Cooperative Backup in Sparsely-Connected Dynamic Systems

D. Powell, L. Courtès, O. Hamouda, M. Kaâniche, M.-O. Killijian

LAAS-CNRS, Toulouse, France

IFIP 10.4 WG Workshop on
Dependability of Large-Scale and Dynamic Systems
Natal, Brazil, 22-23 February 2008

Cooperative Backup

- **Mobile devices are subject to damage, loss, theft...**
- **Typical data backup techniques…**
  - “synchronization” between mobile device and desktop machine
- **… are constraining or costly**
  - require access to desktop machine
  - potentially costly communication (e.g., GPRS, UMTS)

⇒ Backup opportunities are rare, data is at risk

Cooperative Backup

- **Key Ideas**
  - leverage computing device ubiquity
  - opportunistic replication to neighboring devices
  - free shortrange P2P communication (Wi-Fi, Bluetooth)
- **Salient Points**
  - adapted to intermittent connectivity scenarios
    - temporary backup on neighboring devices
    - final backup on reliable Internet store
  - continuous backup & replication
  - user credentials survive data/device loss or failure

Contributors

Internet store

Final backup

Data owner

Intermediate backup

Challenges

- **Backup availability**
  - participants may be malicious
  - participants may fail
- **Security and performance of intermediate backups**
  - participants may be malicious
  - unpredictable encounters and encounter durations
  - scarce resources (storage, energy)
- **Cooperation security**
  - participants may be malicious
  - participants may be selfish

Challenge 1 - Backup Availability

- **Issues**
  - participants may be malicious
  - participants may fail

☞ **need *replicated* intermediate backups**

- **Optimization goals**
  - storage efficiency…
  - … and data availability
- **Approach**
  - devise replication strategies
  - evaluate the efficiency/availability tradeoff

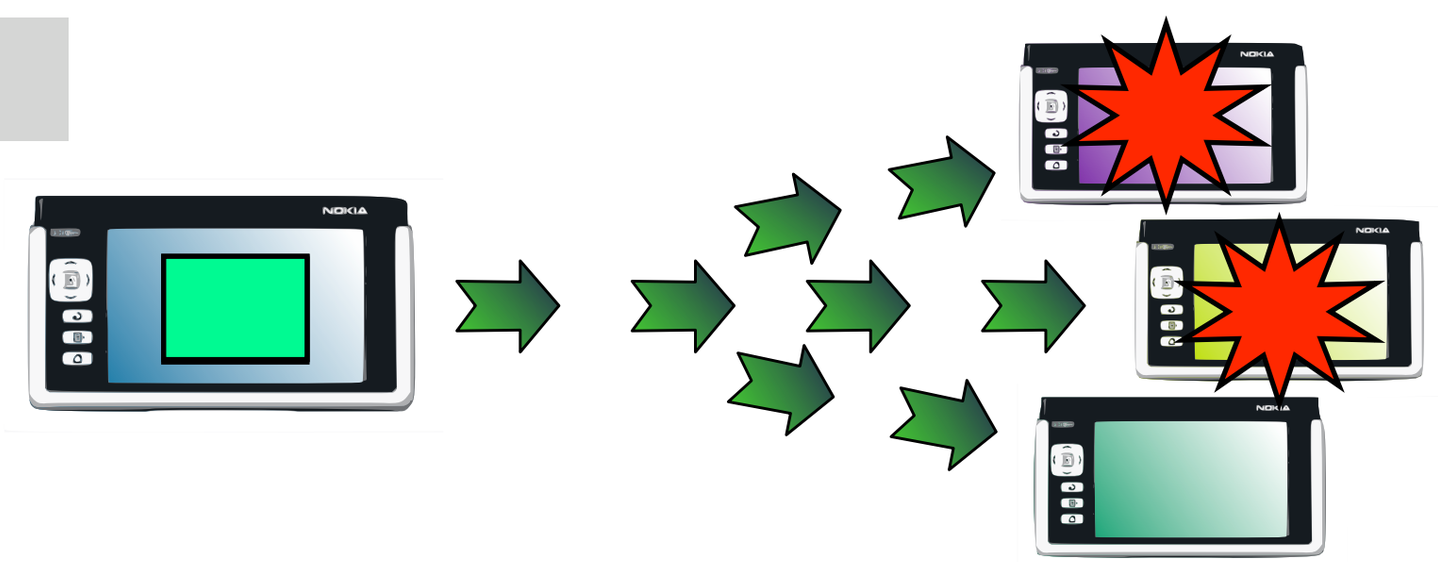
Simple Replication

- **Algorithm**
  - send a total of *n* copies of each data item
  - send 1 copy per contributor
  - recover from any 1 contributor out of *n*
- **Dependability & storage cost analysis**
  - tolerate *f* contributor faults ⇒ storage cost *f* + 1

Simple Replication

- **Algorithm**
  - send a total of *n* copies of each data item
  - send 1 copy per contributor
  - recover from any 1 contributor out of *n*
- **Dependability & storage cost analysis**
  - tolerate *f* contributor faults ⇒ storage cost *f* + 1

n=3 ; *f*=2



Erasure Codes

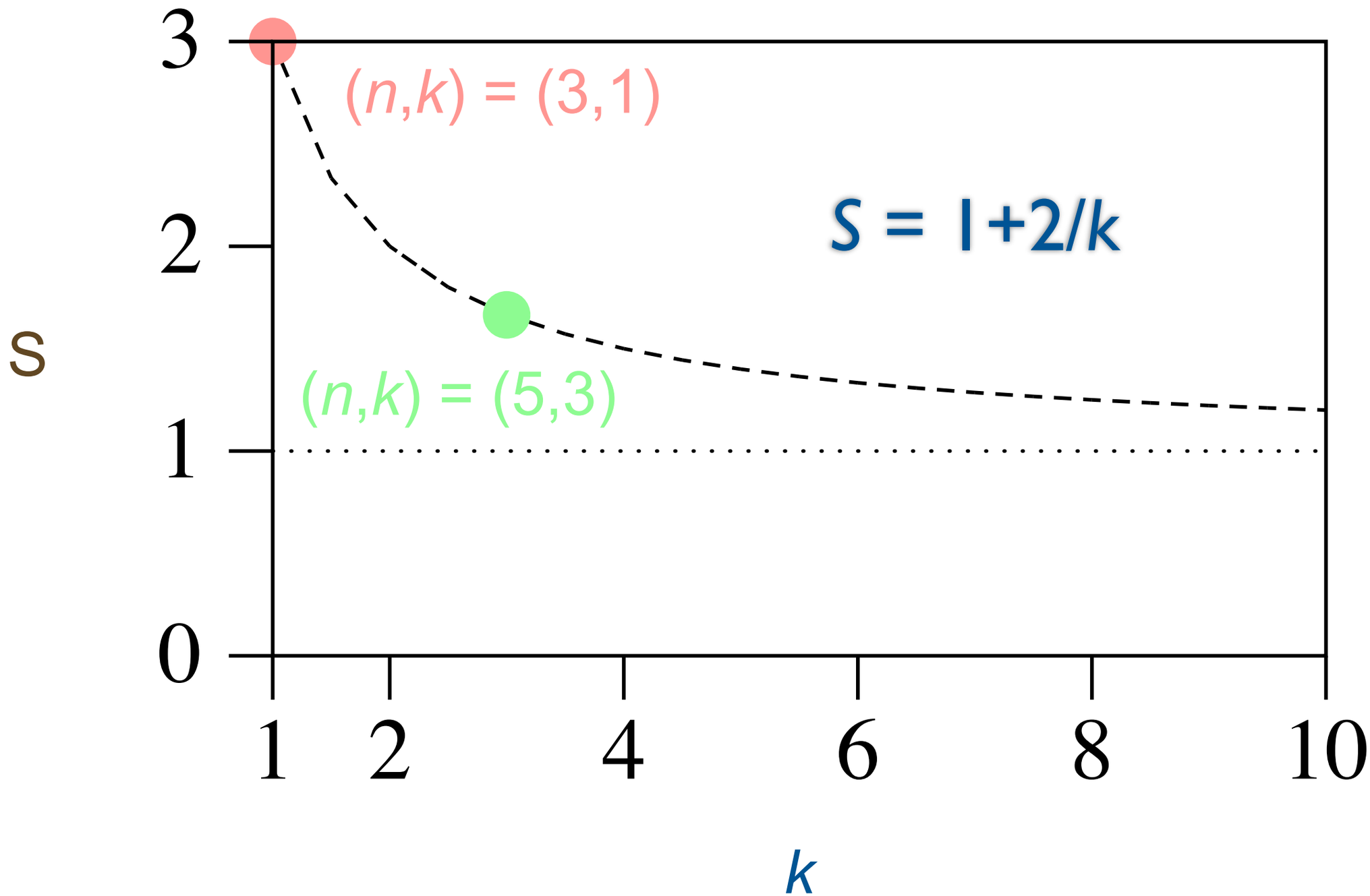
- **Basics**
  - *k*-block input ⟶ *n* coded blocks, $n > k$
  - *m* blocks suffice to recover input data $k \le m < n$
  - tolerate *n*-*m* faults
  - storage cost: $S = n/k$
- **Optimal erasure codes**
  - $m = k$
  - notation: (*n*,*k*) code
  - *n* and *k* are user-defined parameters
  - $k = 1 \Leftrightarrow$ simple replication

Erasure Codes

- **Algorithm**
  1. (n,k) erasure coding → n coded blocks
  2. send 1 coded block per contributor
  3. recover from any k contributors out of n
- **Dependability & storage cost analysis**
  - tolerate f contributor faults ⇒ storage cost = 1+f/k

Erasure Codes

Storage cost for f=2



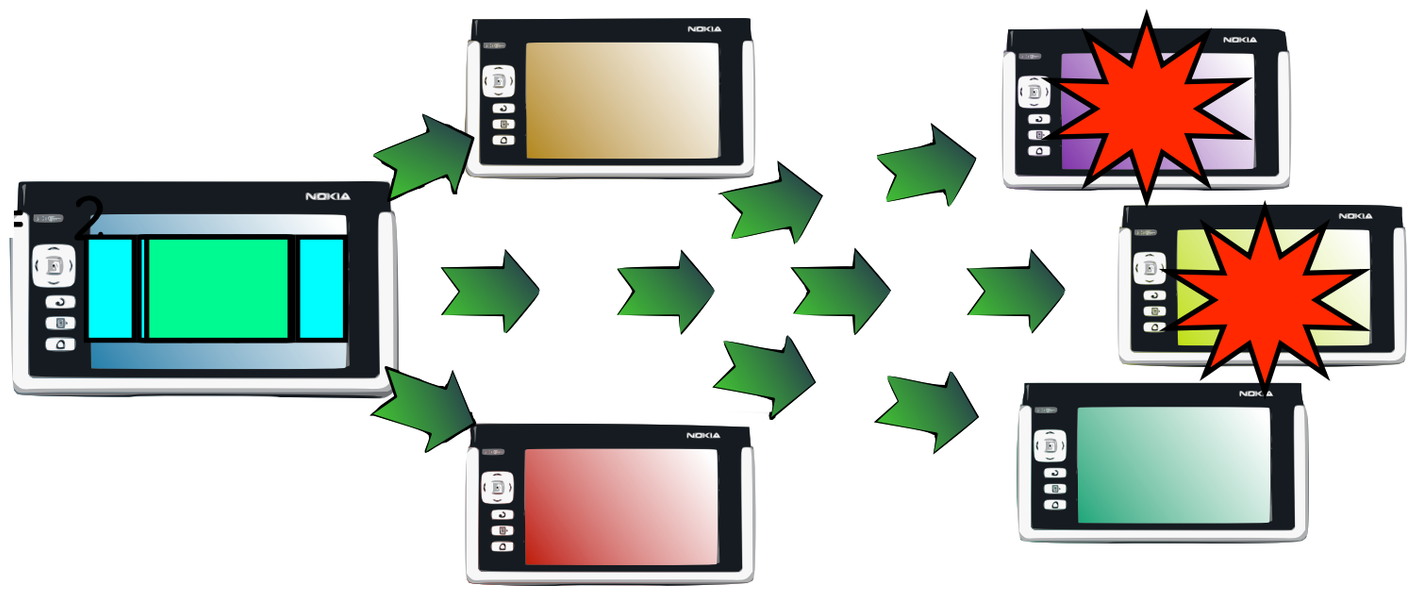
Erasure Codes

- **Algorithm**
  1. (n,k) erasure coding → n coded blocks
  2. send 1 coded block per contributor
  3. recover from any k contributors out of n
- **Dependability & storage cost analysis**
  - tolerate f contributor faults ⇒ storage cost = 1+f/k

Erasure Codes

- **Algorithm**
  1. (n,k) erasure coding → n coded blocks
  2. send 1 coded block per contributor
  3. recover from any k contributors out of n
- **Dependability & storage cost analysis**
  - tolerate f contributor faults ⇒ storage cost = $1+f/k$

(n,k) = (5,3)
$f = n - k = 2$



Dependability Model

- **Parameters**
  - erasure code characteristics (n,k)
  - stochastic processes with exponential distributions
    - device failure (rate λ)
    - encounter with other devices (rate α)

Time between encounters

1/α

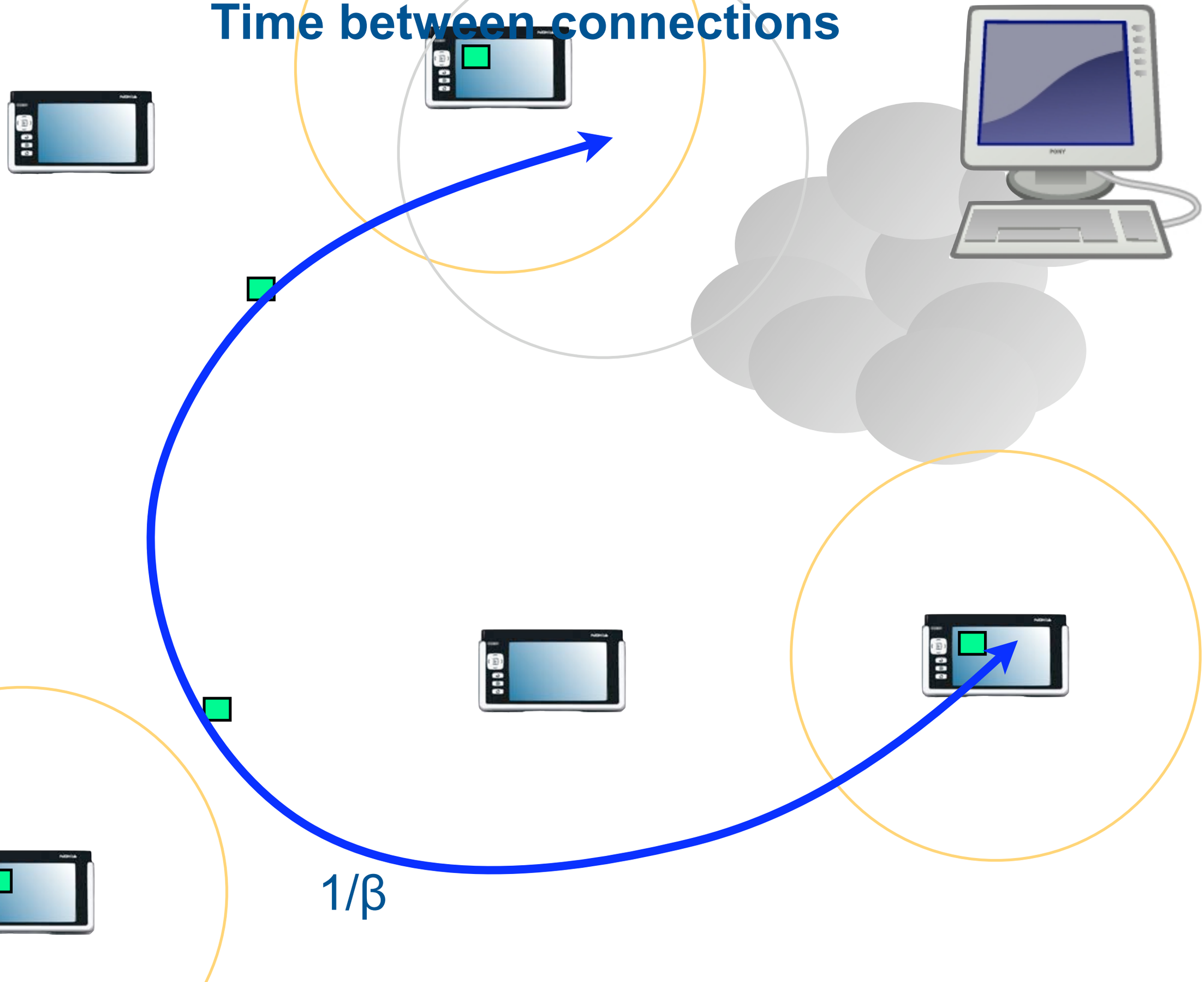
Dependability Model

- **Parameters**
  - erasure code characteristics (n,k)
  - stochastic processes with exponential distributions
    - device failure (rate λ)
    - encounter with other devices (rate α)

Dependability Model

- **Parameters**
  - erasure code characteristics (n,k)
  - stochastic processes with exponential distributions
    - device failure (rate λ)
    - encounter with other devices (rate α)
    - connection to Internet (rate β)

Time between connections

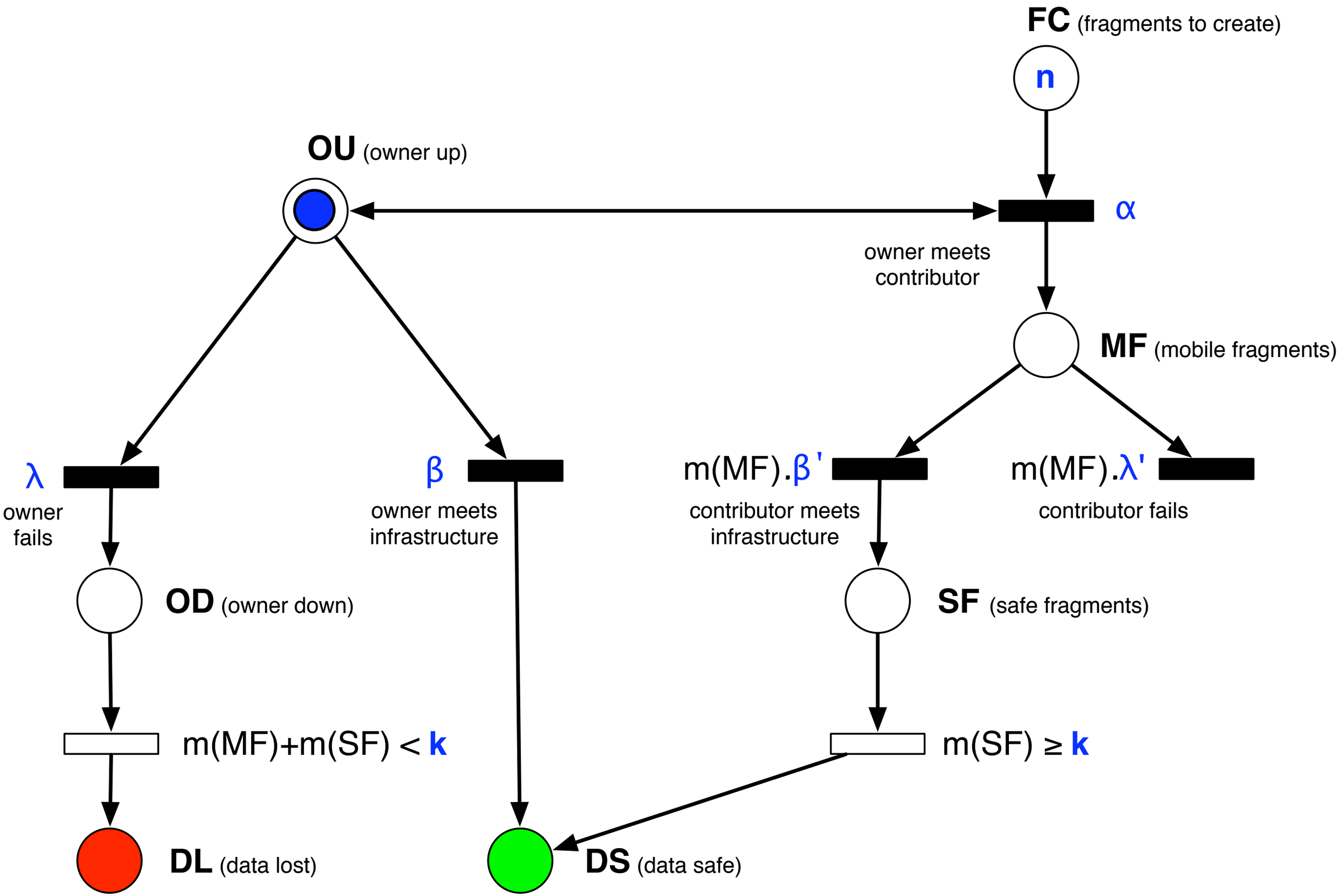


Dependability Model

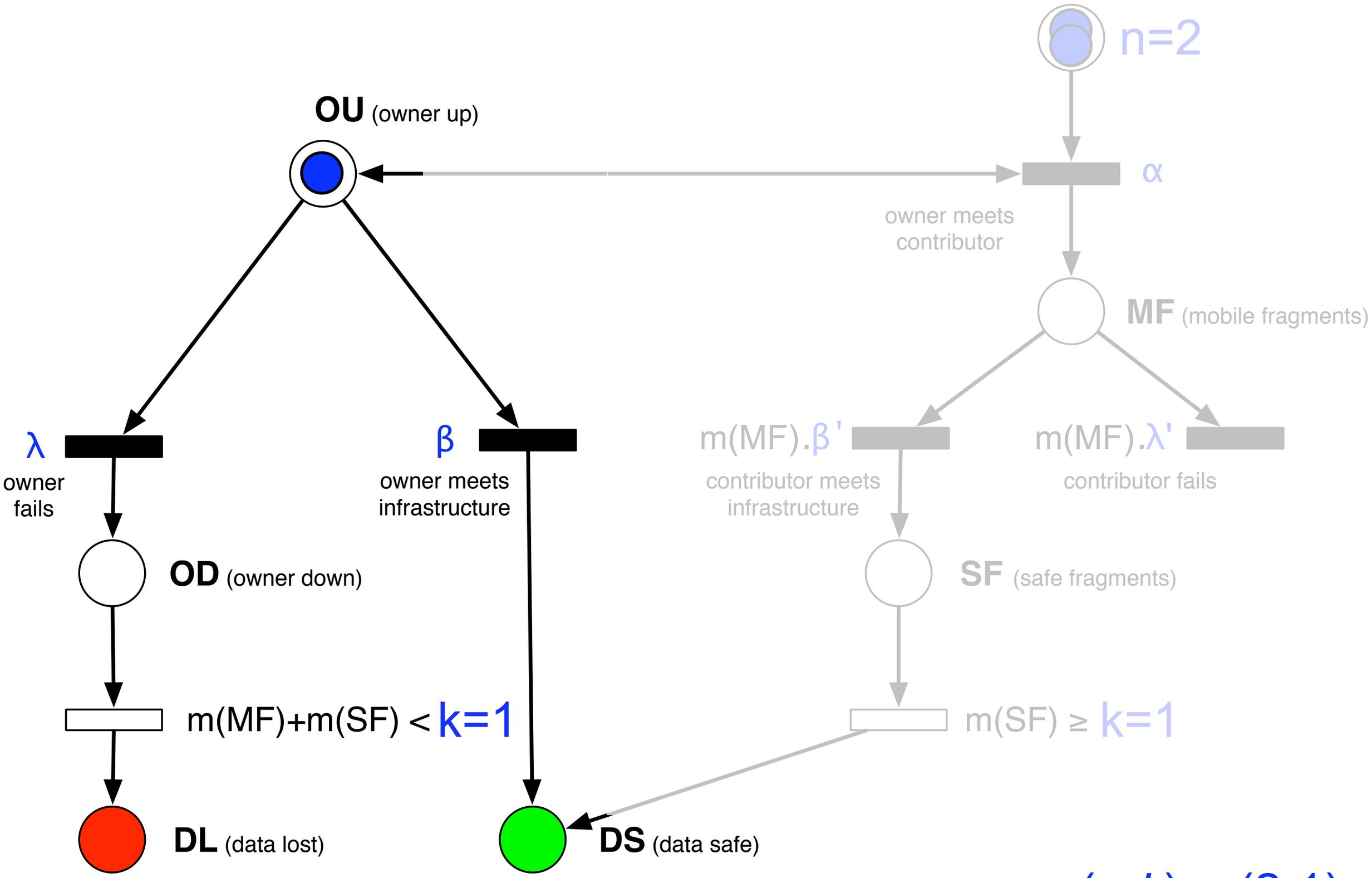
- **Parameters**
  - erasure code characteristics (n,k)
  - stochastic processes with exponential distributions
    - device failure (rate λ)
    - encounter with other devices (rate α)
    - connection to Internet (rate β)

Dependability Model

- **Parameters**
  - erasure code characteristics (n,k)
  - stochastic processes with exponential distributions
    - device failure (rate λ)
    - encounter with other devices (rate α)
    - connection to Internet (rate β)
- **Principle**
  - data safe ⇨ sufficient fragments have reached Internet store
  - data lost ⇨ data owner and contributors failed before sufficient fragments reached Internet store

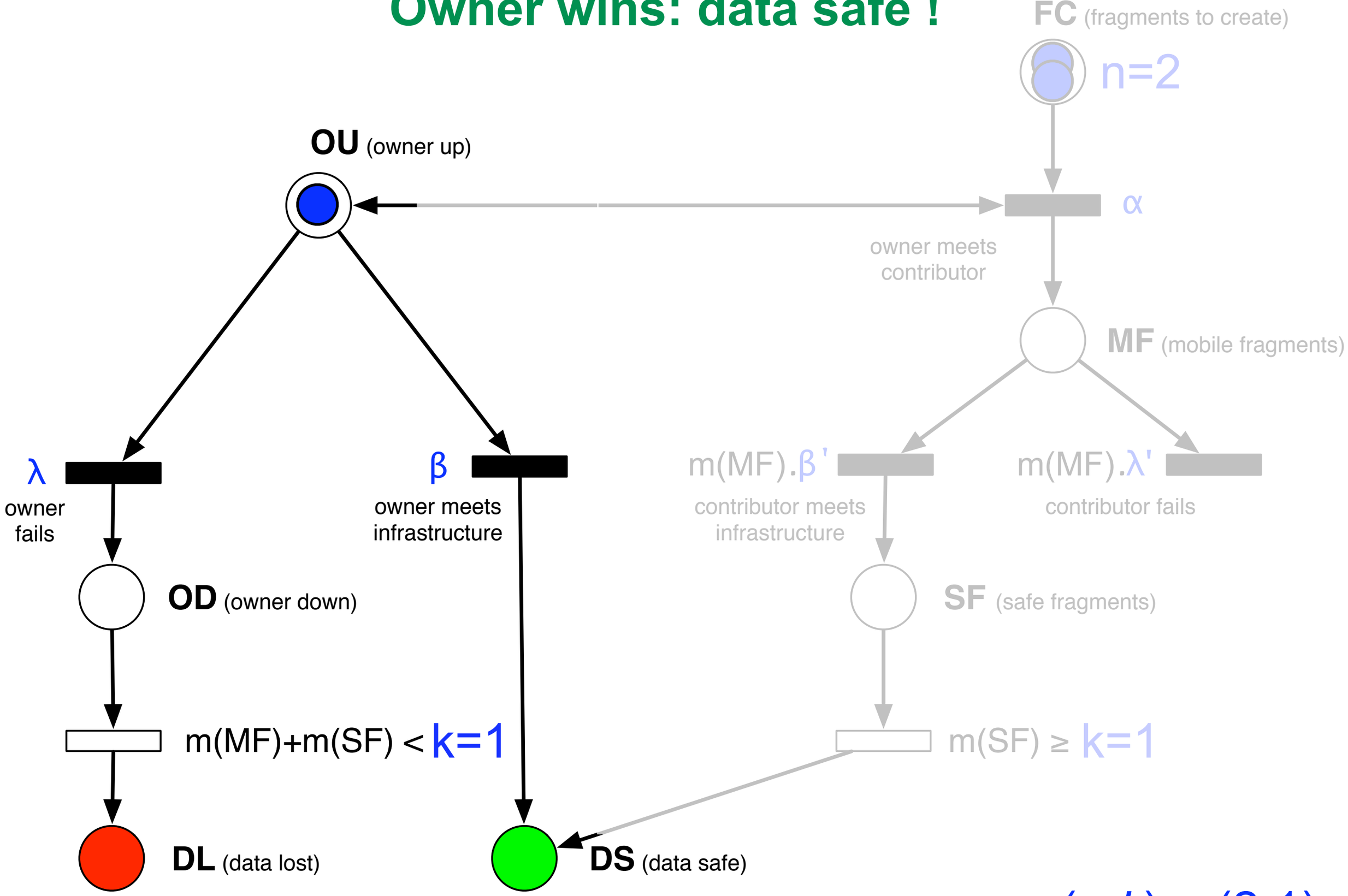

FC (fragments to create)
n
OU (owner up)
α
owner meets
contributor
MF (mobile fragments)
λ
owner
fails
β
owner meets
infrastructure
m(MF).β'
contributor meets
infrastructure
m(MF).λ'
contributor fails
OD (owner down)
SF (safe fragments)
m(MF)+m(SF) < k
m(SF) ≥ k
DL (data lost)
DS (data safe)

Game over: data lost !



$(n,k) = (2,1)$

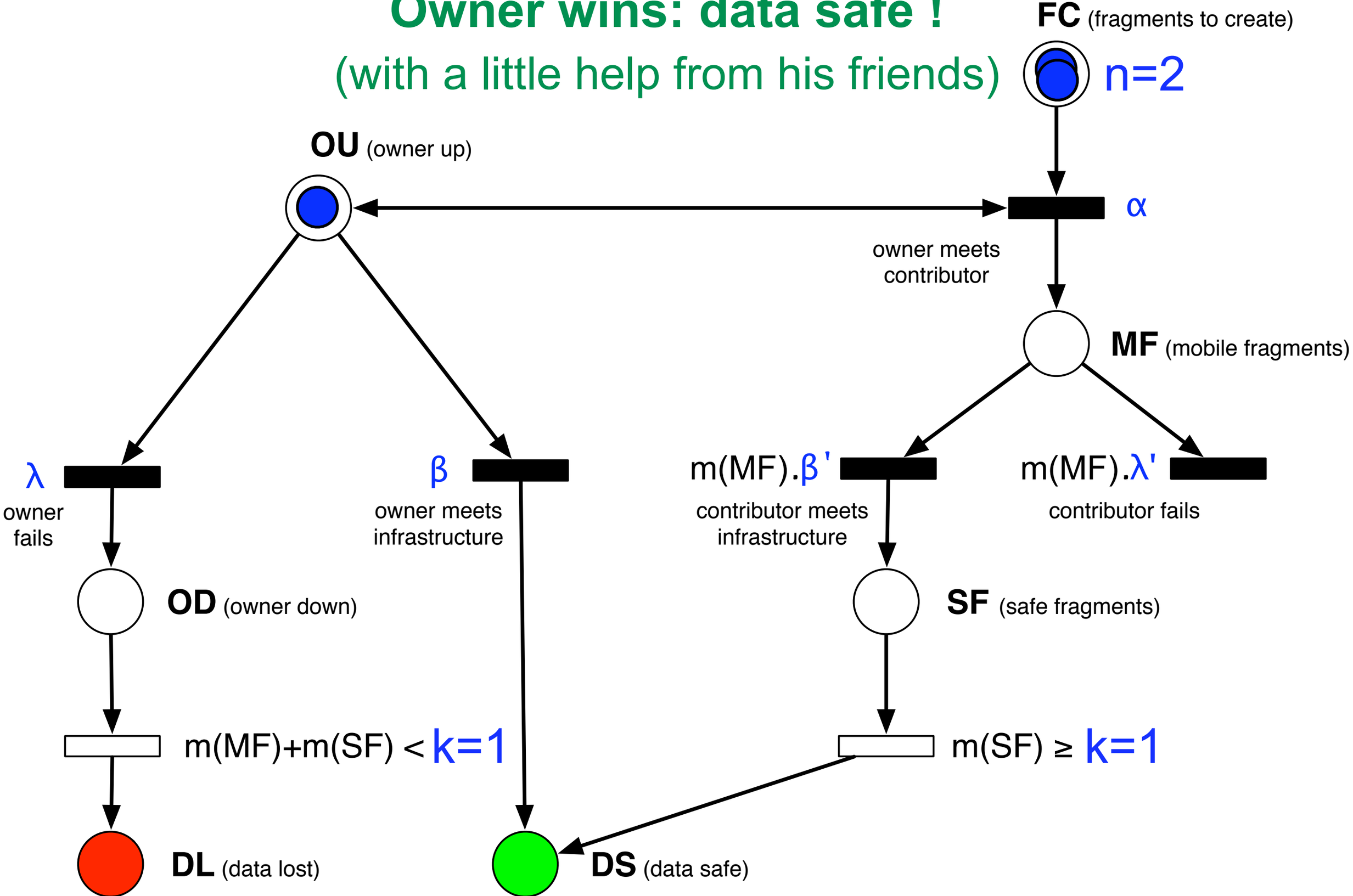
Owner wins: data safe !



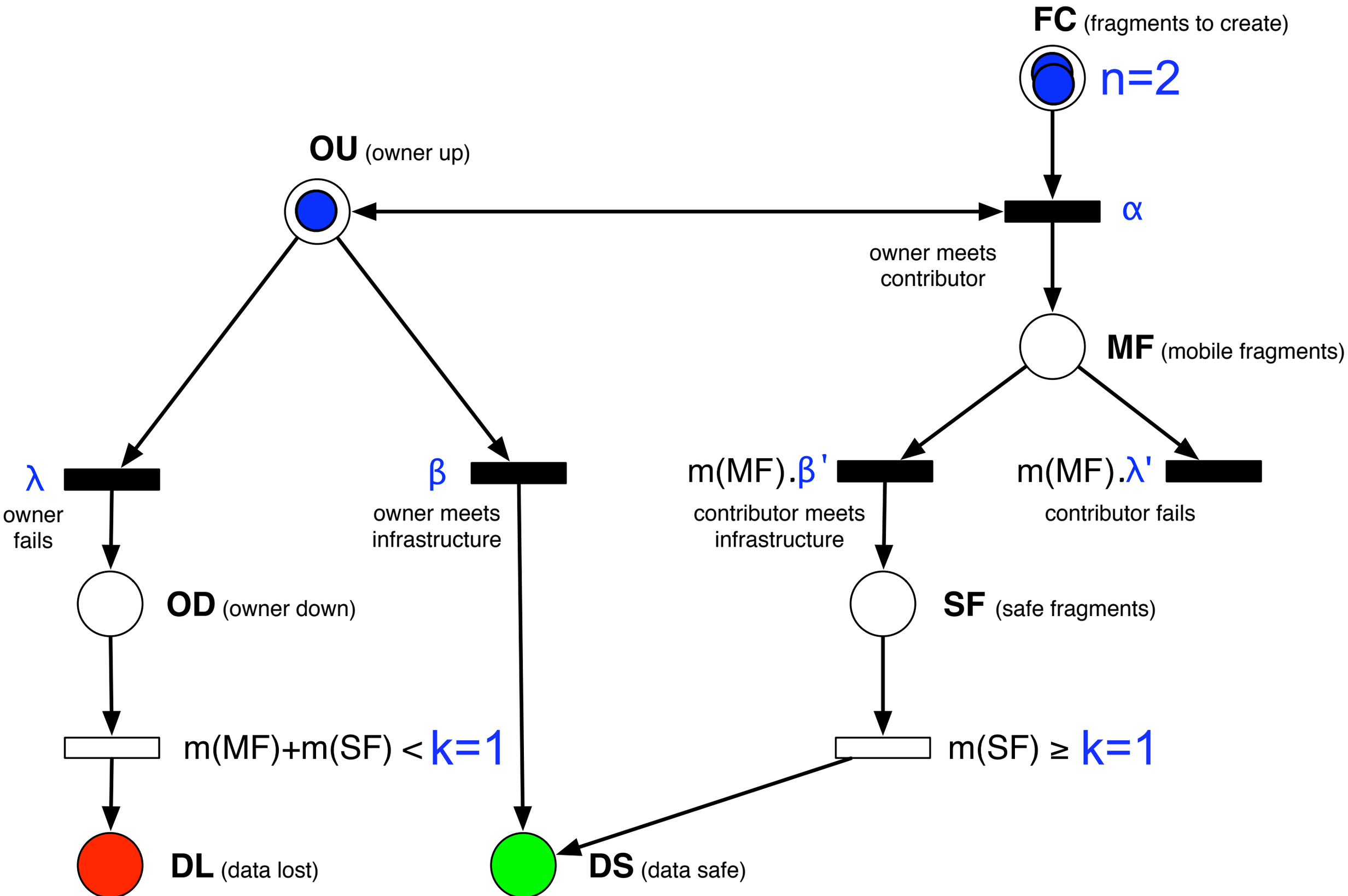
$(n,k) = (2,1)$

Owner wins: data safe !

(with a little help from his friends)



(*n*,*k*) = (2,1)


FC (fragments to create)
n=2
OU (owner up)
α
owner meets
contributor
MF (mobile fragments)
λ
owner
fails
β
owner meets
infrastructure
m(MF).β '
contributor meets
infrastructure
m(MF).λ'
contributor fails
OD (owner down)
SF (safe fragments)
m(MF)+m(SF) < k=1
m(SF) ≥ k=1
DL (data lost)
DS (data safe)
(n,k) = (2,1)

FC (fragments to create)

n=2

OU (owner up)

α

owner meets
contributor

MF (mobile fragments)

λ

owner
fails

β

owner meets
infrastructure

m(MF).β'

contributor meets
infrastructure

m(MF).λ'

contributor fails

OD (owner down)

SF (safe fragments)

m(MF)+m(SF) < k=1

m(SF) ≥ k=1

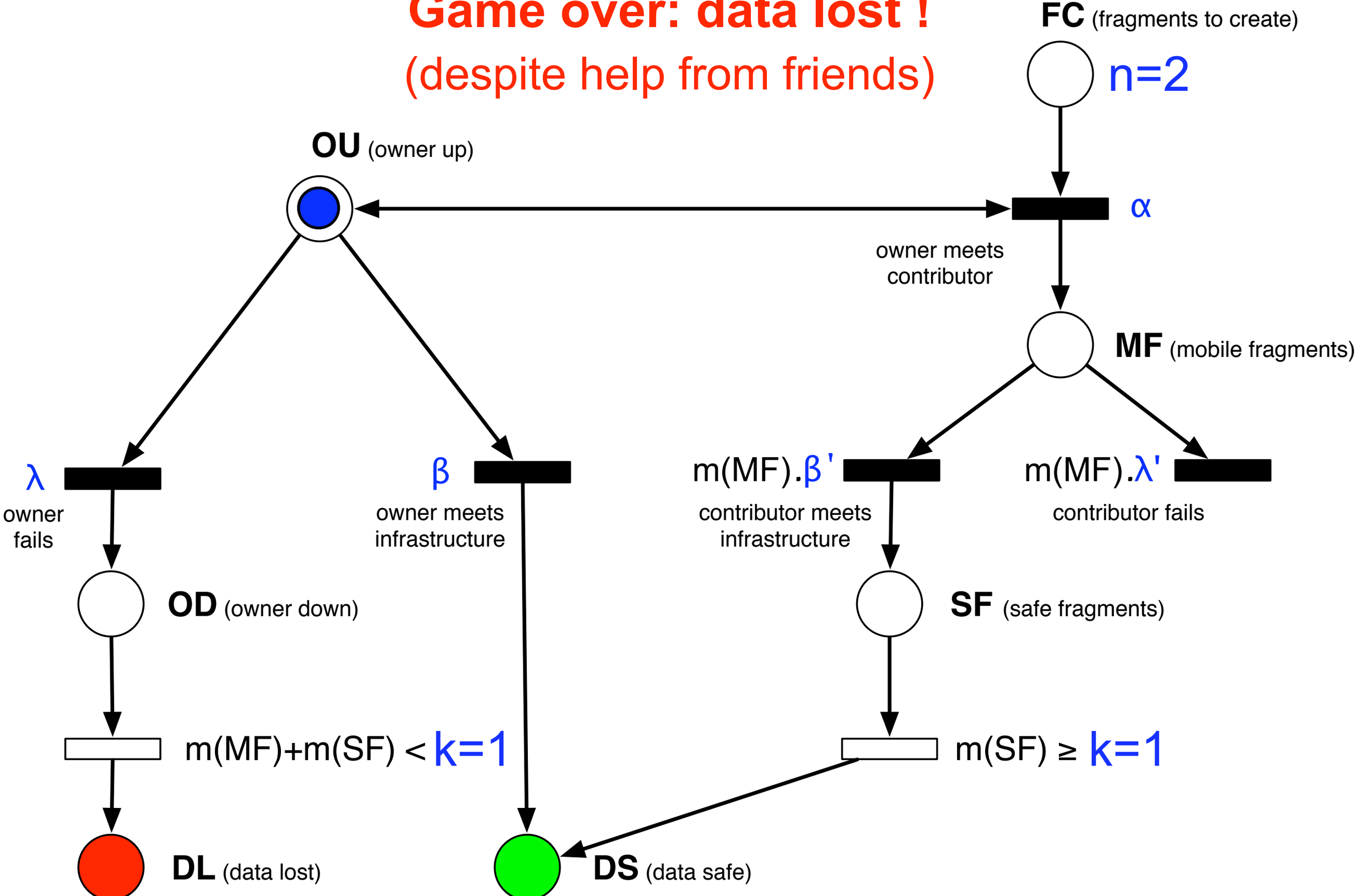
DL (data lost)

DS (data safe)

(*n*,*k*) = (2,1)

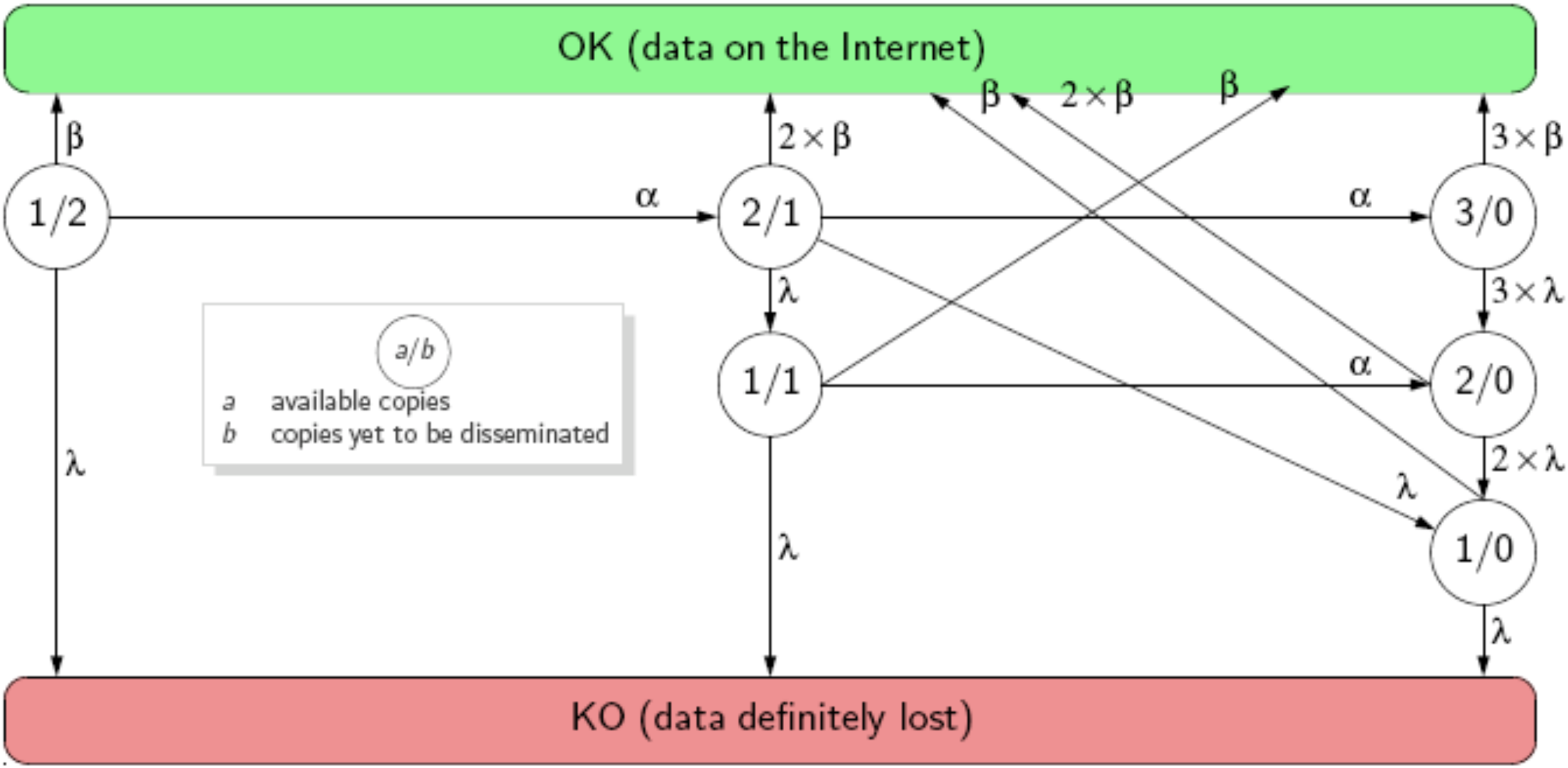
Game over: data lost !

(despite help from friends)

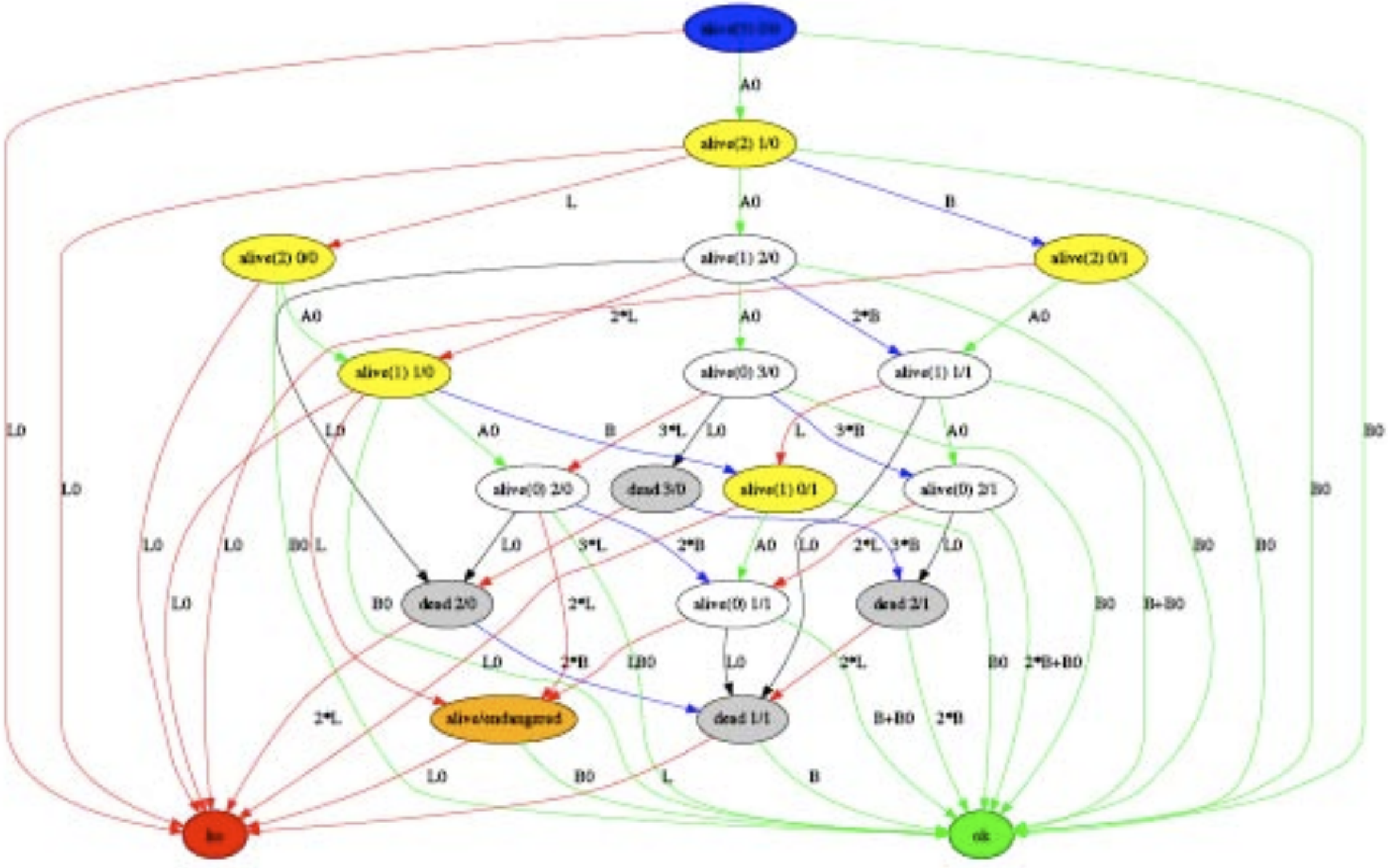


(*n*,*k*) = (2,1)

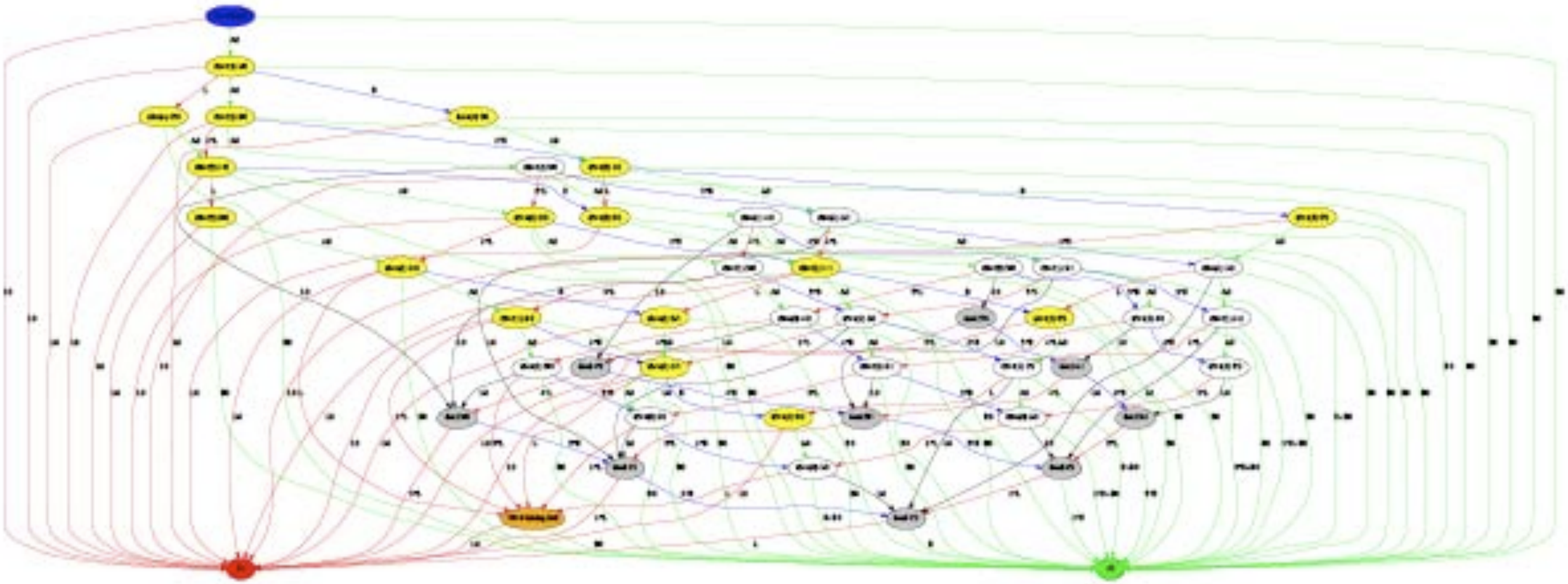
(*n*,*k*) = (2,1)


OK (data on the Internet)
KO (data definitely lost)
1/2
2/1
3/0
1/1
2/0
1/0
β
2×β
3×β
α
λ
2×λ
a/b
a available copies
b copies yet to be disseminated

$(n,k) = (3,2)$

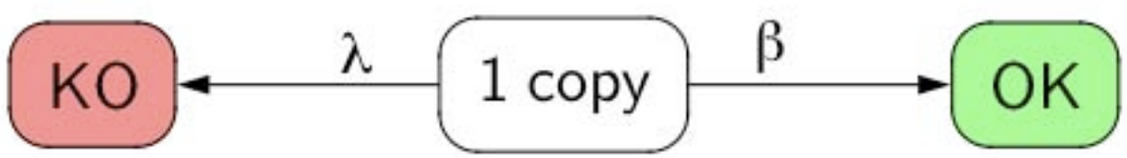

A0
alive(2) 1/0
L
A0
B
alive(2) 0/0
alive(1) 2/0
alive(2) 0/1
A0
2*L
A0
2*B
A0
alive(1) 1/0
alive(0) 3/0
alive(1) 1/1
L0
L0
A0
B
3*L
L0
L
3*B
A0
B0
alive(0) 2/0
dead 3/0
alive(1) 0/1
alive(0) 2/1
L0
L0
B0
L
L0
3*L
2*B
A0
L0
2*L
3*B
L0
B0
B0
L0
B0
dead 2/0
2*L
alive(0) 1/1
dead 2/1
B0
B+B0
L0
2*B
LB0
L0
2*L
B0
2*B+B0
2*L
alive+endangered
dead 1/1
B+B0
2*B
L0
B0
L
B
lw
nk

$(n,k) = (5,3)$



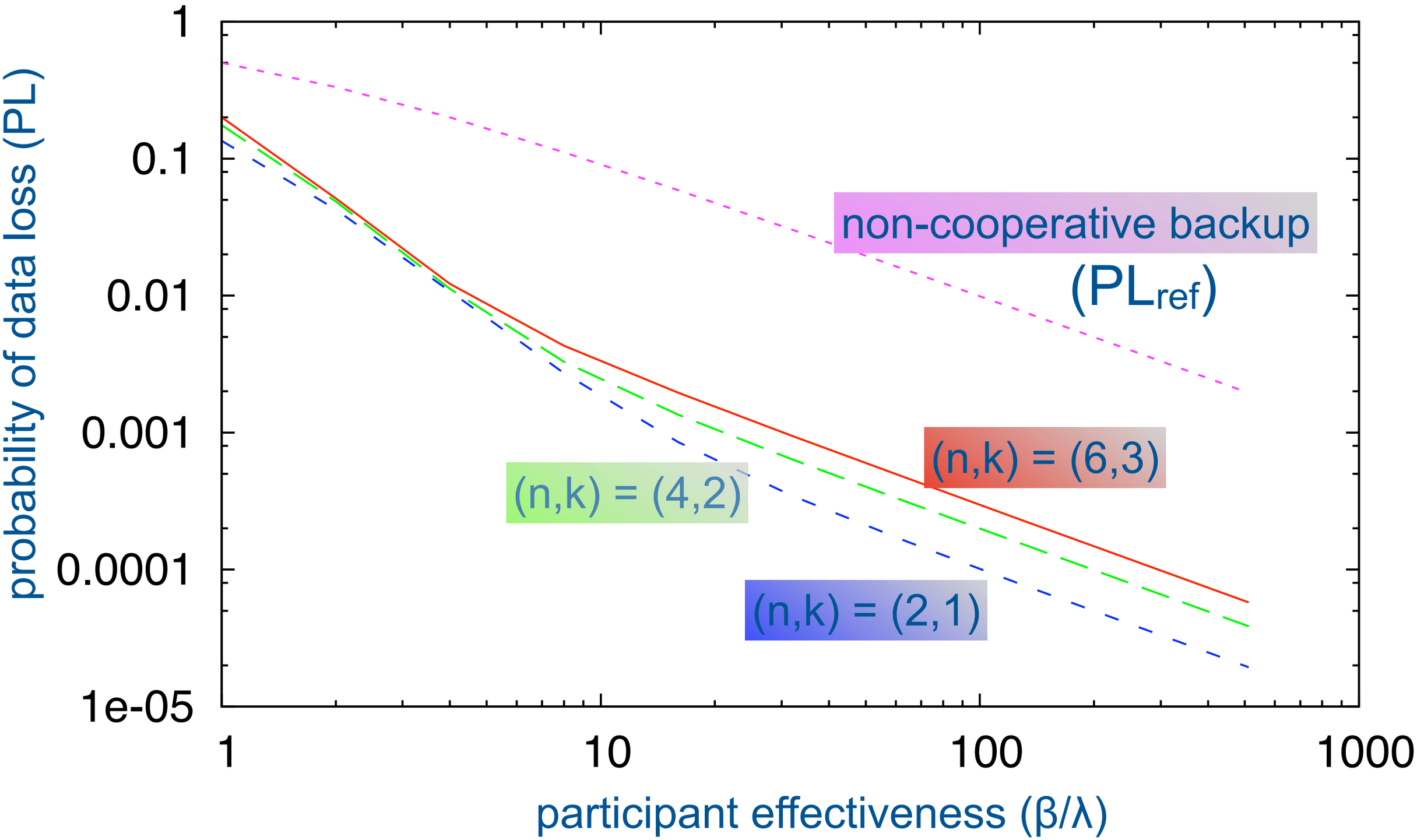
Dependability Measurements

- **PL: probability of data loss**
  - Probability of data owner and contributors failing before sufficient fragments have reached Internet store
- **LRF: data loss reduction factor**
  - PL compared to non-cooperative backup
    - $LRF = PL_{ref} / PL$
  - Non-cooperative backup
    - only one device ⇔ $\alpha = 0$
    - either fails or connects to the Internet
    - $PL_{ref} = \lambda / (\lambda + \beta)$



PL: Probability of data loss

(connectivity ratio α/β = 100)

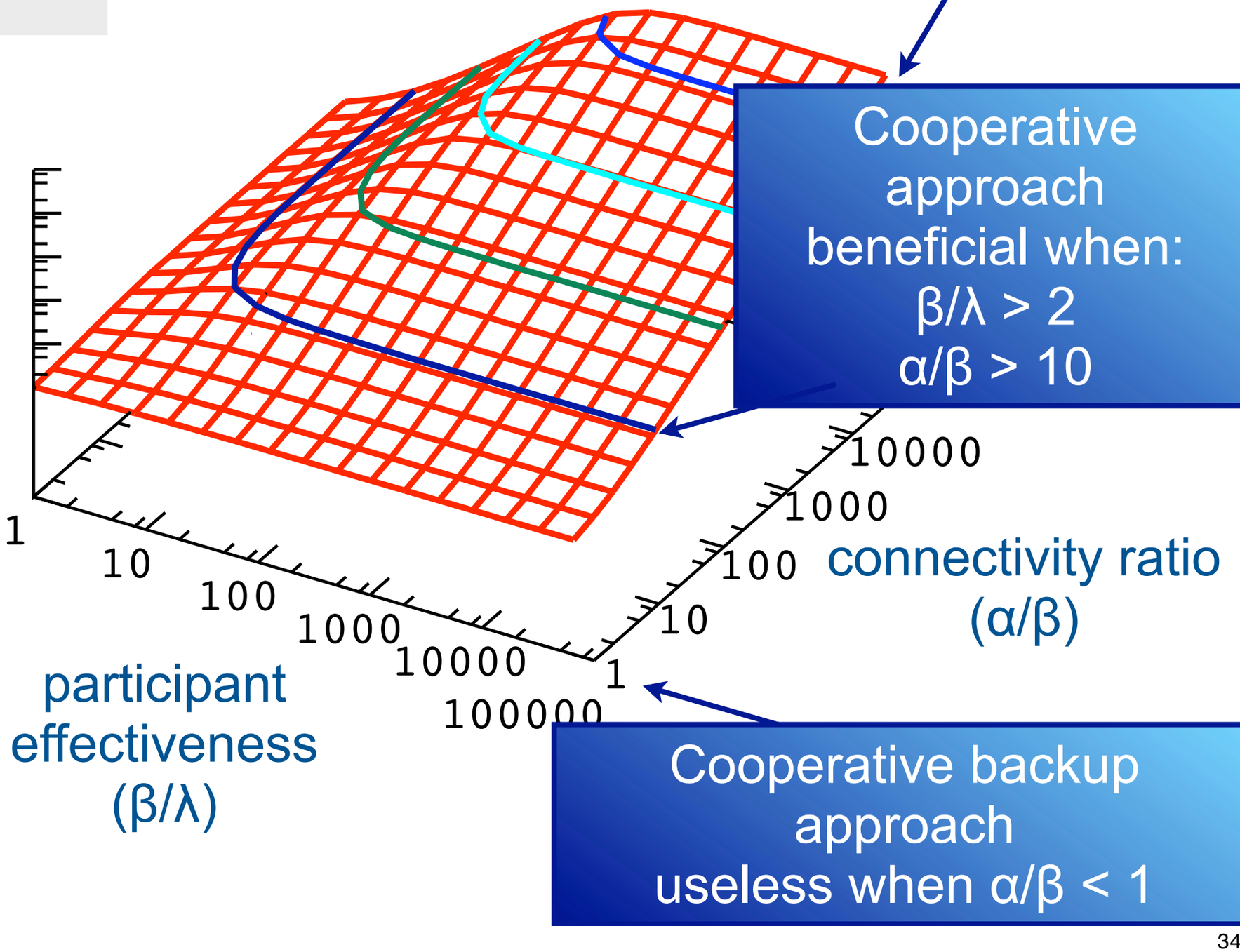


LRF vs. basic parameters

$(n,k) = (3,2)$

α : device encounter rate
β : internet connection rate
λ : device failure rate

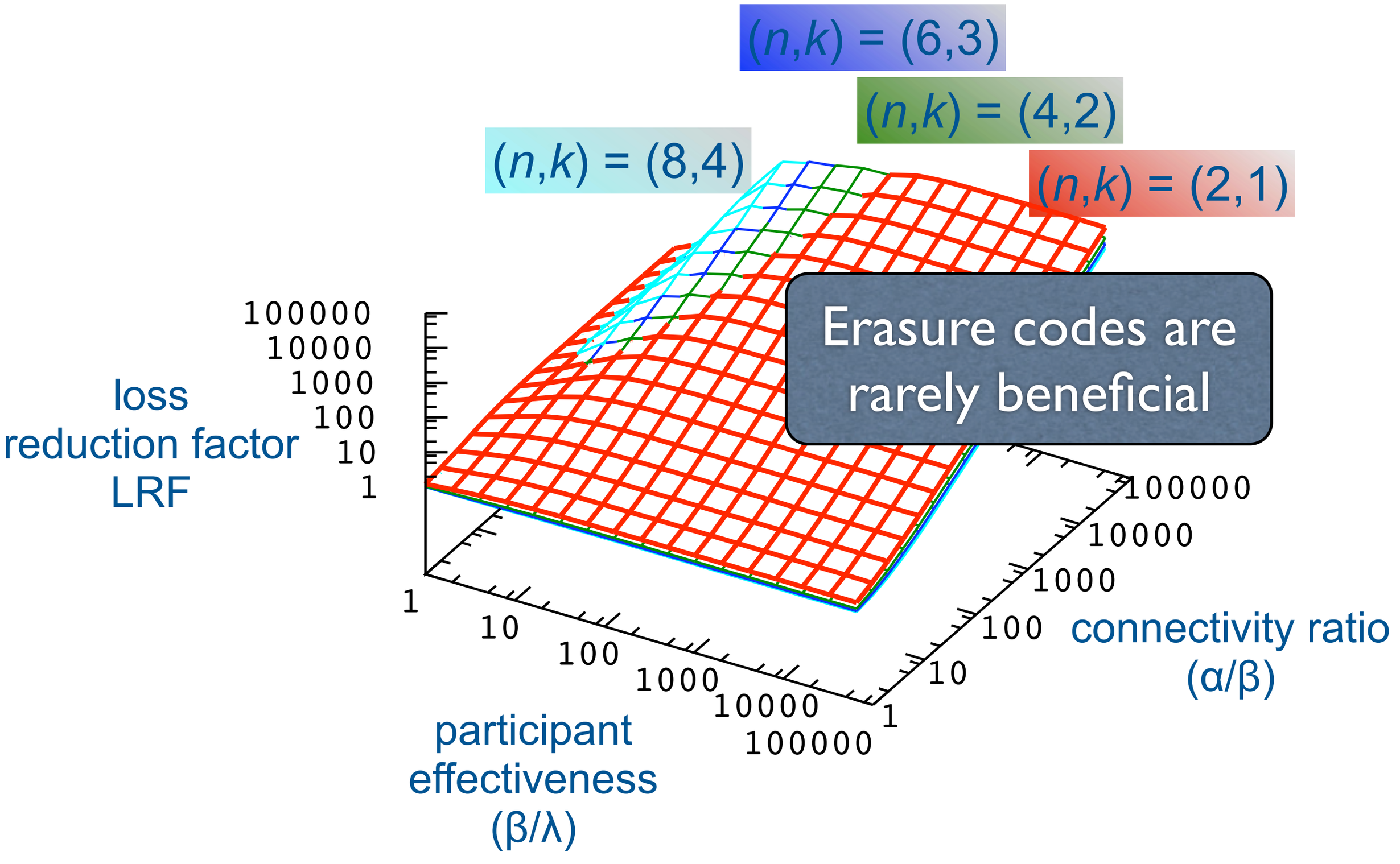
Data loss probability decreased by up to α/β



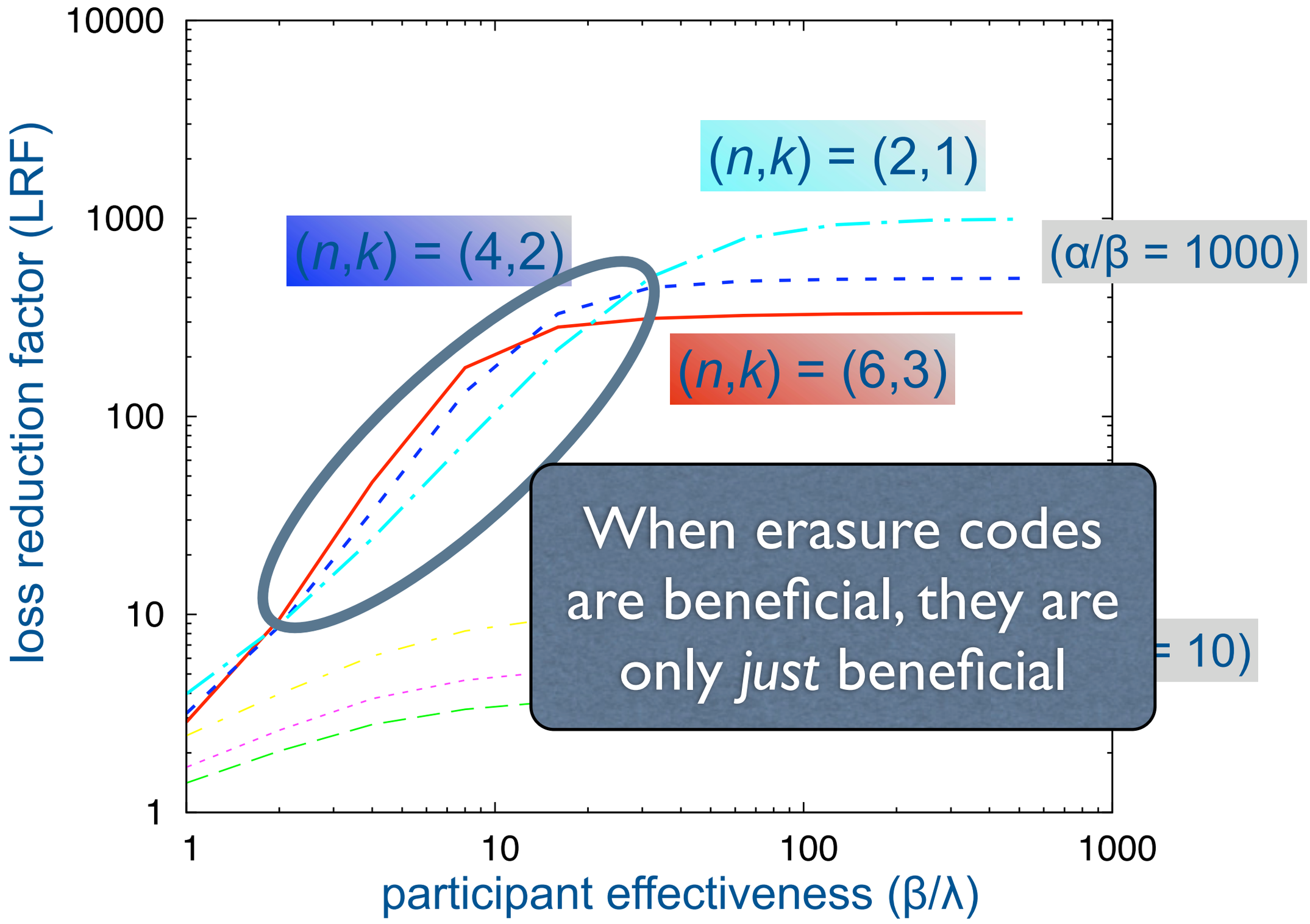
Cooperative approach beneficial when:
β/λ > 2
α/β > 10

Cooperative backup approach useless when α/β < 1

LRF vs. coding parameters



LRF vs. coding parameters



Backup Availability Summary

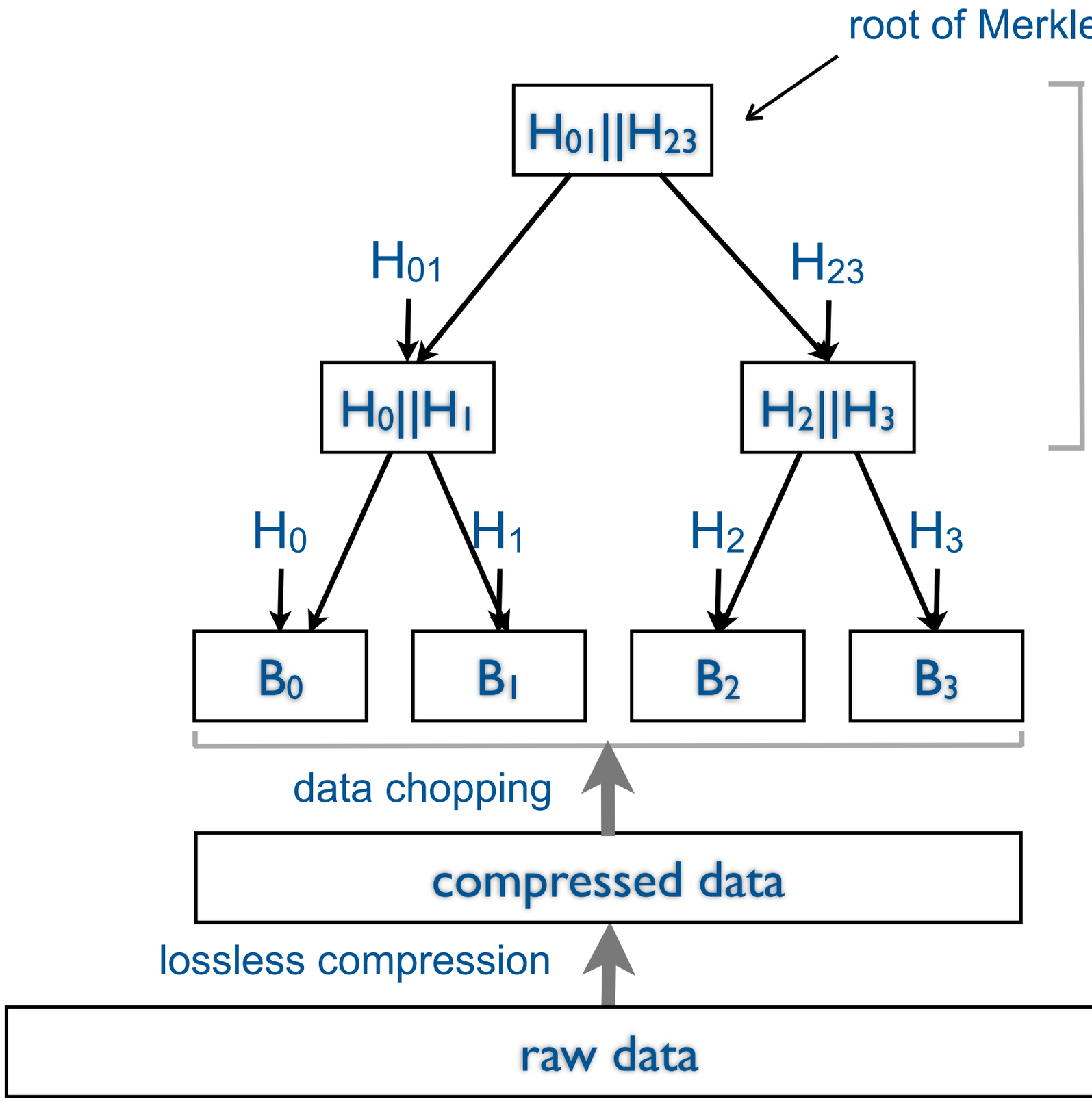
- Intermediate backups through cooperation
- LRF up to connectivity ratio α/β
- Order of magnitude gain when α/β>10 and β/λ>2
- Erasure codes have small advantage over simple replication in only a very narrow domain

Challenge 2 - Security & Performance of Intermediate Backups

- **Participants may be malicious**
  - enforce data & meta-data confidentiality
  - check data & meta-data integrity and authenticity
- **Unpredictable encounters and encounter durations**
  - chop data into small blocks
  - distribute blocks to encountered contributors
- **Scarce resources (storage, energy)**
  - maximize storage efficiency
  - energy-efficient algorithms (compression, crytpo)

→ *experimental evaluation of data chopping & compression techniques yielding best CPU/storage trade-off [EDCC-6]*

Storage Layer Design Choices

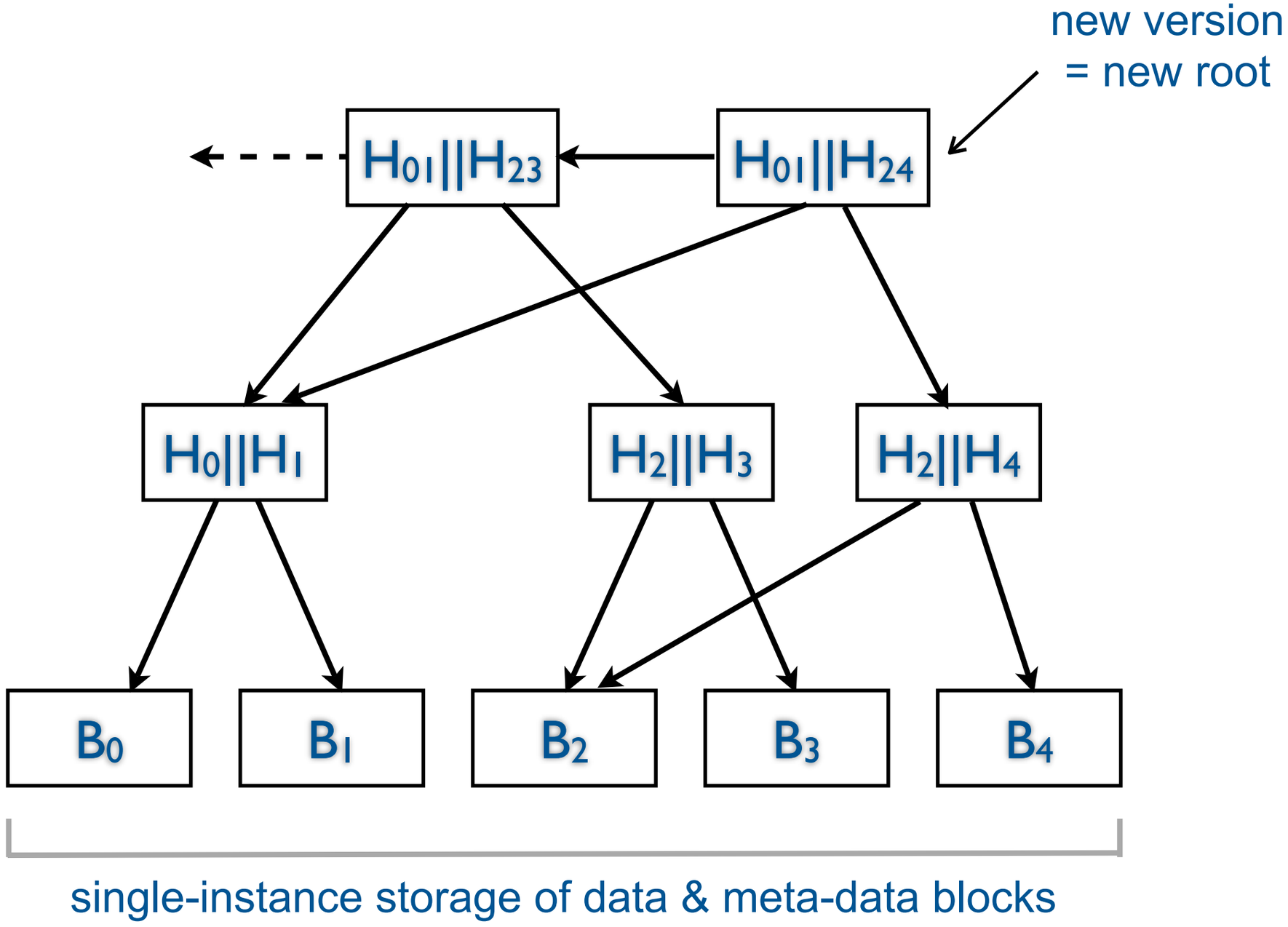


fixed-size meta-data blocks

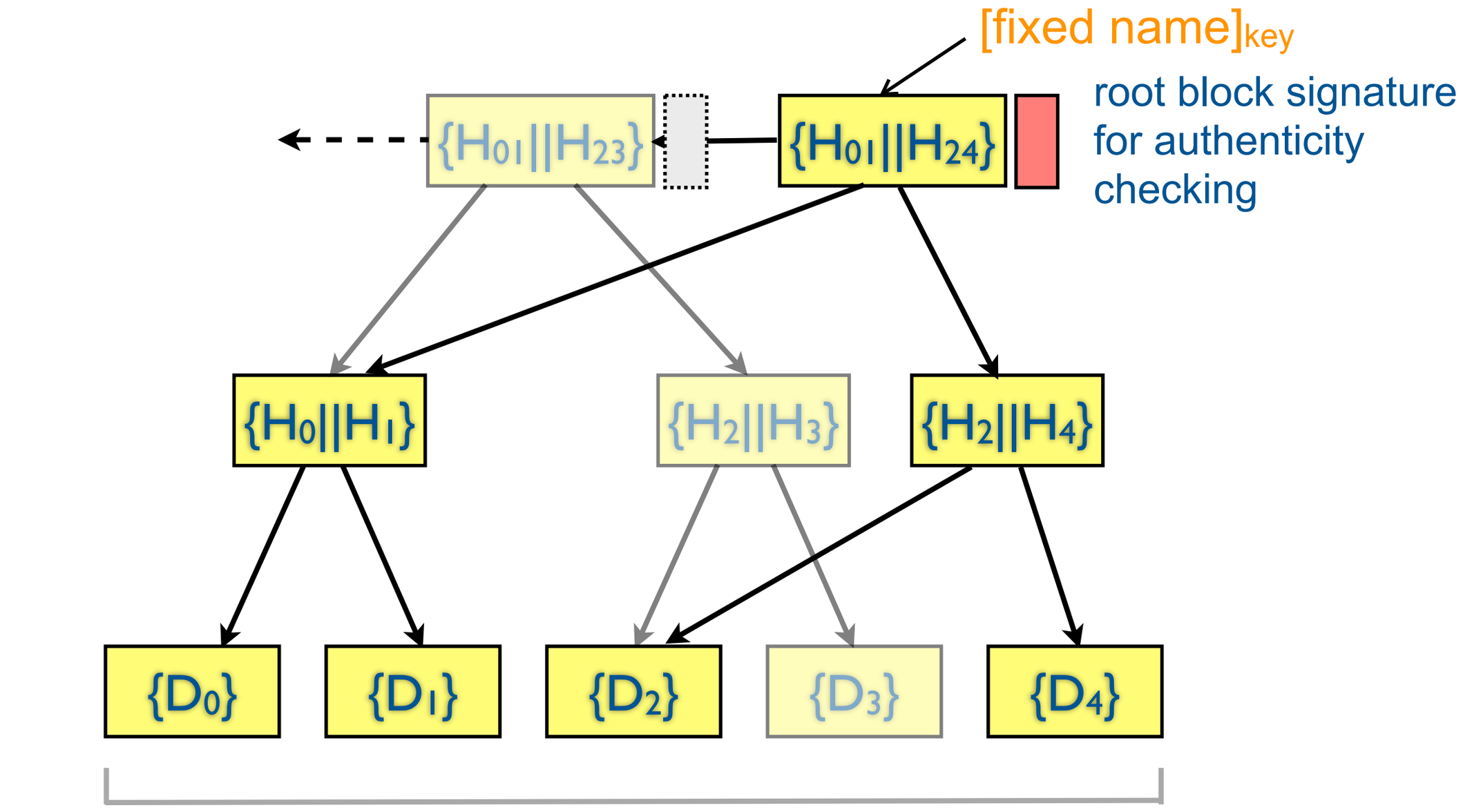
$H_i = hash(B_i)$
(content-based addressing)
→ single instance storage

fixed-size data blocks

Storage Layer Design Choices



Storage Layer Design Choices



convergent block encryption for confidentiality and integrity checking

$$\{X\} = \text{SymmetricEncryption}(X)_{key=hash(X)}$$

Challenge 3 - Cooperation Security

- **Issues**
  - participants may be malicious
  - participants may be selfish
- **Threats**
  - data retention/deletion
    - contributor does not give data back
  - flooding
    - data owner exhausts storage resources
  - free-riding
    - no return for service rendered (Tragedy of the Commons)

Security Management Options

- **Single-authority**
  - distribution of certificates or other security material to all users
  - imposition of particular policy or mechanism
  - external sanctions for misbehavior
- **Self-organization**
  - no reliance on single point of trust
  - user chooses his own cooperation policy
    - ➥ need *policy-neutral mechanisms*
  - communal sanctions for misbehavior
    - ➥ need *accountability*

Accountability

- **Device designation**
  - (also for per-owner block name spaces)
  - self-organized: users create their own designator in the form of a public key
    - statistically unique & cryptographically verifiable (SUCV)
      - authentication: device holds secret key
  - public keys too large in practice
    - identifier: signed cryptographic hash of public key
- **Binding of actions to devices**
  - guarantee communication integrity and authenticity
  - keyed Hash-based Message Authentication Code (HMAC)

Self-organized Security

- **Sybil attacks**
  - pitfall of self-generated identifiers
  - can escape accountability for past actions
  - thwart partially by appropriate cooperation policies:
    - short-term advantage of using *new* identifier (some resources granted to bootstrap cooperation)
    - long-term advantage of using *same* identifier
- **Cooperation policies**
  - based on social relationships (friends of friends)
  - based on observations of past behavior (reputation systems)

Related Work

- FLASHBACK
- UbiStore
- Asynchronous wireless sensor networks
- Delay- and disruption-tolerant networks

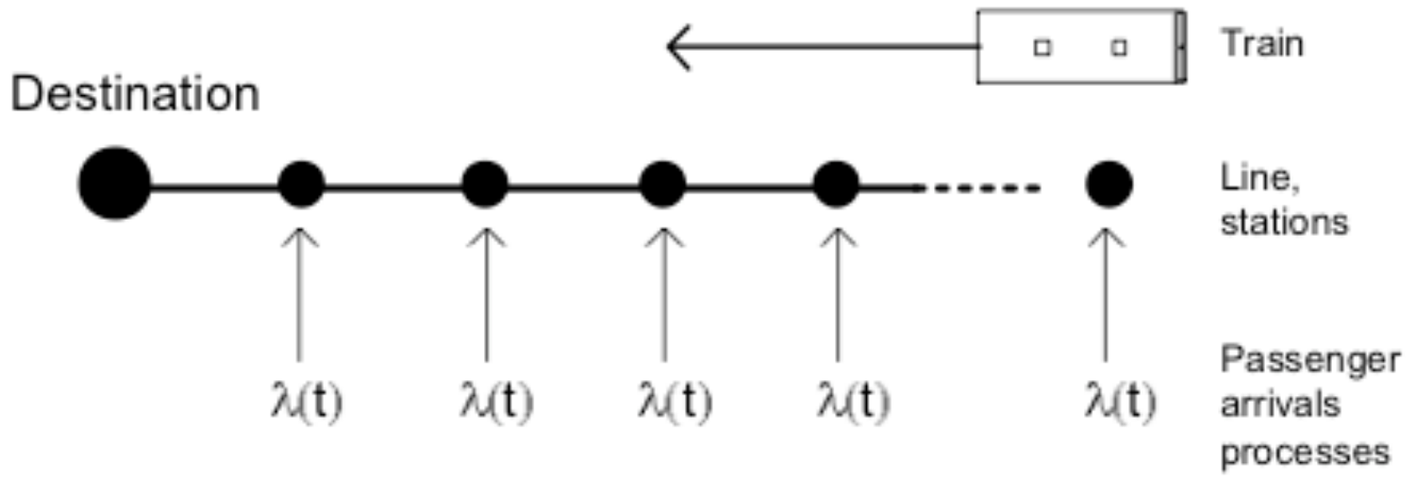
FLASHBACK [Loo *et al.* 2003]

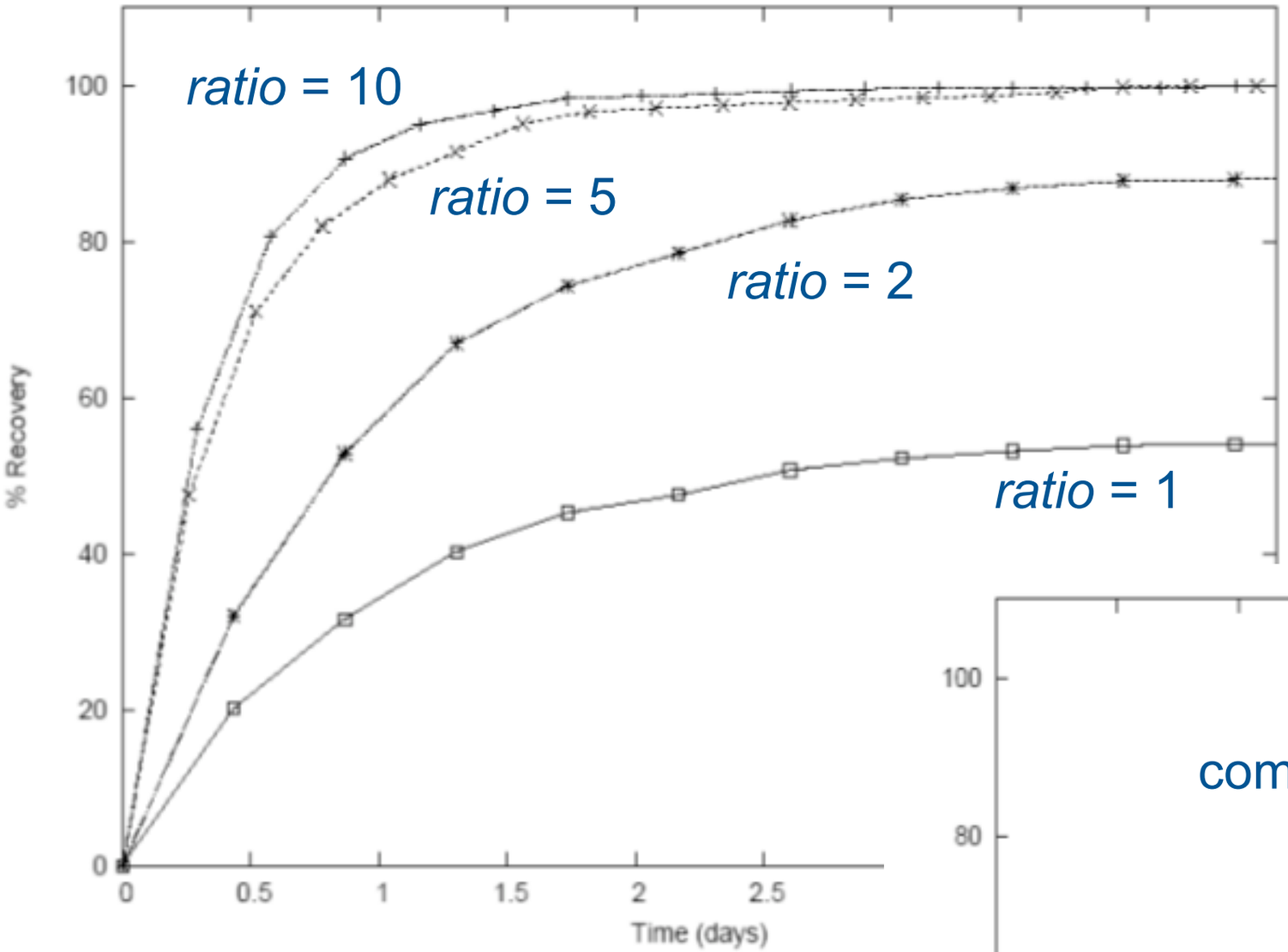
- **Like MoSAIC**
  - digital devices (phones, PDAs, laptops…)
  - free shortrange P2P communication
  - automatic collaborative backup
- **Unlike MoSAIC**
  - trusted PAN of devices belonging to single user
  - repetitive, mostly static connectivity
  - choice of contributors according to a utility function based on:
    - available power
    - available storage
    - pairwise locality

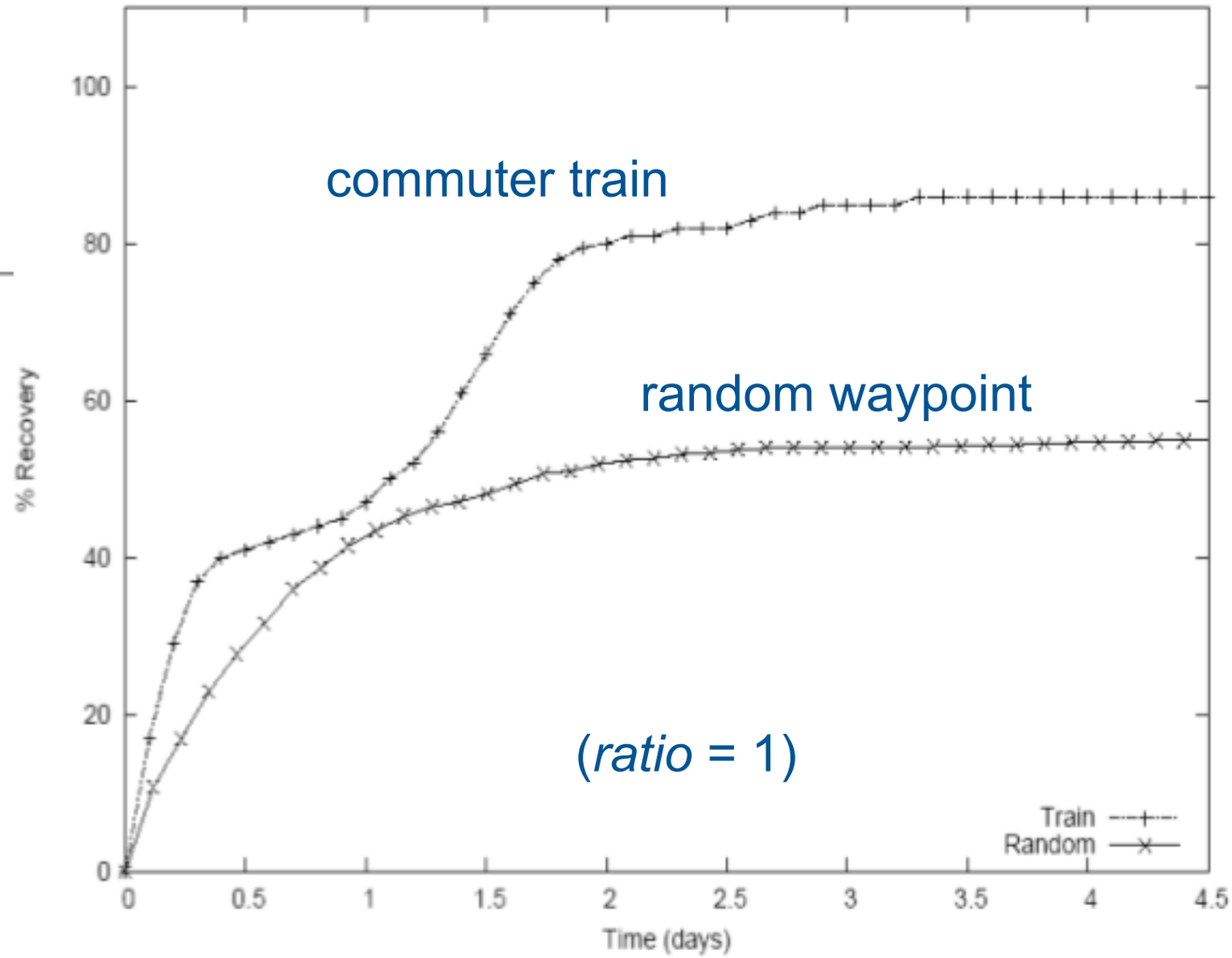
UbiStore [Tan *et al.* 2007]

- **Like MoSAIC**
  - digital devices (phones, PDAs, laptops…)
  - free shortrange P2P communication
  - automatic collaborative backup
  - user credentials survive data/device failure or loss
- **Unlike MoSAIC**
  - repetitive mobility patterns of human device owners
  - same peers re-encountered with high probability
  - data recovered directly from peer devices
  - data blocks prioritized according to dynamically estimated “recoup” time: $t_{n+1}(i) = t_n(i) + [t_n(i) - t_{n-1}(i)]$
  - priority decides order of block backup and deletion

- Evaluation by simulation
  - 100 nodes:
    - data owners: 200 Mb data
    - contributors: ratio x 200 Mb backup storage with ratio ∈ [1,10]
  - 2s transmission time (fixed priority) per 0.4 Mb block
  - 10 days of back-up only behavior
  - time to recover N% of data by 10 nodes
- Two mobility models
  - random waypoint (1km^2; v∈[1,5] m/s; δ=5 min)
  - correlated “commuter train” model



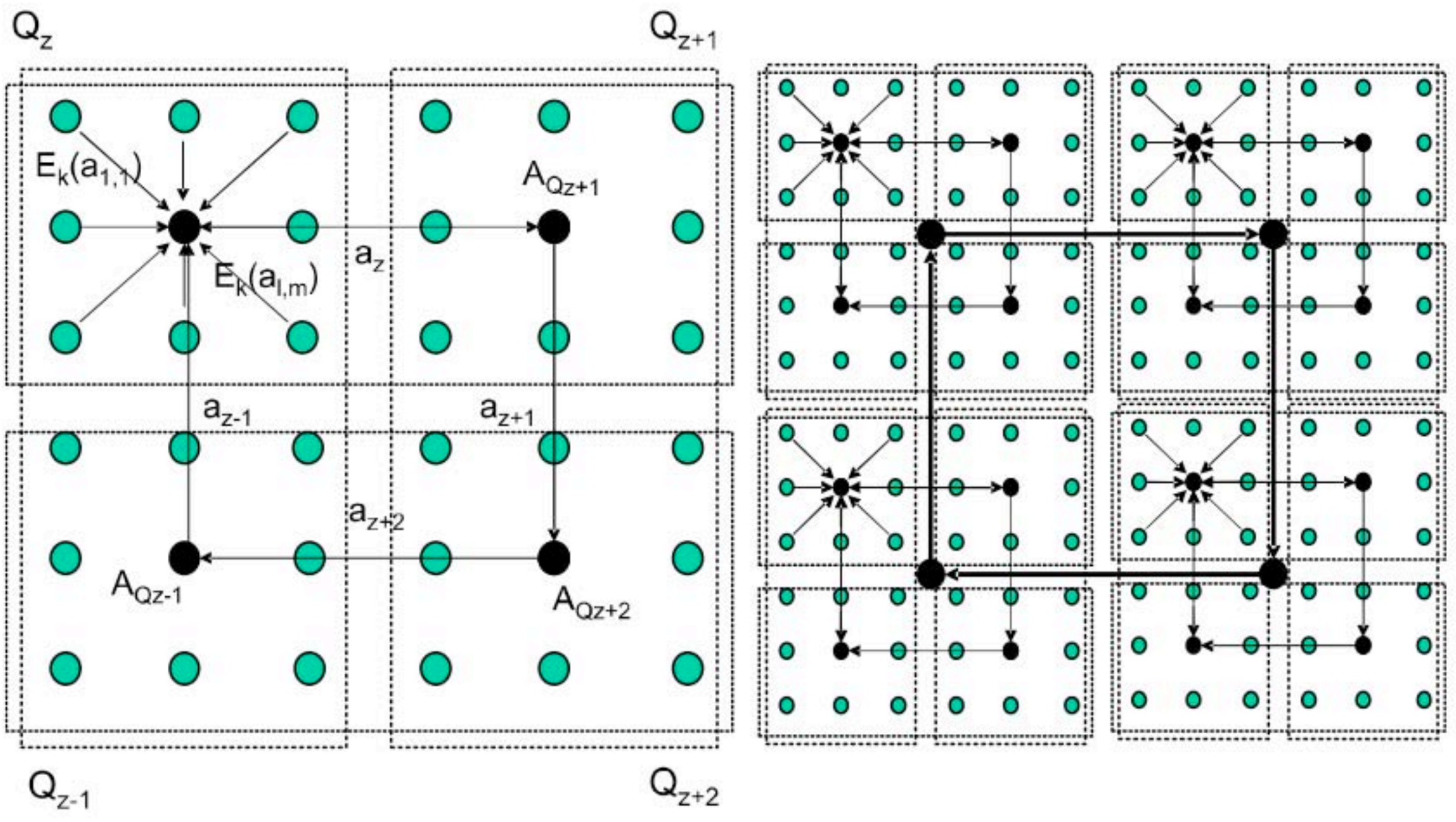

ratio = 10
ratio = 5
ratio = 2
ratio = 1
% Recovery
Time (days)
0
20
40
60
80
100
0
0.5
1
1.5
2
2.5


commuter train
random waypoint
(ratio = 1)
Train
Random
% Recovery
Time (days)
0
20
40
60
80
100
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5

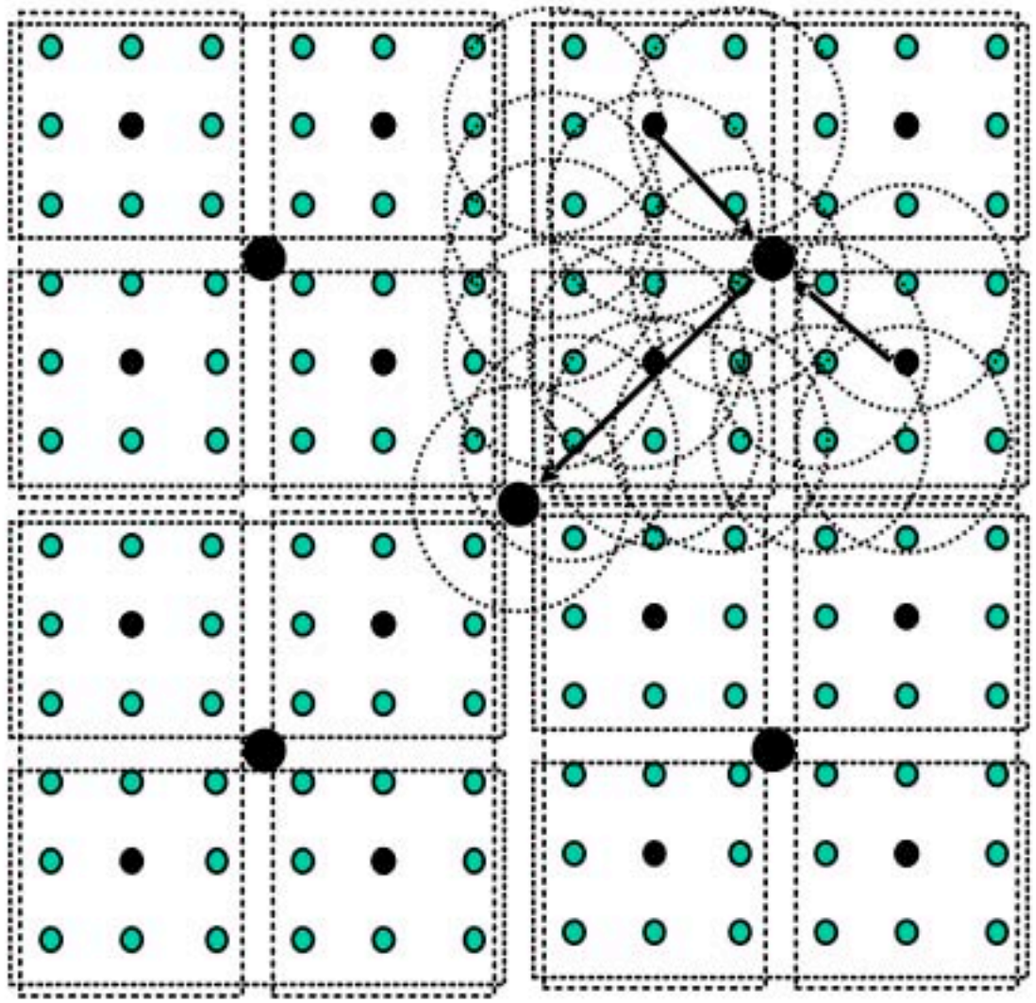
Asynchronous Wireless Sensor Networks

- **Characteristics**
  - Continuous monitoring and *storage* of data
  - Very limited storage space
  - Energy consumption critical
- **Collaborative storage** [Tilak *et al.* 2005]
  - Collaboration to *decrease* redundancy and storage
- **Persistent (encrypted) storage** [Girao *et al.* 2007]
  - Replication to survive node failures and disasters
    - nodes or regions of nodes may exhaust their energy
    - disasters: nodes within circle of radius *r* can die together

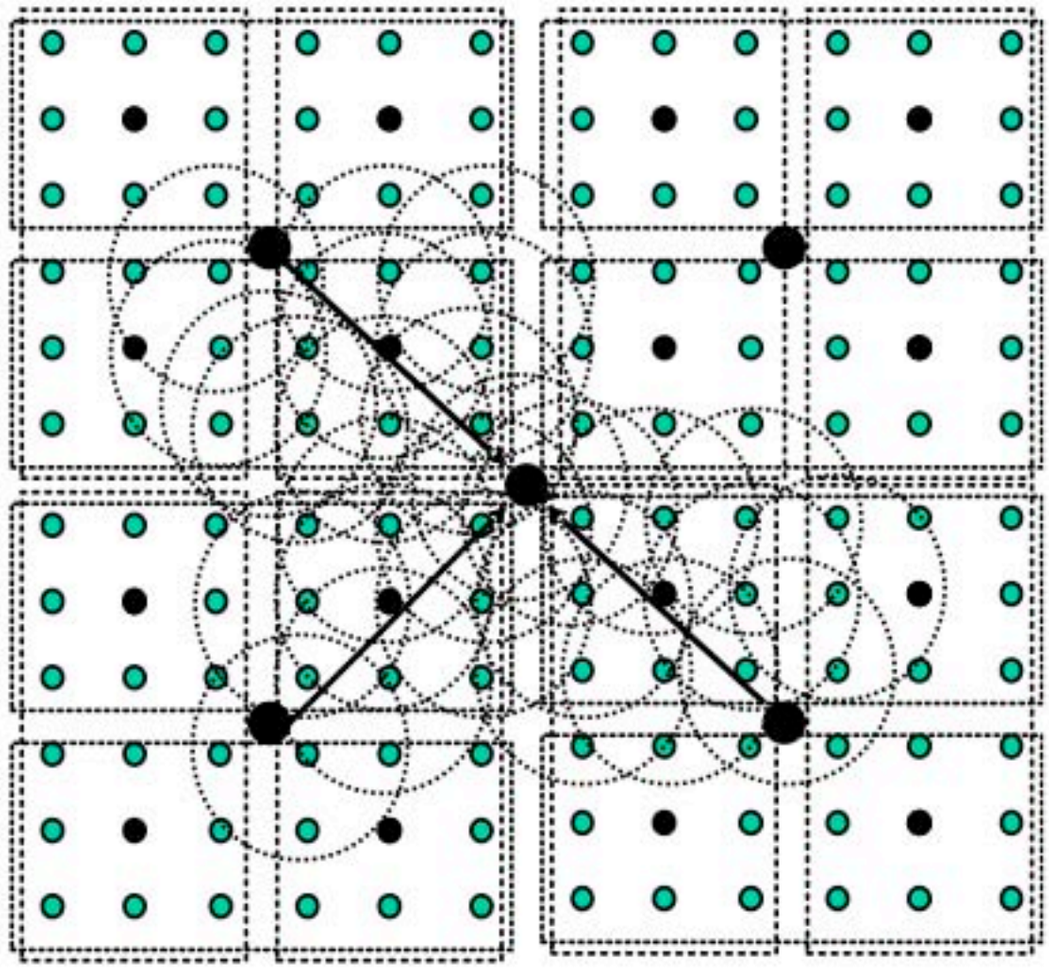
- Nodes organized into a hierarchy of clusters
  - cluster centers at distance > 2r
  - data aggregated at cluster head
  - back-up to neighboring cluster head



- Reader node (mobile)
  - at approximate center of monitored area
  - knows when a disaster has occurred
  - triggers appropriate read protocol



Continuous query



Disaster query

Delay- and Disruption-Tolerant Networking

- **Characteristics**
  - reliable non-interactive messaging (≈ e-mail)
  - {mobile, exotic media, ad-hoc, sensor} networks

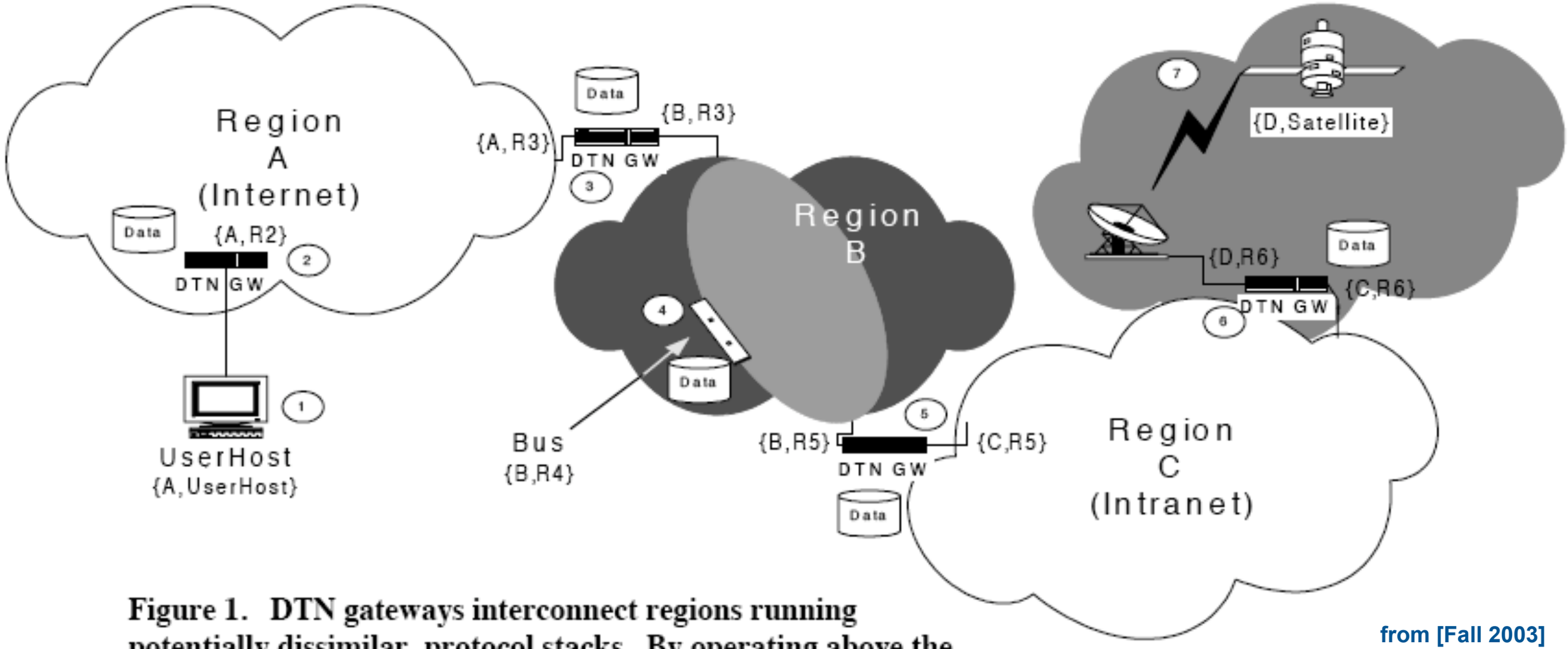
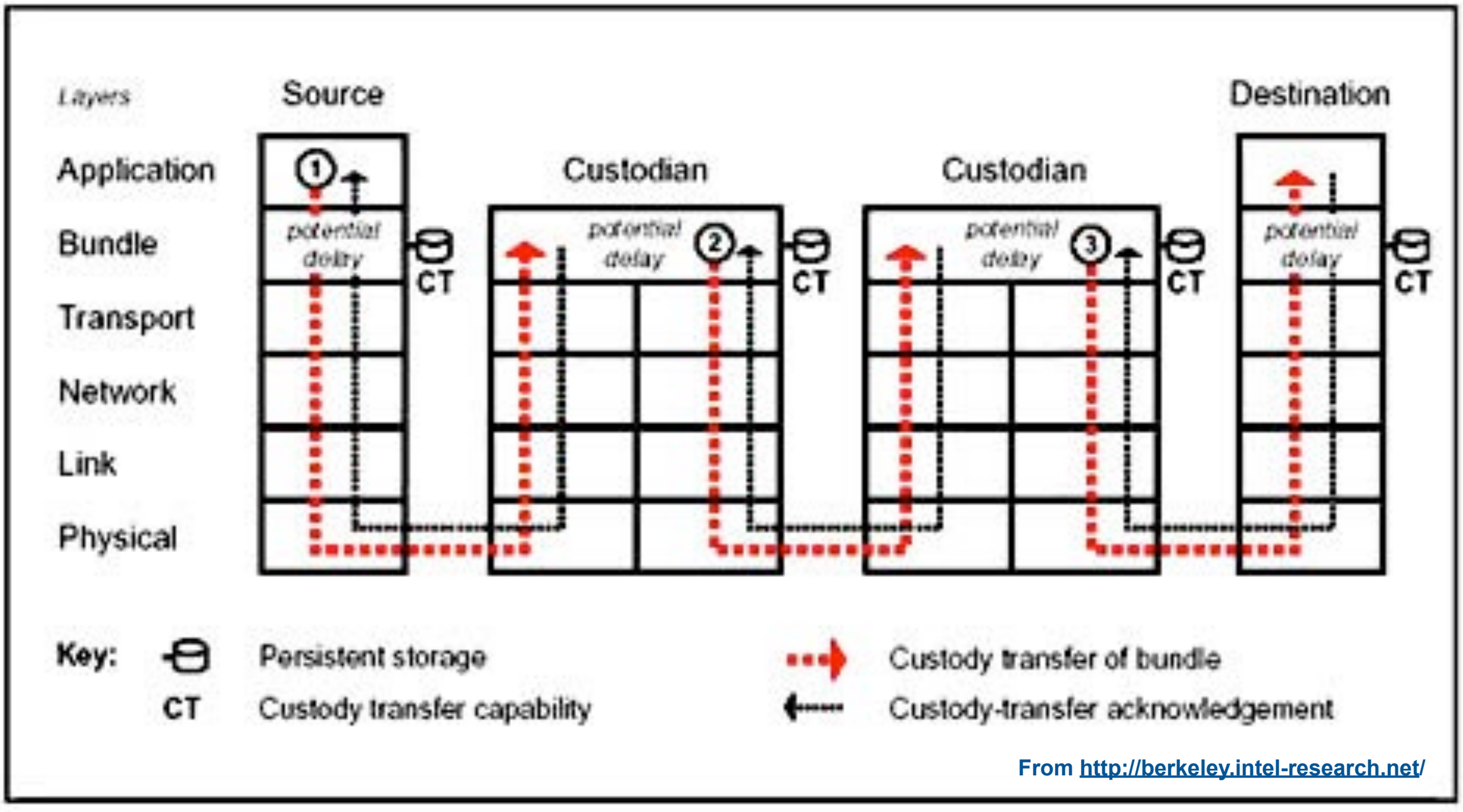


Figure 1. DTN gateways interconnect regions running potentially dissimilar protocol stacks. By operating above the transport protocols in use on the incident networks, they provide virtual message switching, in-network retransmission, and name mapping, allowing globally-interoperable names to be mapped

from [Fall 2003]

- Bundle forwarders or custodians
  - accept delegated responsibility for delivering bundles of messages
  - mobile custodians: “data mules”



From http://berkeley.intel-research.net/

Future Work

- Cooperation policies
- Effect of data-chopping on dependability
- Rate-less erasure codes
- Experimental assessment of α and β (and λ)

References

MoSAIC

- Courtès+, Storage Tradeoffs in a Collaborative Backup Service for Mobile Devices, EDCC'06
- Courtès+, Security Rationale for a Cooperative Backup Service for Mobile Devices, LASC'07
- Courtès+, Dependability Evaluation of Cooperative Backup Strategies for Mobile Devices, PRDC'07
- Courtès, Cooperative Data Backup for Mobile Devices, PhD, University of Toulouse, 2007 http://www.laas.fr/~lcourtes/phd/phd-thesis.fr+en.pdf

Related work

- Fall, A Delay-Tolerant Network Architecture for Challenged Internets., SIGCOMM'03
- Girao+, "TinyPEDS: Tiny persistent encrypted data storage in asynchronous wireless sensor networks", Ad Hoc Networks, 5 (7), 2007
- Loo+, Peer-to-Peer Backup for Personal Area Networks. Intel, Report, 2003
- Tan+, Ubistore: Ubiquitous and Opportunistic Backup Architecture, PerComW'07
- Tilak+, Collaborative Storage Management in Sensor Networks, Int. Journal of Ad Hoc Ubiquitous Computing, 1(1-2), 2005