBN-PBBN, etc.)
  - This method works even for MPLS!
- Provide sub-50 ms fault recovery
  - Probably. We must look at the Split Brain situation further.

How the NNI via LACP fares with criteria

- The effects of protection events in one network must not affect other networks.
  - This requirement is placed on the distributed bridge implementation, in one sense. In another sense, the multi-chassis bridge is required to accept a service on any Link.
- Design the interconnected zone in a way that will ensure determinism and predictability.
  - LACP works this way, now.
- It must be possible to ensure the use of the same link in both directions for every service.
  - This why we are introducing Service-to-Link assignment configuration.

How the NNI via LACP fares with criteria

- The NNI protects services, not parts of services.
  - We must make this type of Aggregation mandatory.
- If one service provider cloud becomes split into multiple disjoint clouds, it cannot depend on the interconnect cloud or any adjacent service provider cloud to provide connectivity among its parts.
  - This is inherent in LACP – a frame transmitted over the aggregated Link cannot be returned, any more than it can be returned on a single physical Link.
- We cannot assume an ultra-reliable link.
  - This is why we are extending LACP for split-brain detection.

How the NNI via LACP fares with criteria

- Support inhomogeneous links -- not all the same speed or cost
  - Homogeneity is an arbitrary imposition, at present.
  - There is no need to specify how to dynamically split the services over unequal speed Links if the decision is mandated by configuration.
- Do we support an encapsulation scheme in the interconnect cloud, or is the ENNI independent of the encapsulation?
  - Independent!

How the NNI via LACP fares with criteria

- Do we assume that the bandwidth (or other Traffic Engineering parameter) of the interconnect cloud is adequate for all of the services, or do we do something special if it is insufficient?
  - If the Link usage is pre-configured, this is taken care of (some Services may be dropped in certain situations).
- Do we need protocol for conveying service creating/deletion or traffic engineering requirements between Service Providers?
  - Good question! To Be Determined. But if so, it should be part of a separate PAR in this author's opinion.