Cooperative Backup in Sparsely-Connected Mobile Systems

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

LAAS-CNRS, Toulouse, France

Internet of Things Workshop, LAAS-CNRS, 21 October 2008

Cooperative Backup
(MoSAIC project)

Cooperative Backup
(MoSAIC project)

- Mobile devices are subject to damage, loss, theft...

Cooperative Backup
(MoSAIC project)

- **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)
  - long distance wireless bandwidth increasing more slowly than rate of production of data on mobile devices

Cooperative Backup
(MoSAIC project)

- **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)
  - long distance wireless bandwidth increasing more slowly than rate of production of data on mobile devices

⇒ **Backup opportunities are rare, data is at risk**

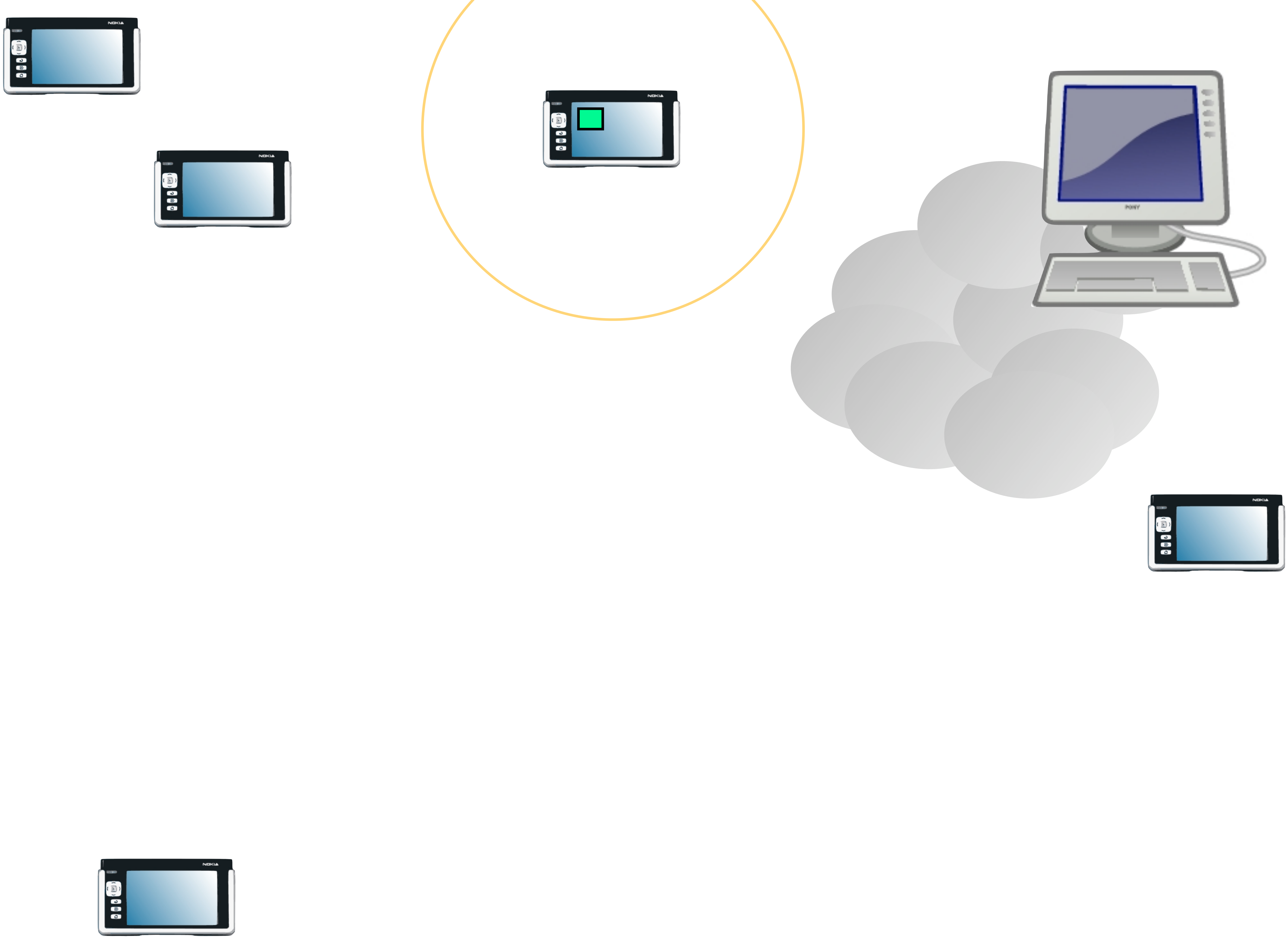
Cooperative Backup

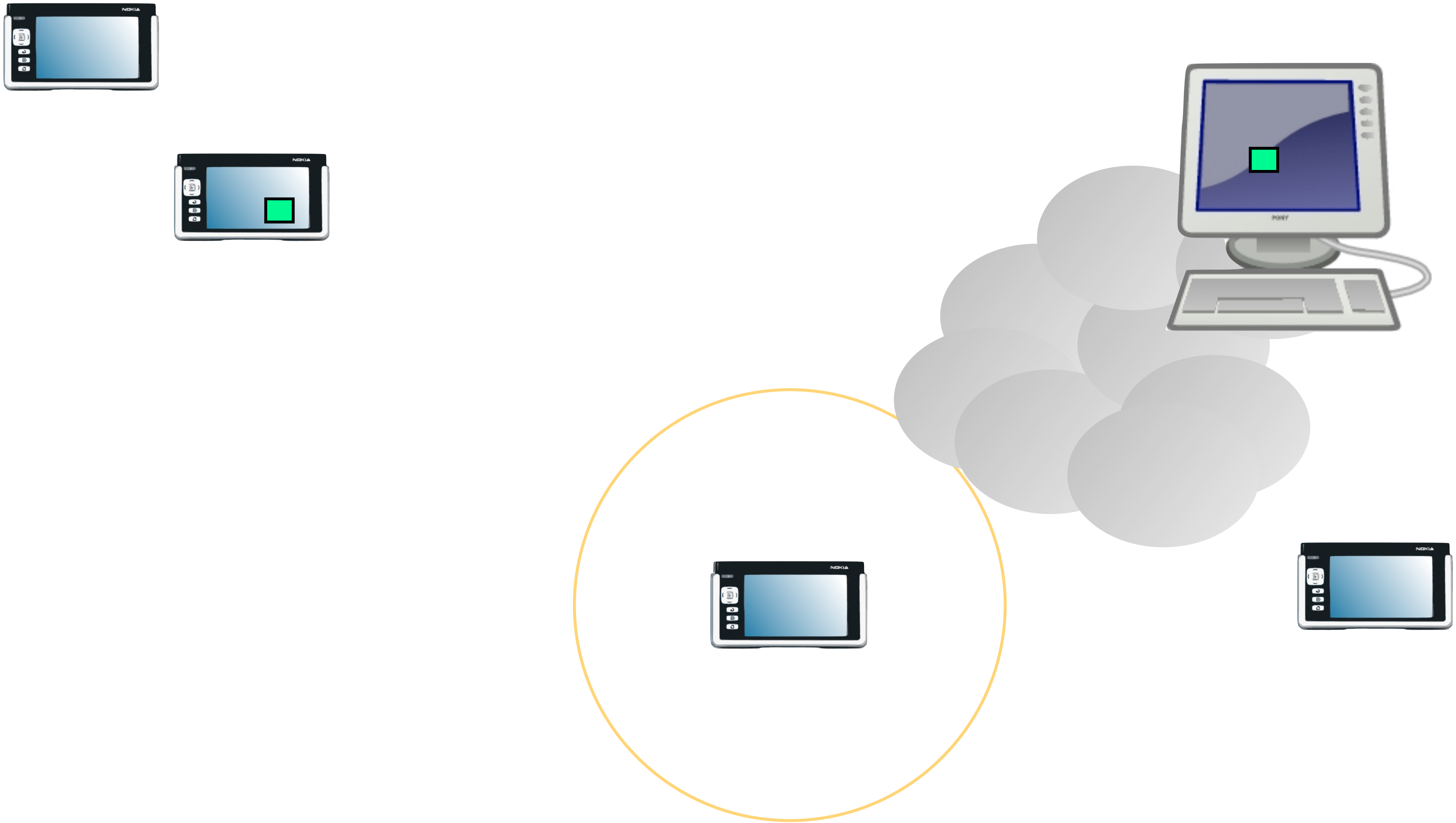
Cooperative Backup

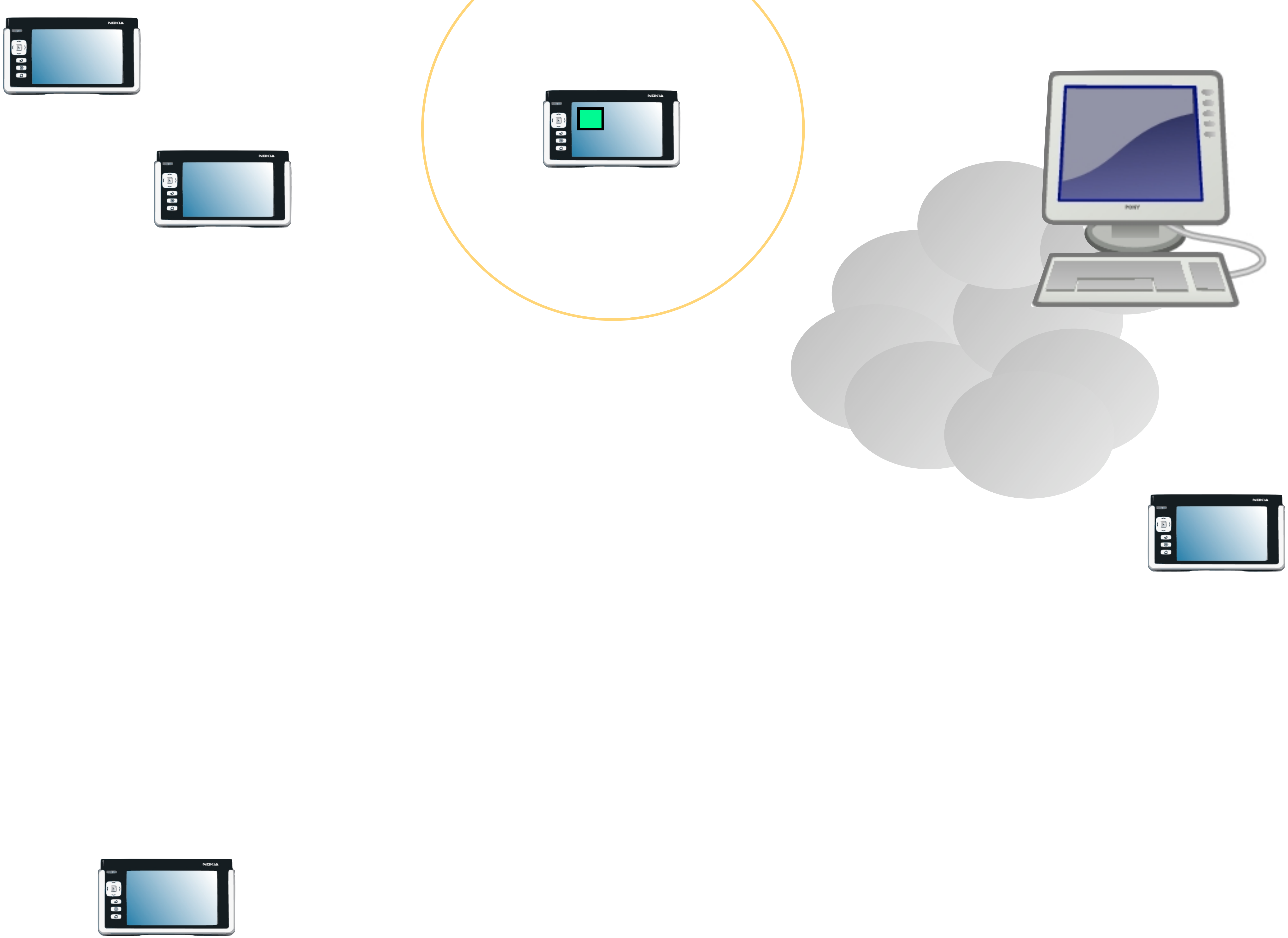
- **Key Ideas**
  - leverage computing device ubiquity
  - opportunistic replication to neighboring devices
  - free shortrange P2P communication (Wi-Fi, Bluetooth)

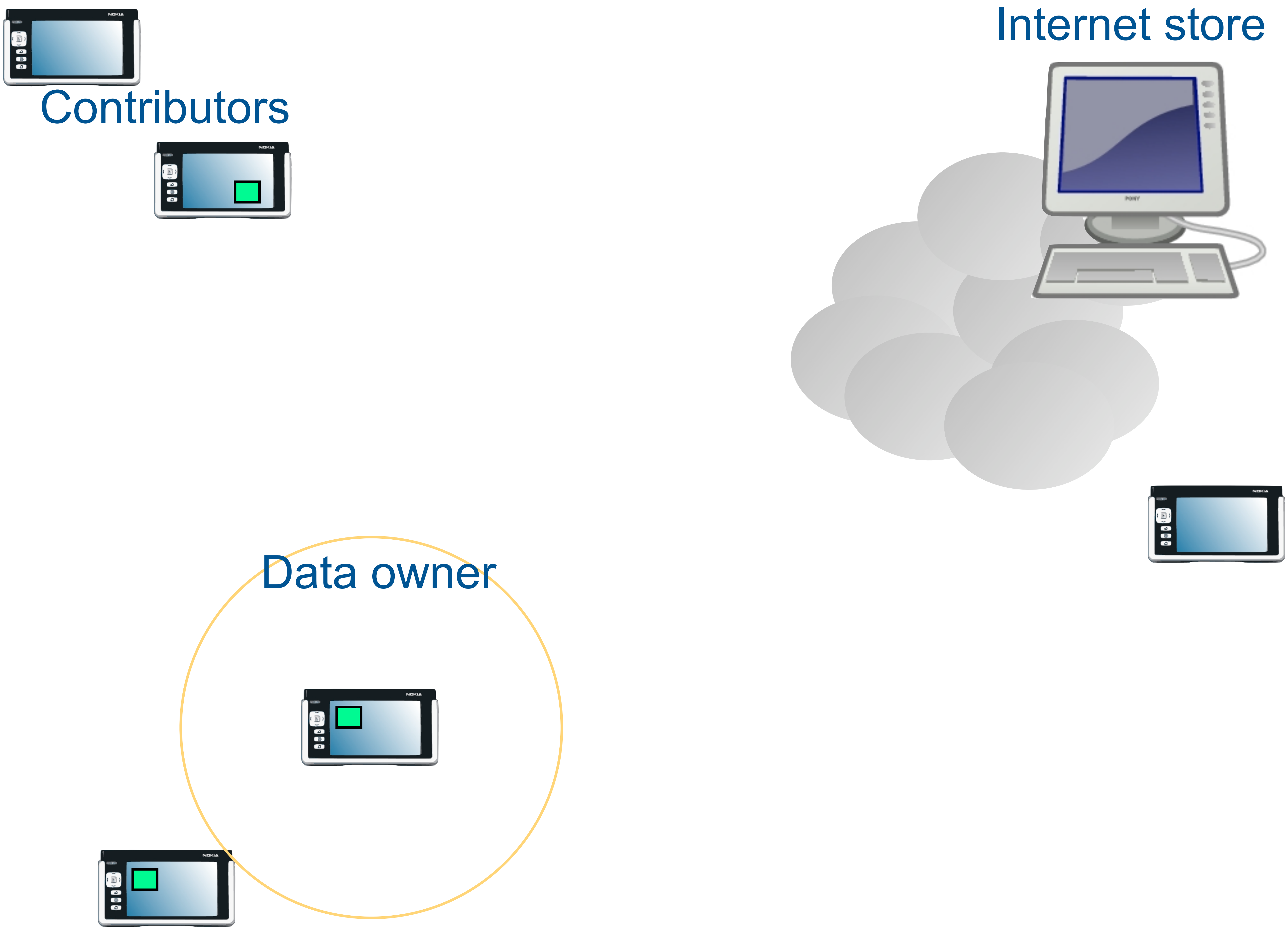
Cooperative Backup

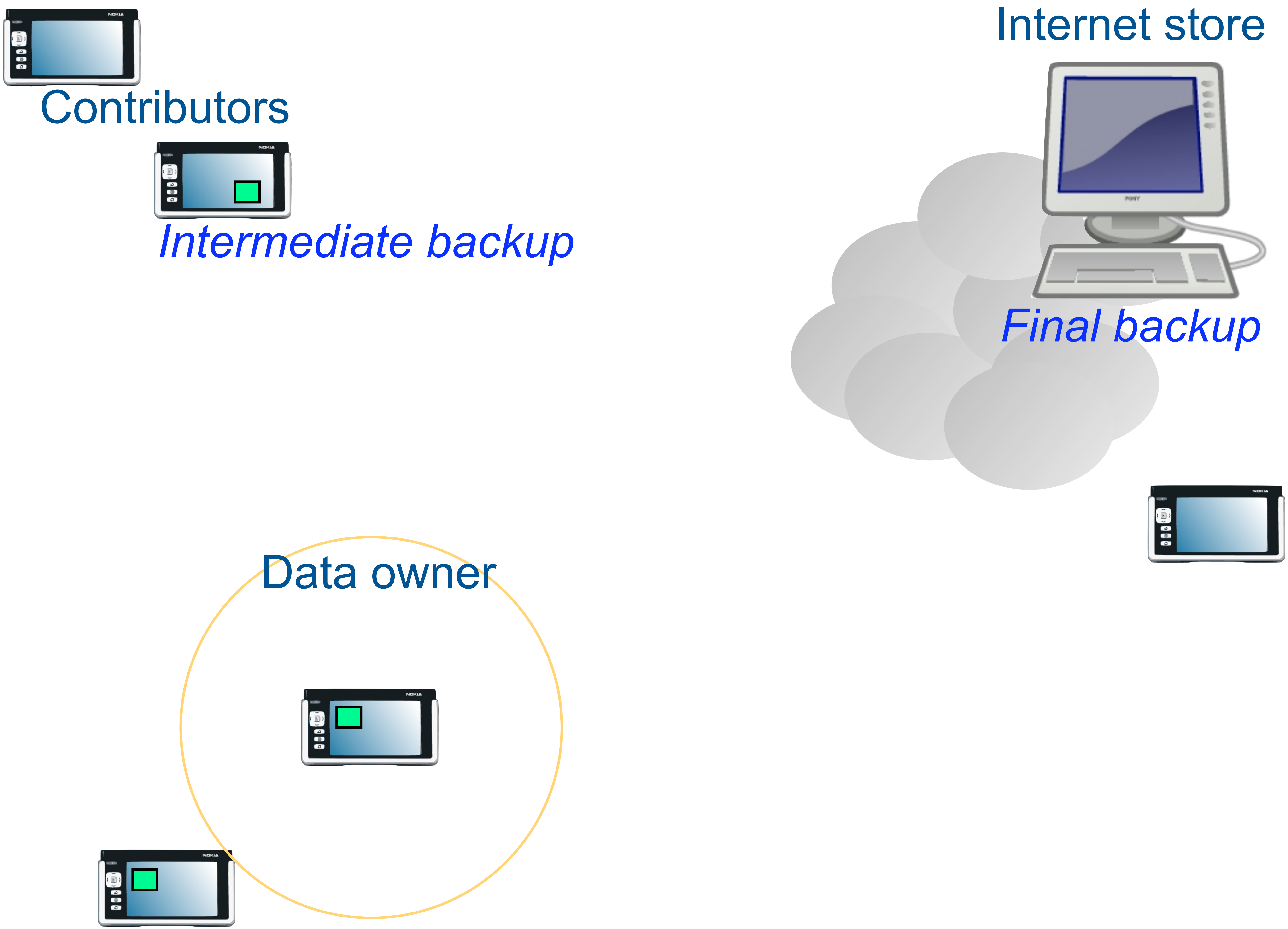
- **Key Ideas**
  - leverage computing device ubiquity
  - opportunistic replication to neighboring devices
  - free shortrange P2P communication (Wi-Fi, Bluetooth)
- **Salient Points**
  - adapted to sparsely-connected mobile systems with intermittent connectivity
    - *intermediate backup* on neighboring devices
    - *final backup* on reliable Internet store
  - continuous backup & replication








Internet store
Contributors
Data owner


Internet store
Contributors
Intermediate backup
Final backup
Data owner

Challenges

- **Backup availability**
  - participants may fail
  - participants may maliciously delete backups
- **Performance and security of intermediate backups**
  - unpredictable encounters and encounter durations
  - scarce resources (storage, energy)
  - participants may maliciously read or modify backups
- **Cooperation effectiveness and security**
  - participants may be selfish
  - participants may maliciously sabotage cooperation

Challenge 1 - Backup Availability

Challenge 1 - Backup Availability

- **Issues**
  - participants may maliciously delete backups
  - participants may fail

Challenge 1 - Backup Availability

- **Issues**
  - participants may maliciously delete backups
  - participants may fail

☞ **need *replicated* intermediate backups**

Challenge 1 - Backup Availability

- **Issues**
  - participants may maliciously delete backups
  - participants may fail
- ☞ **need *replicated* intermediate backups**
- **Optimization goals**
  - data availability…
  - … and storage efficiency

Challenge 1 - Backup Availability

- **Issues**
  - participants may maliciously delete backups
  - participants may fail
  - ☞ **need *replicated* intermediate backups**
- **Optimization goals**
  - data availability…
  - … and storage efficiency
- **Approach**
  - devise replication strategies
  - evaluate the efficiency/availability tradeoff

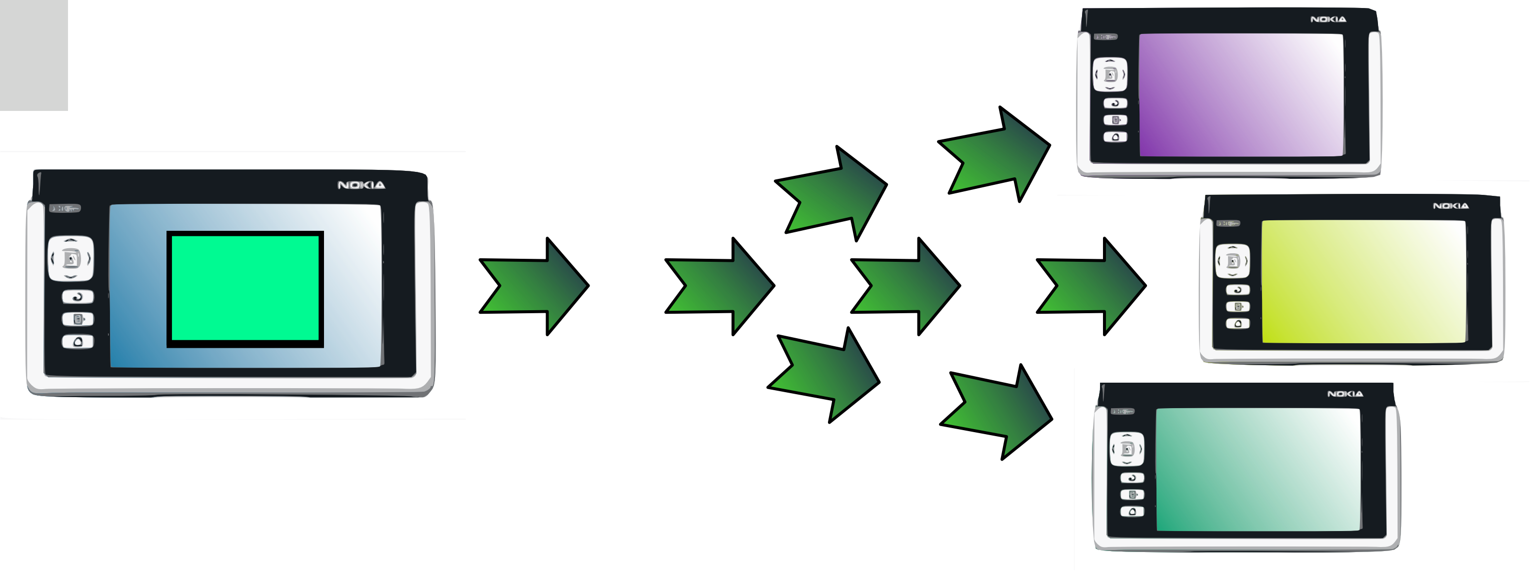
Simple Replication

- **Algorithm**
  - send a total of *n* copies of each data item
  - send 1 copy per contributor
  - recover from any 1 contributor out of *n*

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*

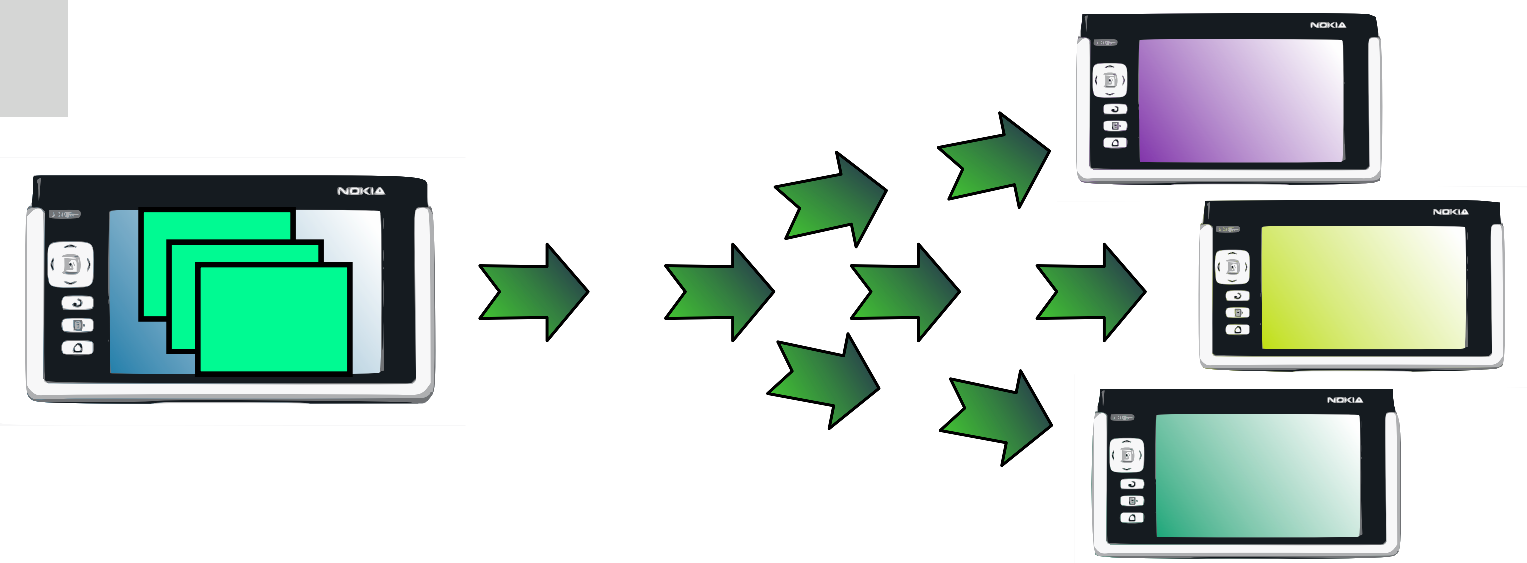
n=3 ; *f*=2



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*

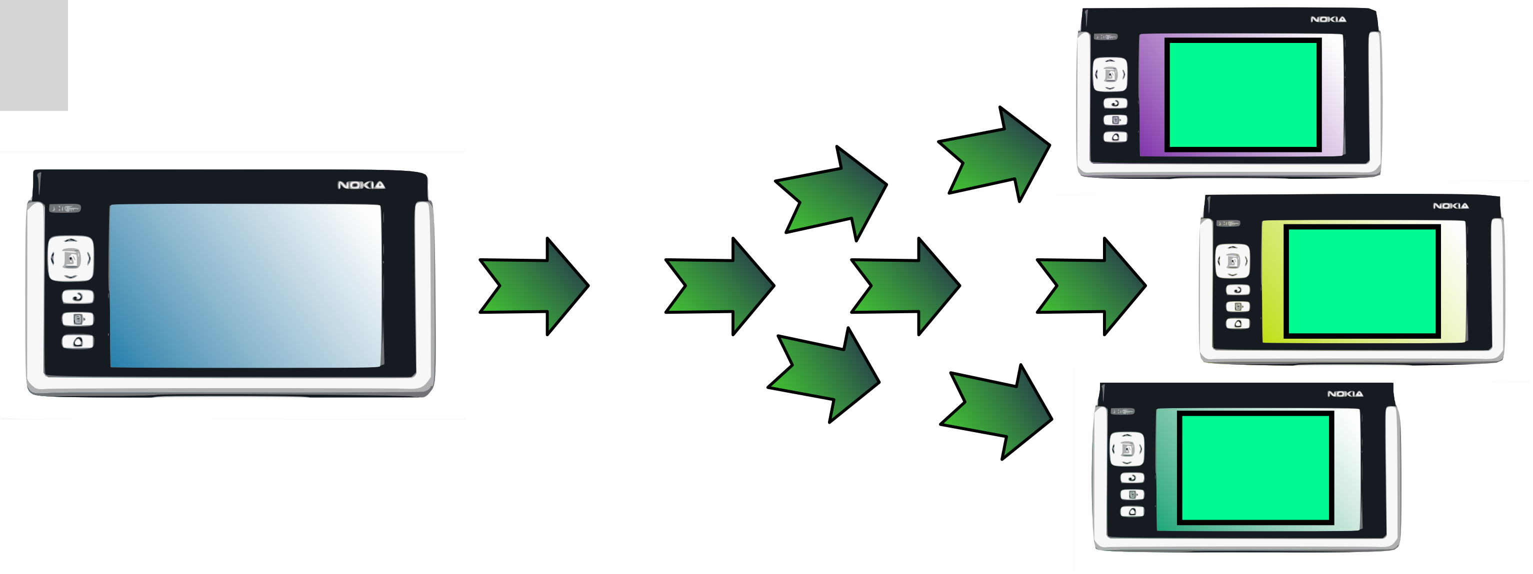
n=3 ; *f*=2



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*

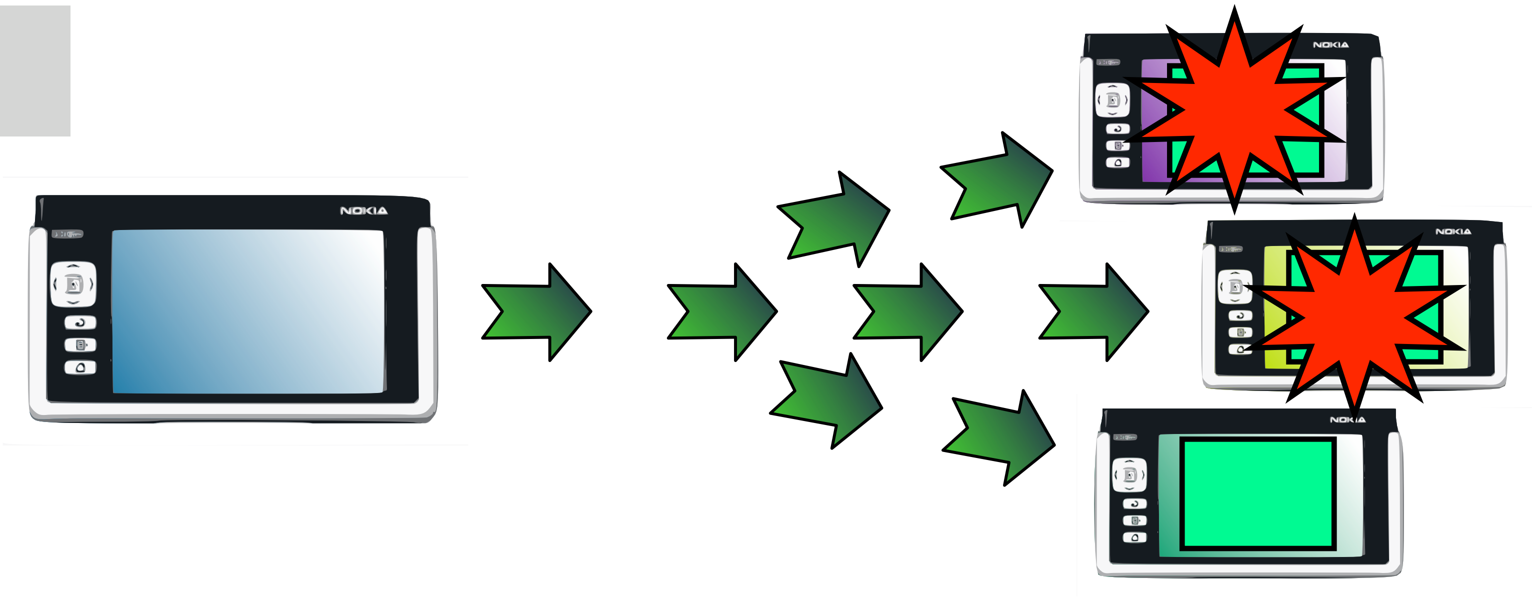
n=3 ; *f*=2



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*

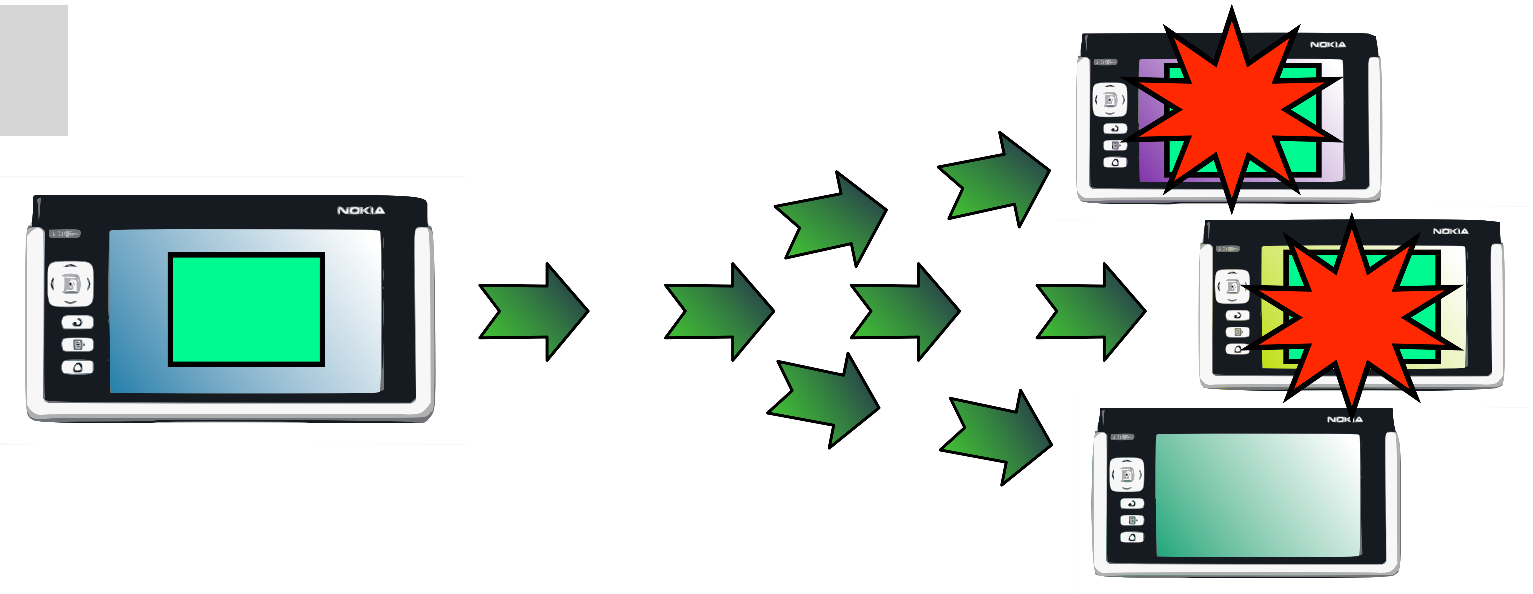
n=3 ; *f*=2



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*

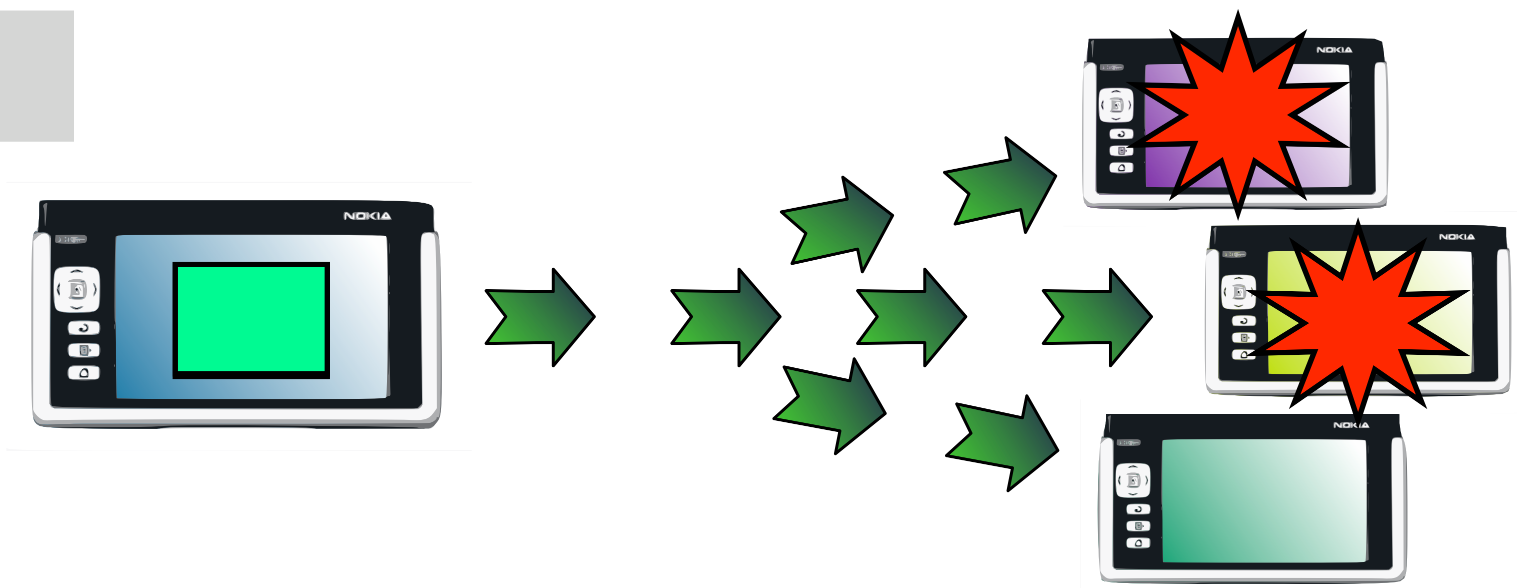
n=3 ; f=2



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*

n=3 ; *f*=2



- **Dependability & storage cost analysis**
  - tolerate *f* contributor faults ⇒ storage cost *f* + 1

Erasure Codes

Erasure Codes

- **Basics**
  - *k*-block input → *n* coded blocks, $n > k$
  - *m* blocks suffice to recover input data $k \leq m < n$
  - tolerate n-m faults
  - storage cost: $S = n/k$

Erasure Codes

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

Erasure Codes

Erasure Codes

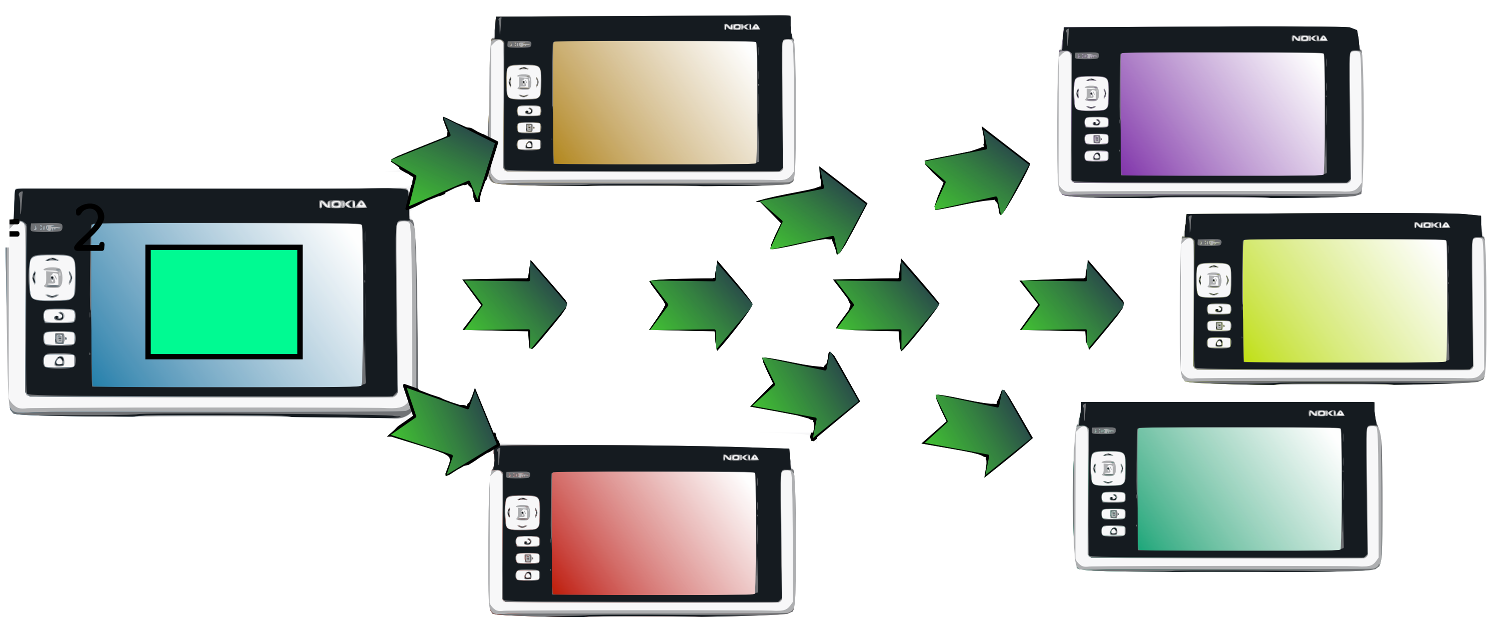
- **Algorithm**
  1. (n,k) erasure coding → n coded blocks
  2. send 1 coded block per contributor
  3. recover from any k contributors out of n

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

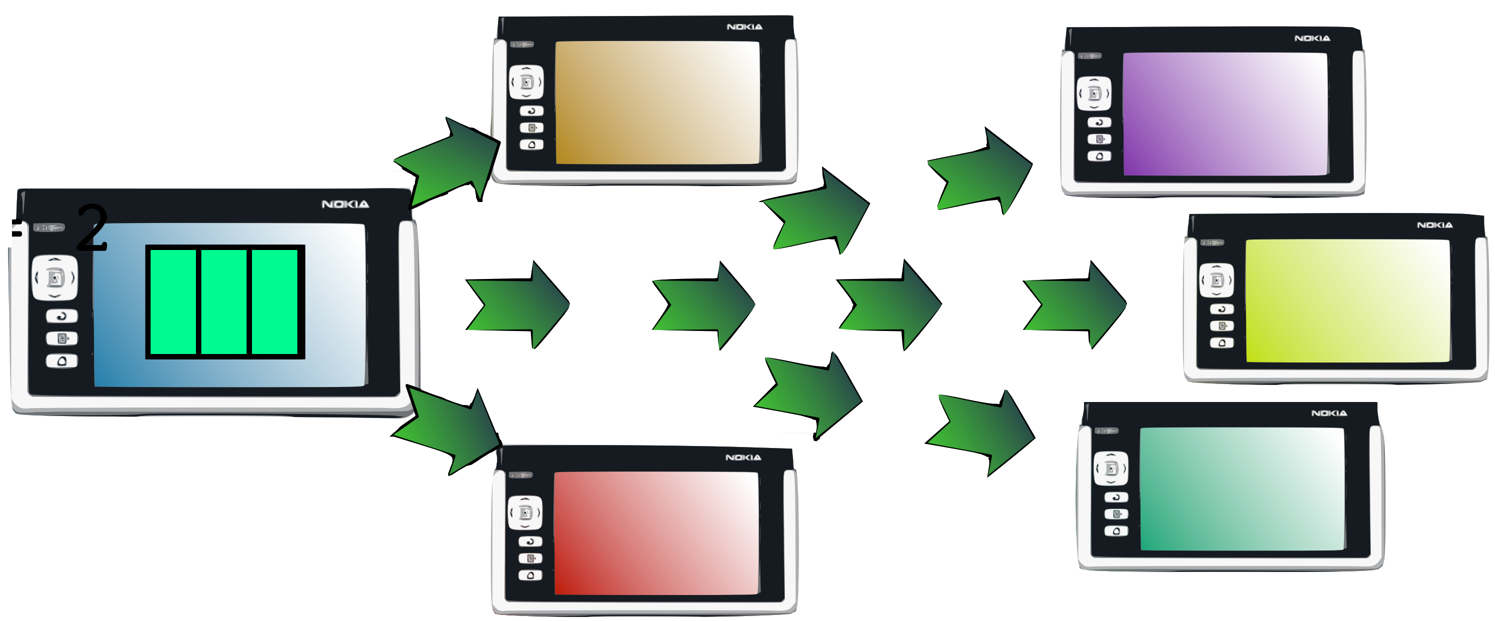
$(n,k) = (5,3)$
$f = n - k = 2$



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

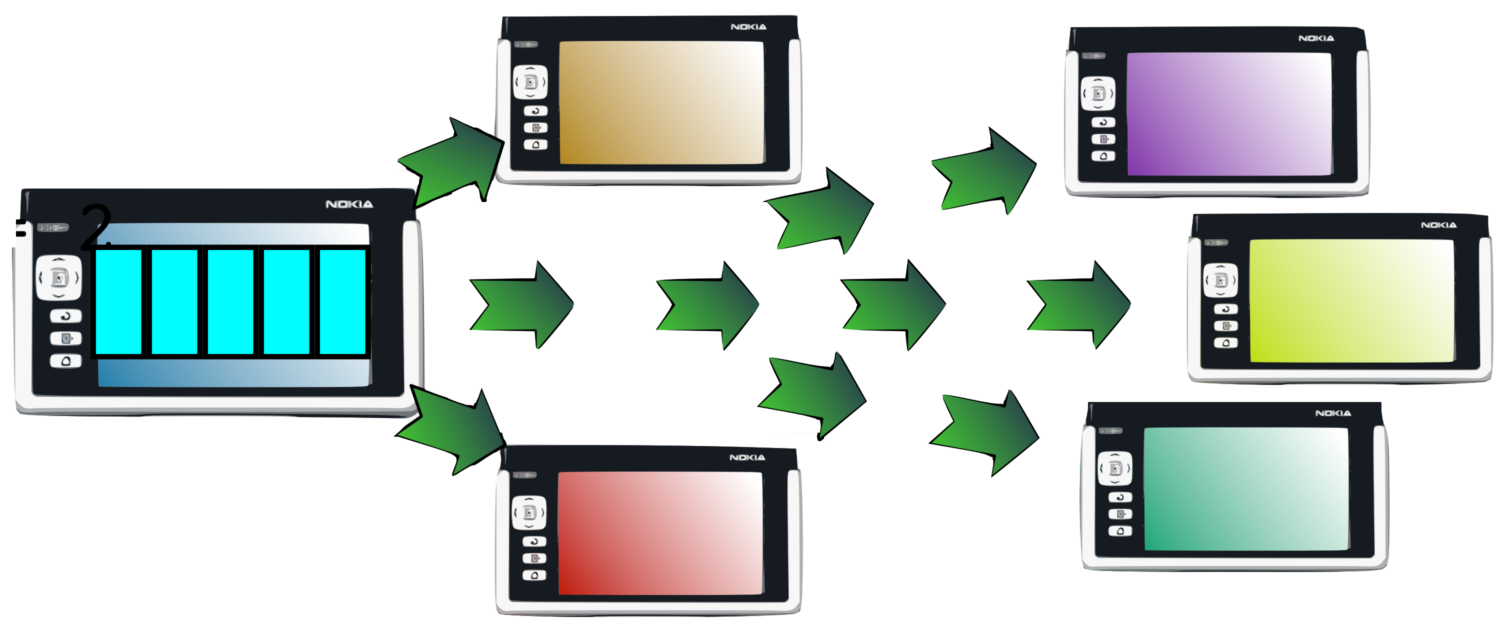
$(n,k) = (5,3)$
$f = n - k = 2$



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

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

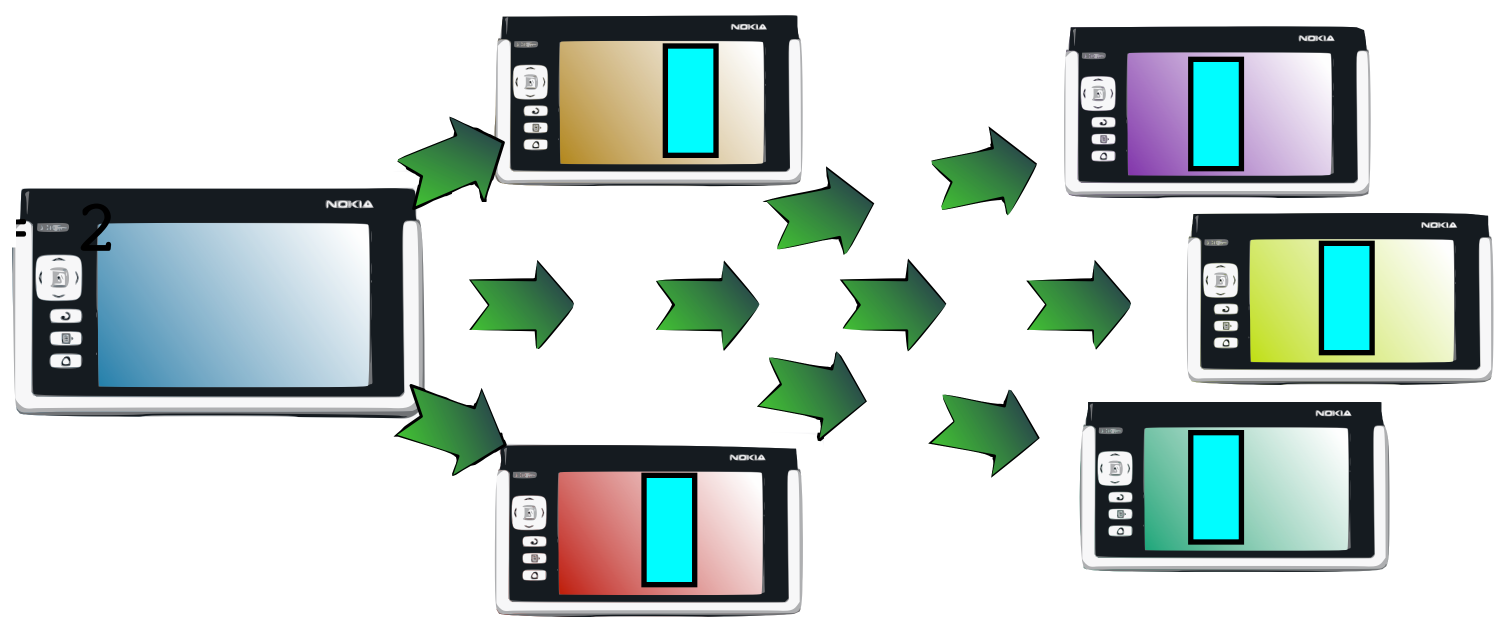


Erasure Codes

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

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

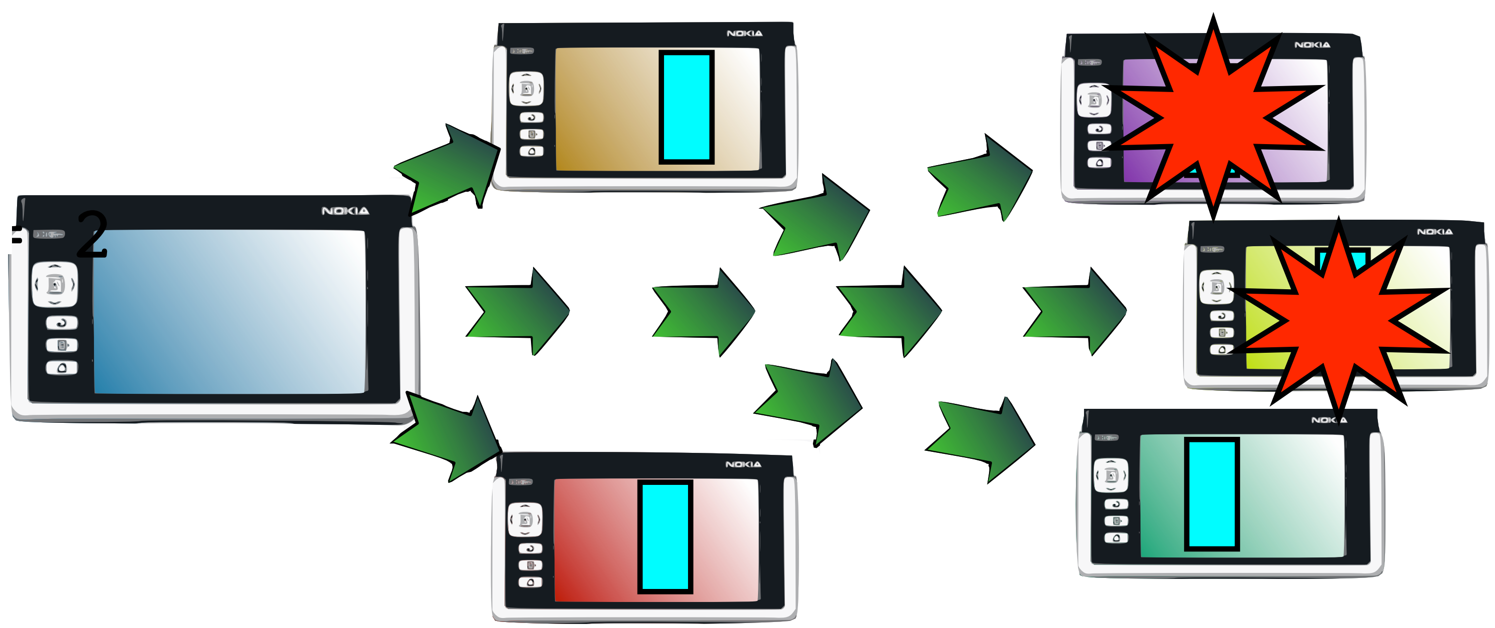


Erasure Codes

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

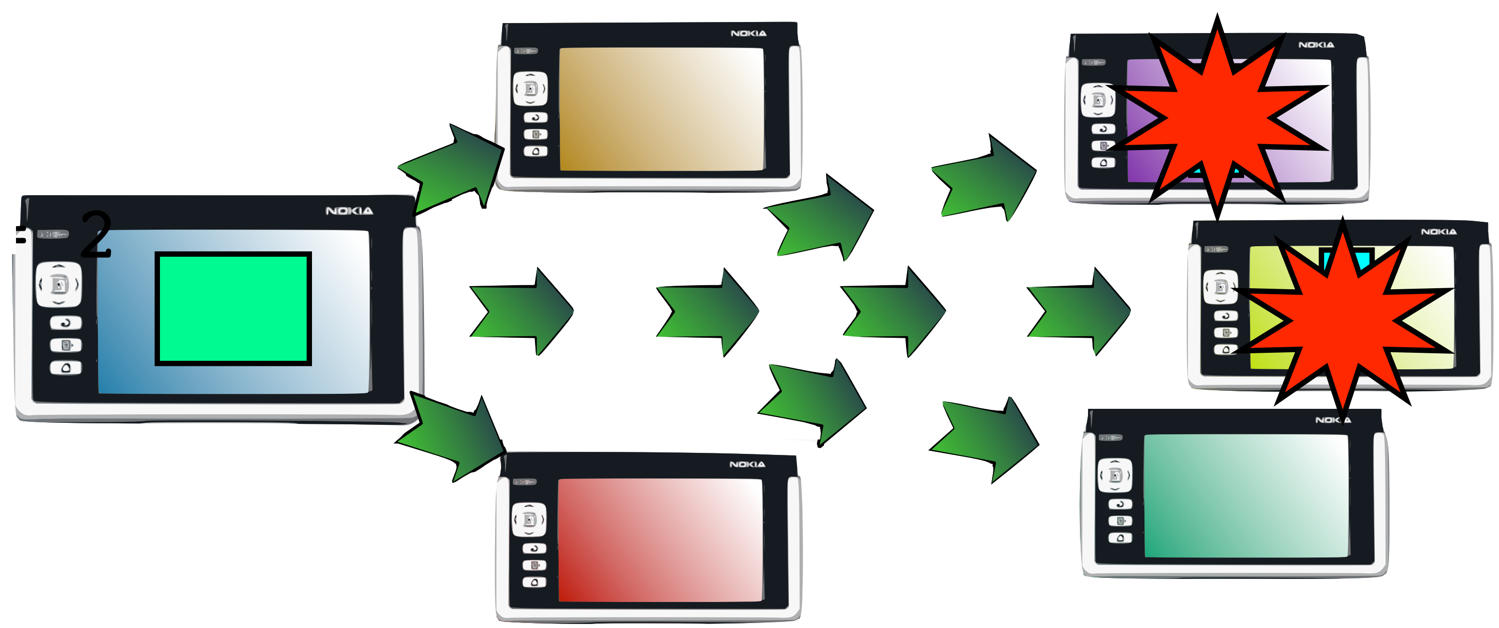
(n,k) = (5,3)
$f = n - k = 2$



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

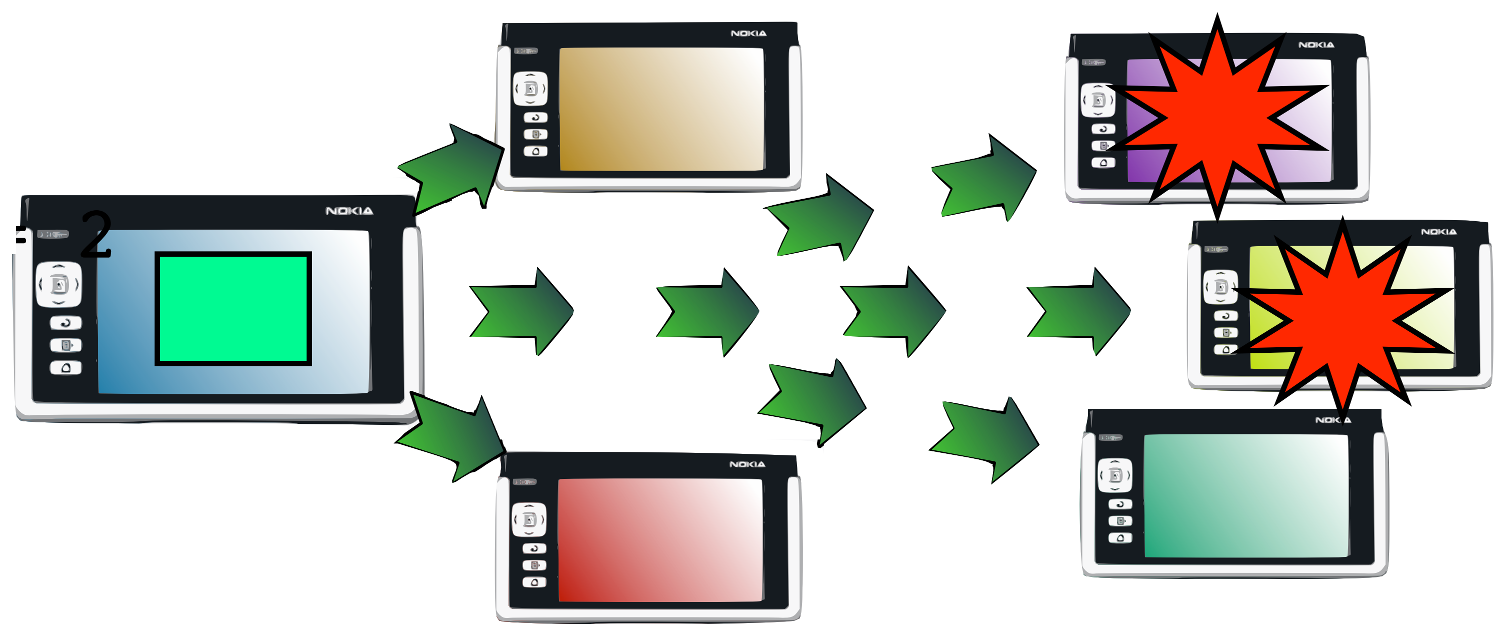
$(n,k) = (5,3)$
$f = n - k = 2$



Erasure Codes

- **Algorithm**
  1. (n,k) erasure coding $\rightarrow$ n coded blocks
  2. send 1 coded block per contributor
  3. recover from any k contributors out of n

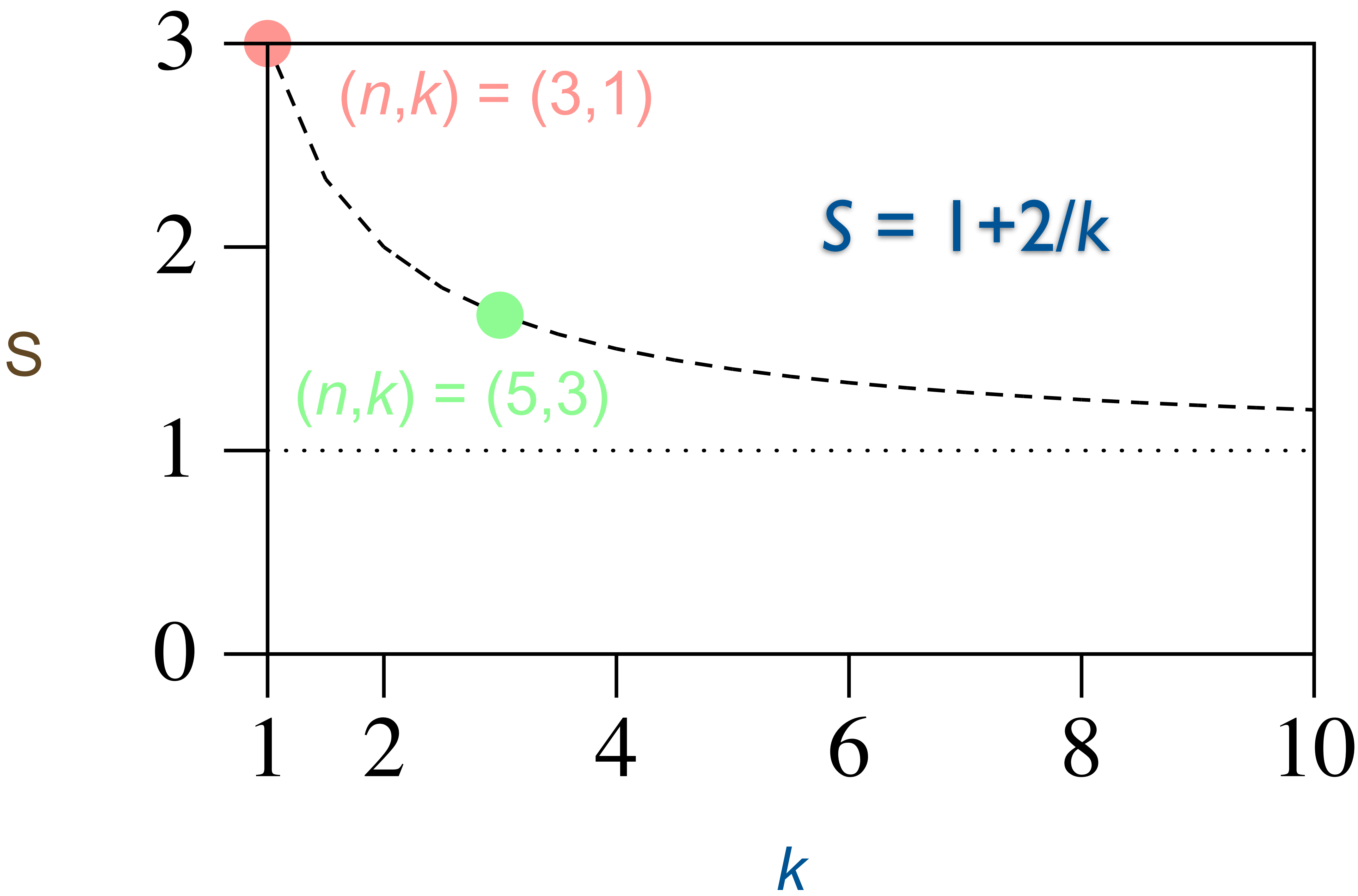
$(n,k) = (5,3)$
$f = n - k = 2$



- **Dependability & storage cost analysis**
  - tolerate f contributor faults $\Rightarrow$ storage cost $= 1 + f/k$

Erasure Codes

Storage cost for f=2



Dependability Evaluation

Dependability Evaluation

Device failure model

- crash failures
- stochastic process
- exponential distribution (rate λ)

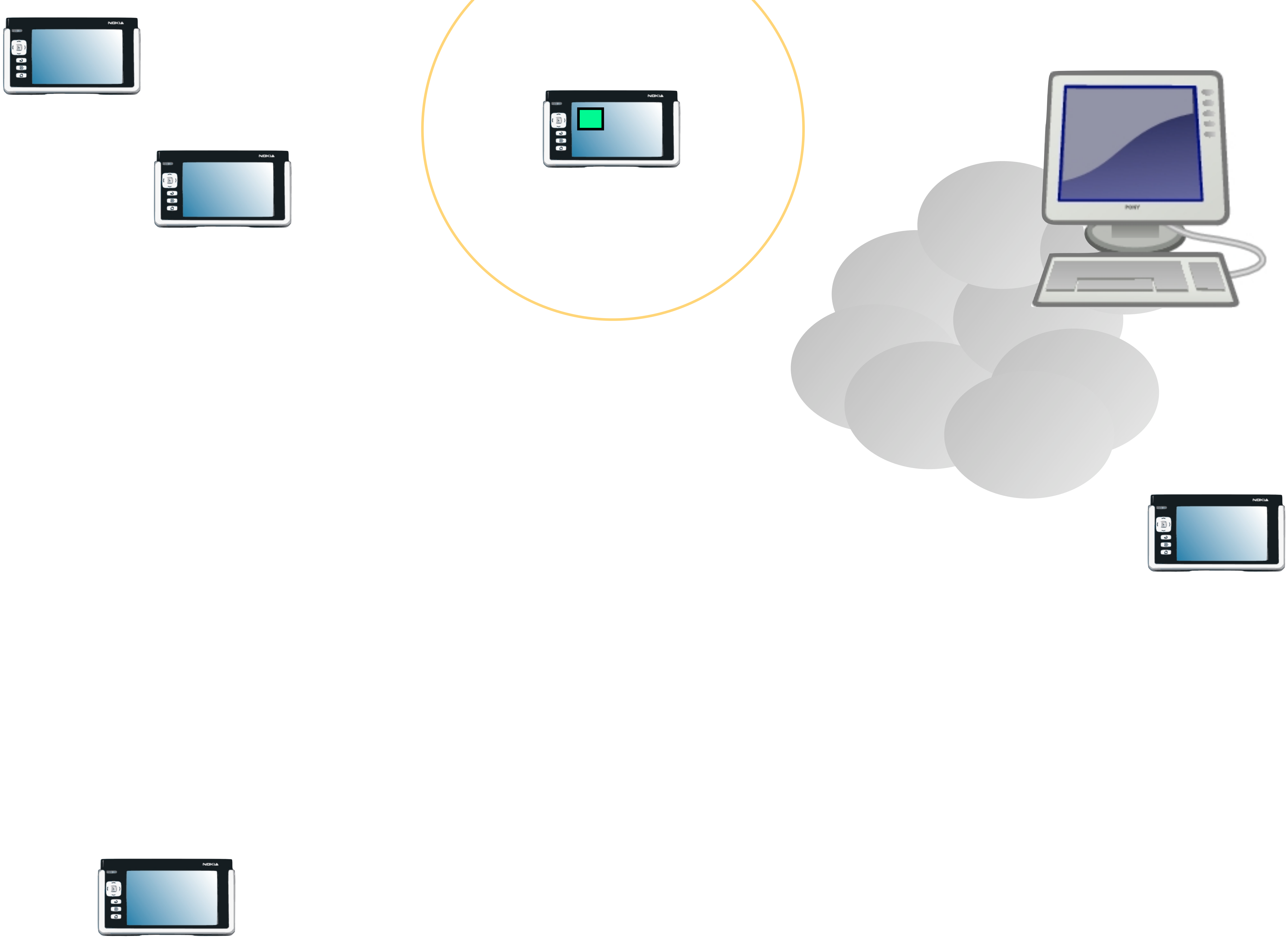
Dependability Evaluation

Device failure model

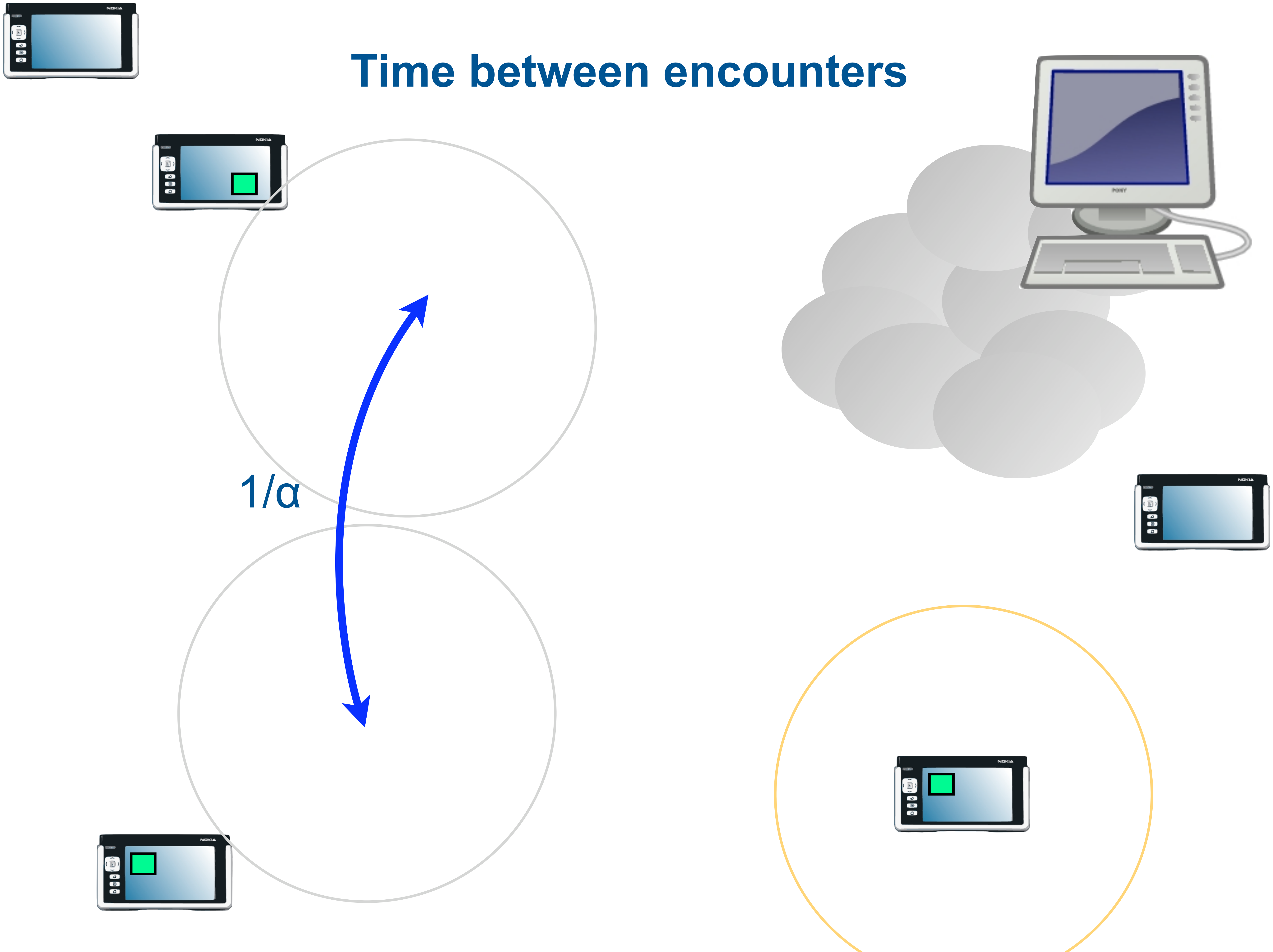
- crash failures
- stochastic process
- exponential distribution (rate λ)

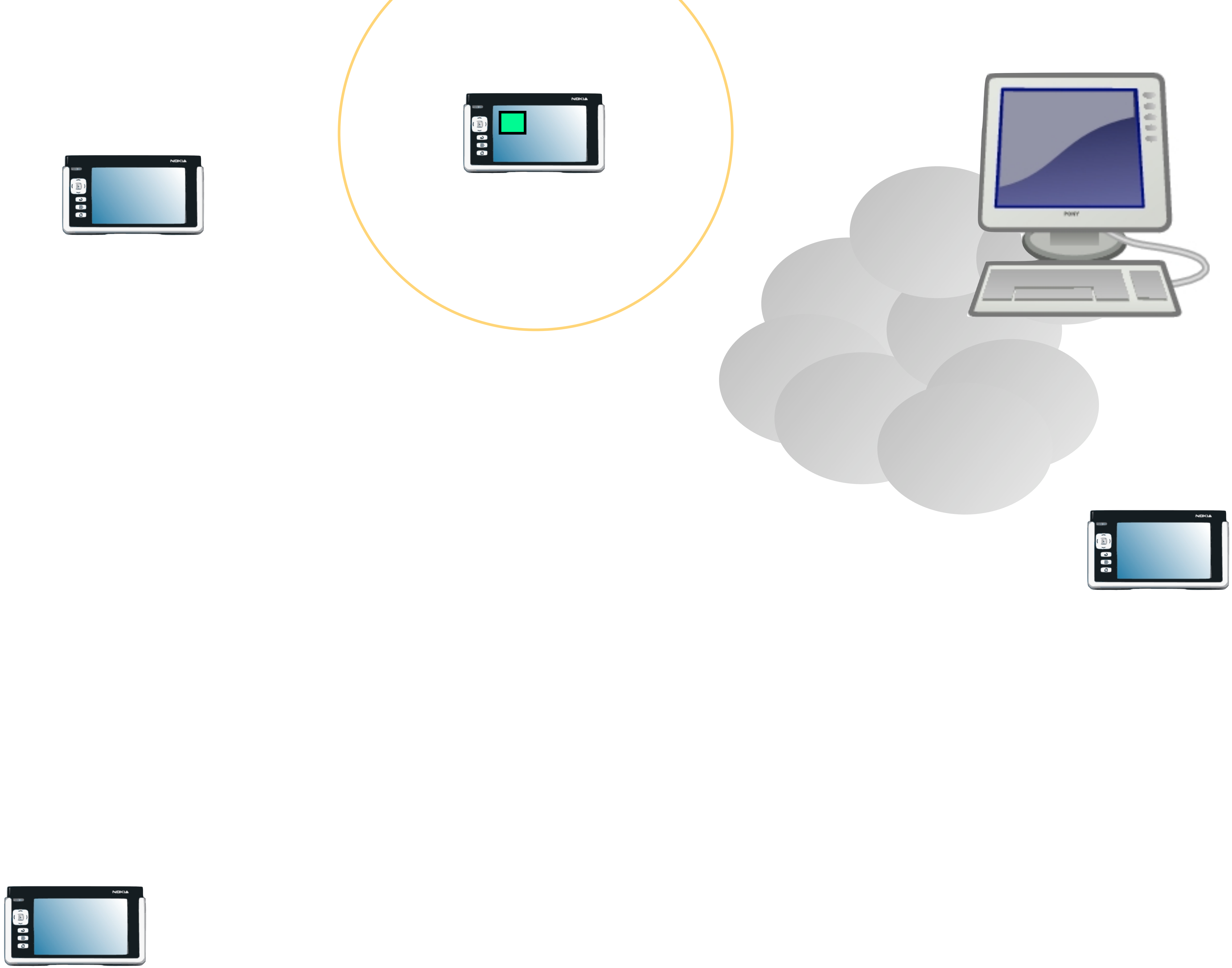
Device mobility model

- stochastic processes
- exponential distributions
  - encounters with other devices (rate α)
  - connections to Internet (rate β)

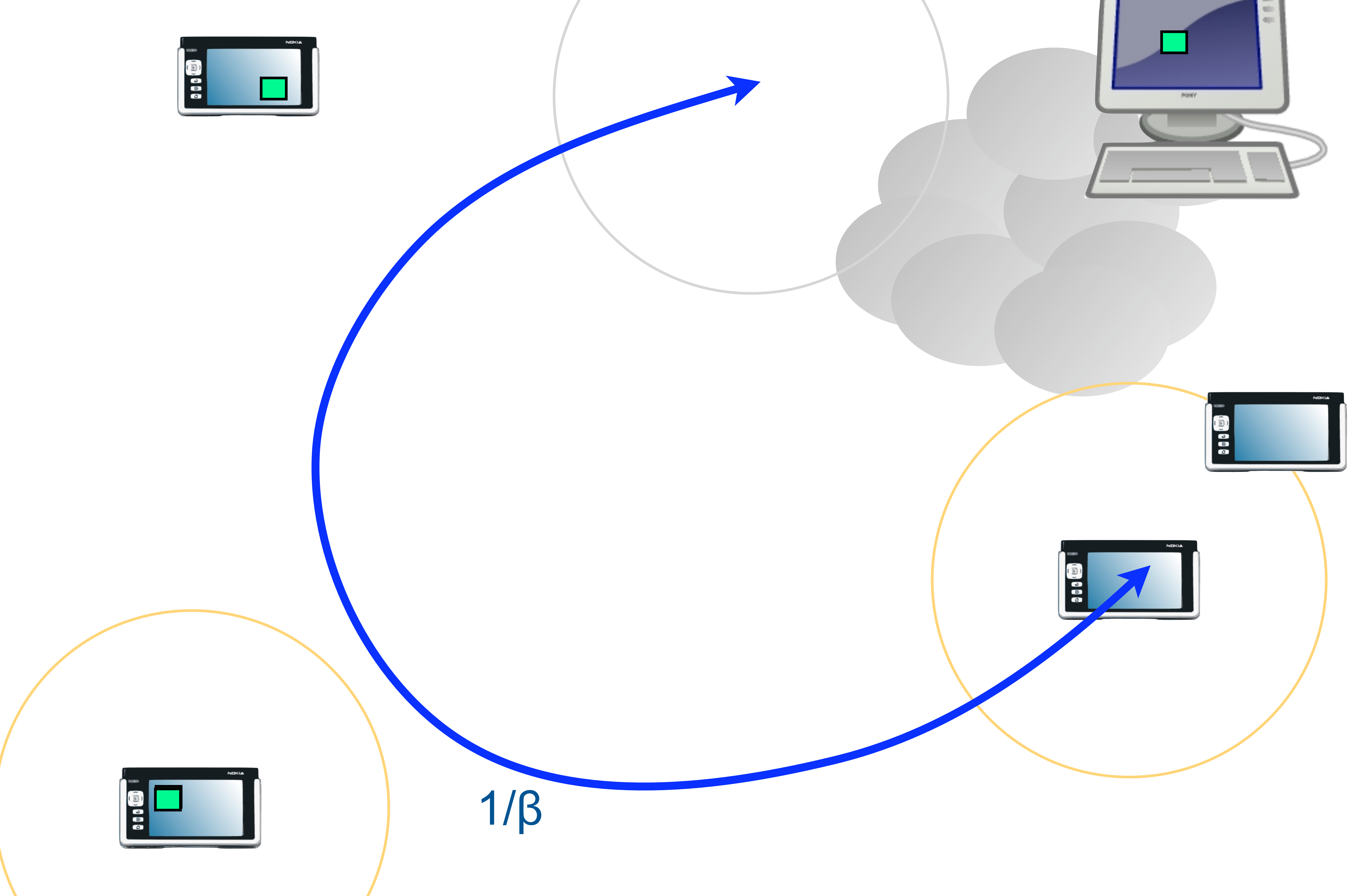


Time between encounters





Time between connections



Dependability Evaluation

- **Device failure model**
  - crash failures
  - stochastic process
  - exponential distribution (rate λ)

- **Device mobility model**
  - stochastic processes
  - exponential distributions
    - encounters with other devices (rate α)
    - connections to Internet (rate β)

Dependability Evaluation

Device failure model

- crash failures
- stochastic process
- exponential distribution (rate λ)

Device mobility model

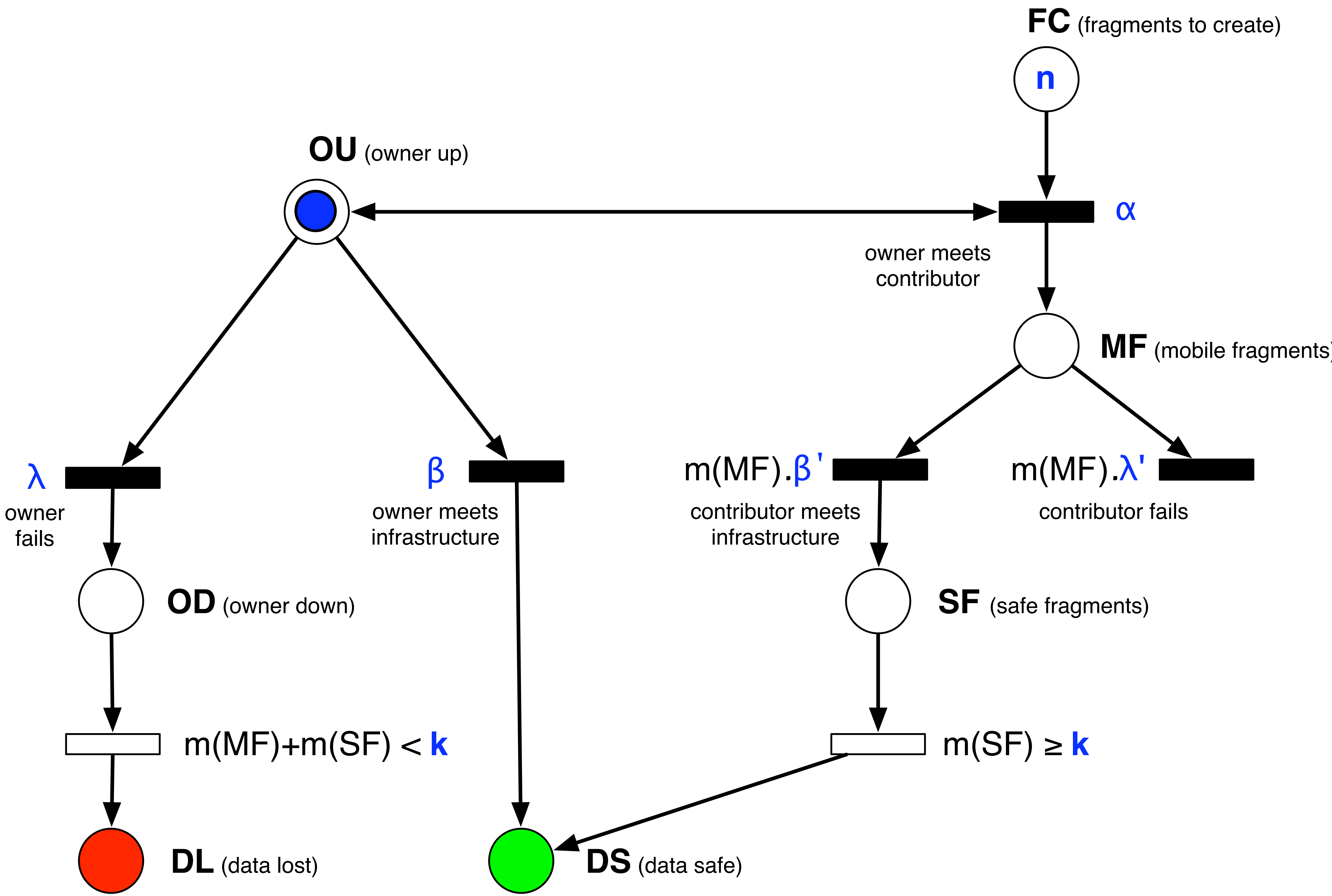
- stochastic processes
- exponential distributions
  - encounters with other devices (rate α)
  - connections to Internet (rate β)

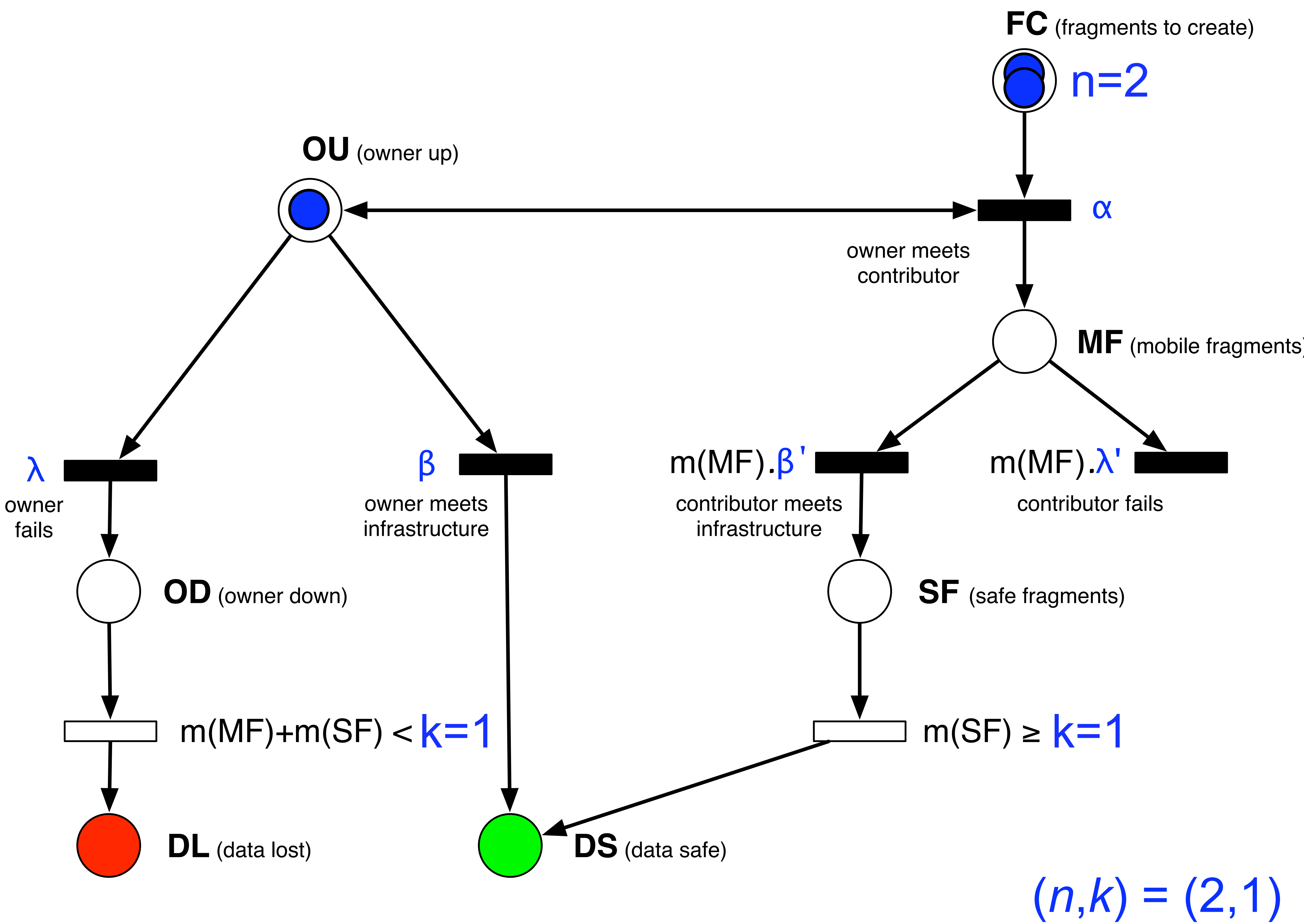
System model

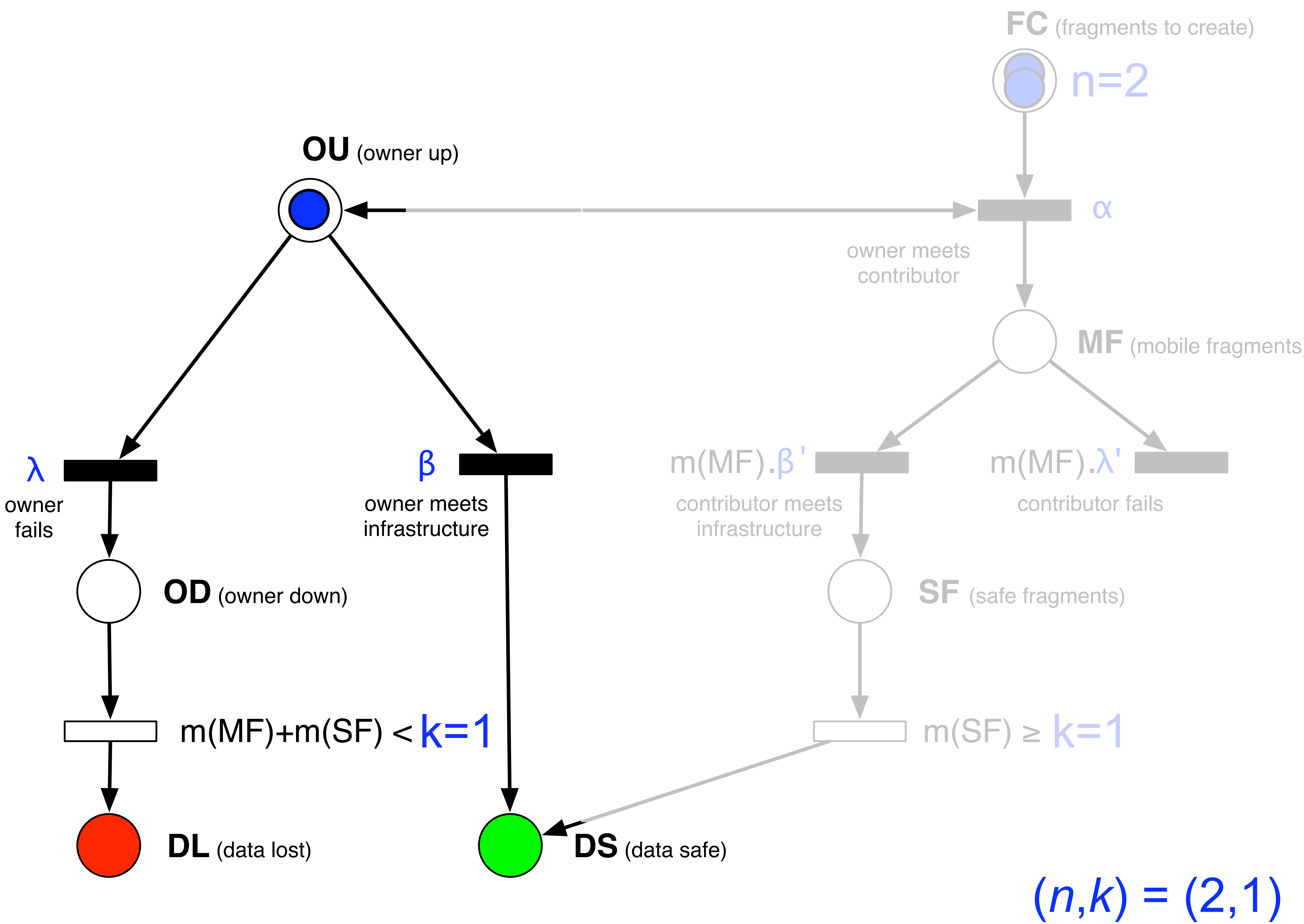
- (n,k) erasure code : up to n fragments sent to contributors
- data safe

  ⇨ original data or k fragments have reached Internet store
- data lost

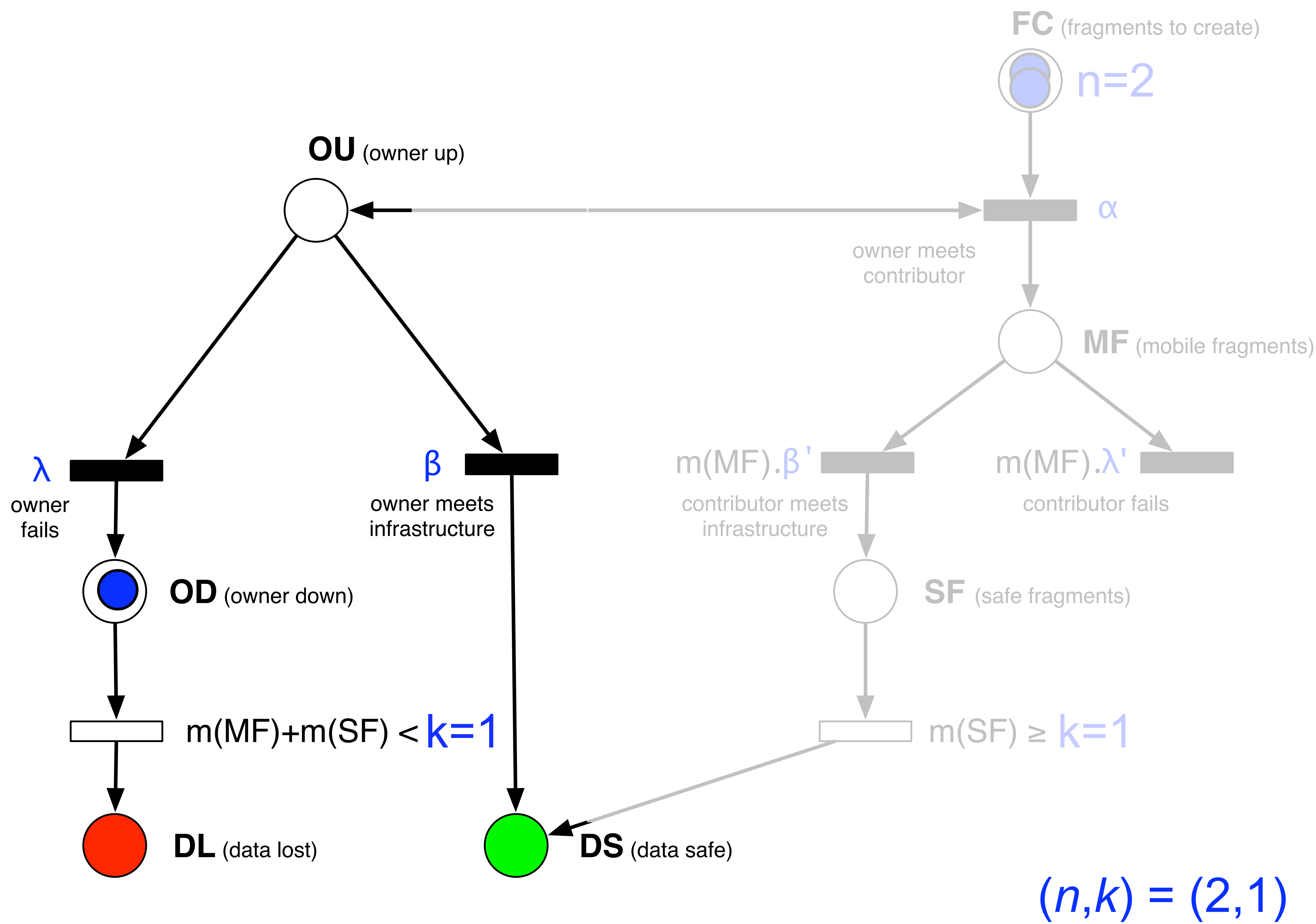
  ⇨ data owner and contributors failed before k fragments reached Internet store


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)


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)

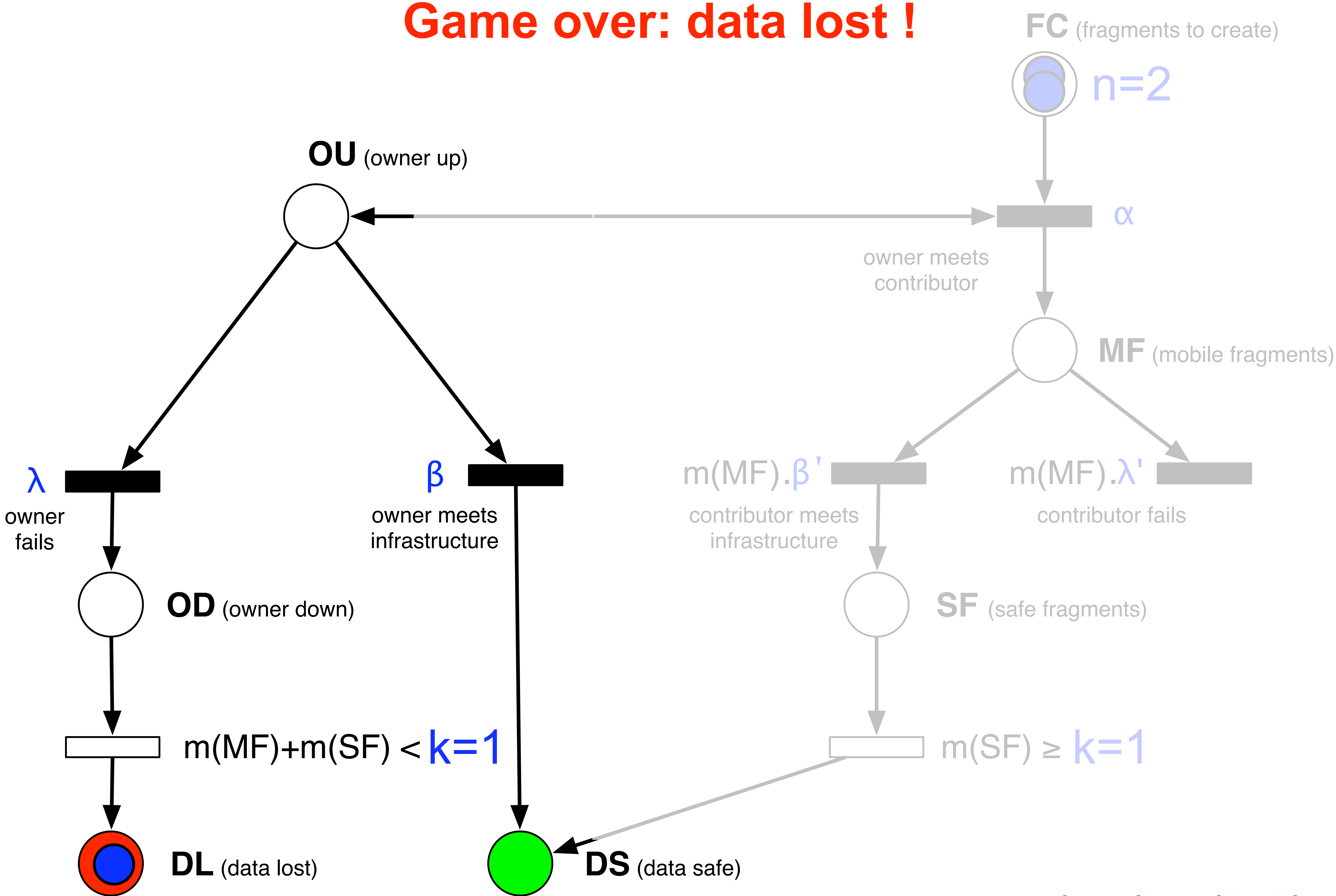


FC (fragments to create)
n=2
OU (owner up)
α
owner meets
contributor
MF (mobile fragments)
λ
owner
fails
β
owner meets
infrastructure
m(MF).β'
contributor meets
infrastructure
m(MF).λ'
contributor fails
OD (owner down)
SF (safe fragments)
m(MF)+m(SF) < k=1
m(SF) ≥ k=1
DL (data lost)
DS (data safe)
(n,k)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

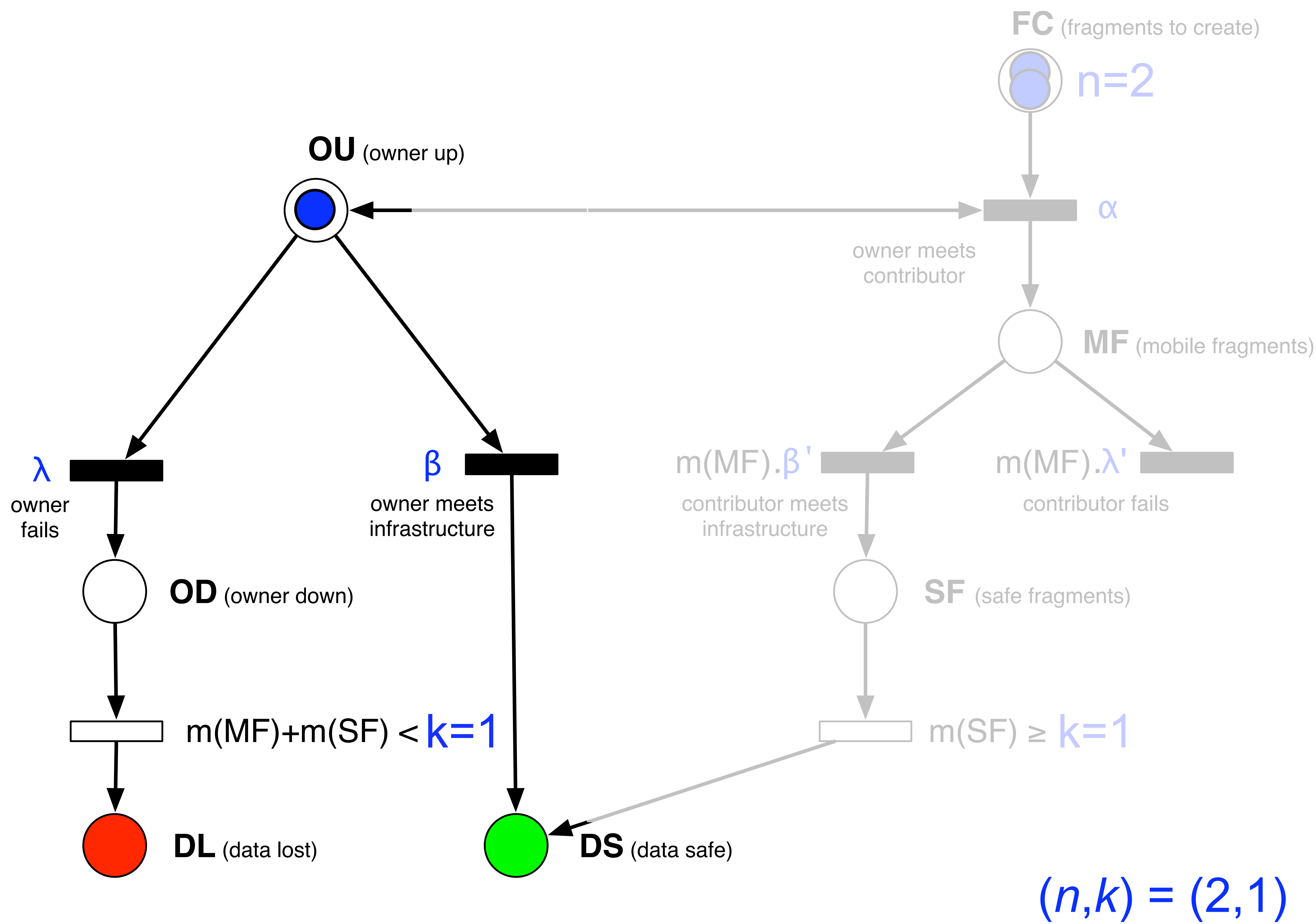


FC (fragments to create)
n=2
OU (owner up)
α
owner meets contributor
MF (mobile fragments)
λ
owner fails
β
owner meets infrastructure
m(MF).β'
contributor meets infrastructure
m(MF).λ'
contributor fails
OD (owner down)
SF (safe fragments)
m(MF)+m(SF) < k=1
m(SF) ≥ k=1
DL (data lost)
DS (data safe)
(n,k)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

Game over: data lost !

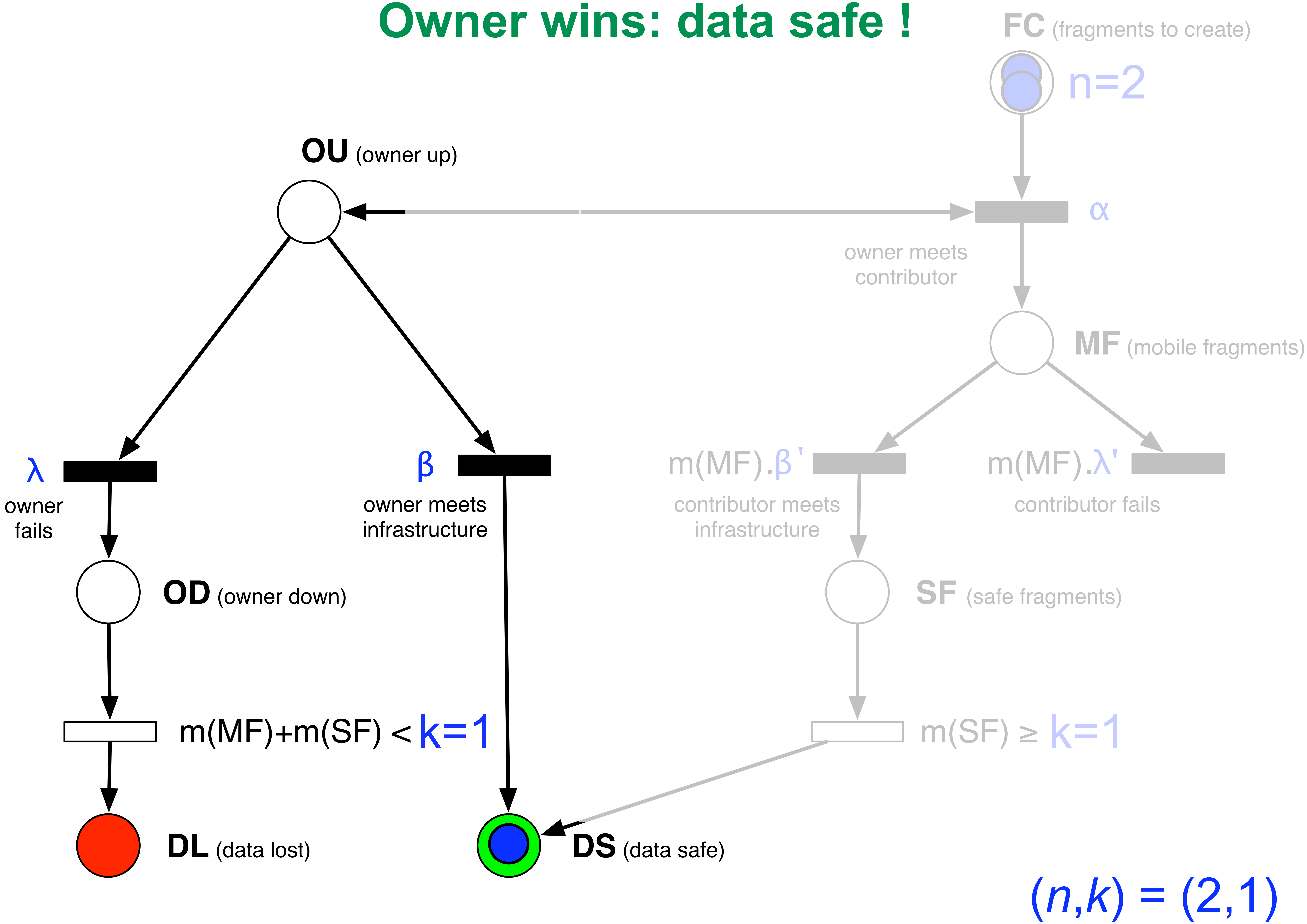


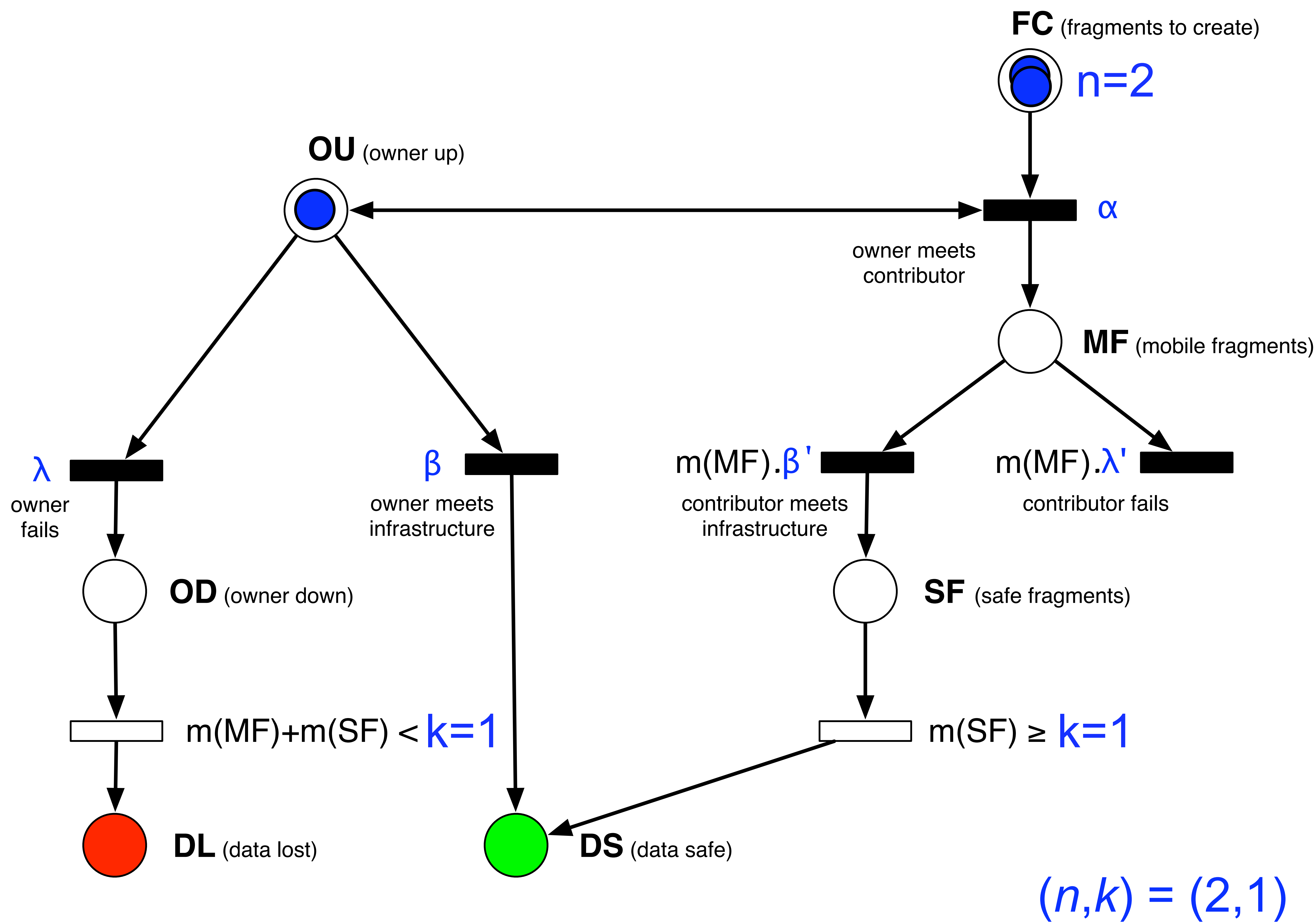
$(n,k) = (2,1)$



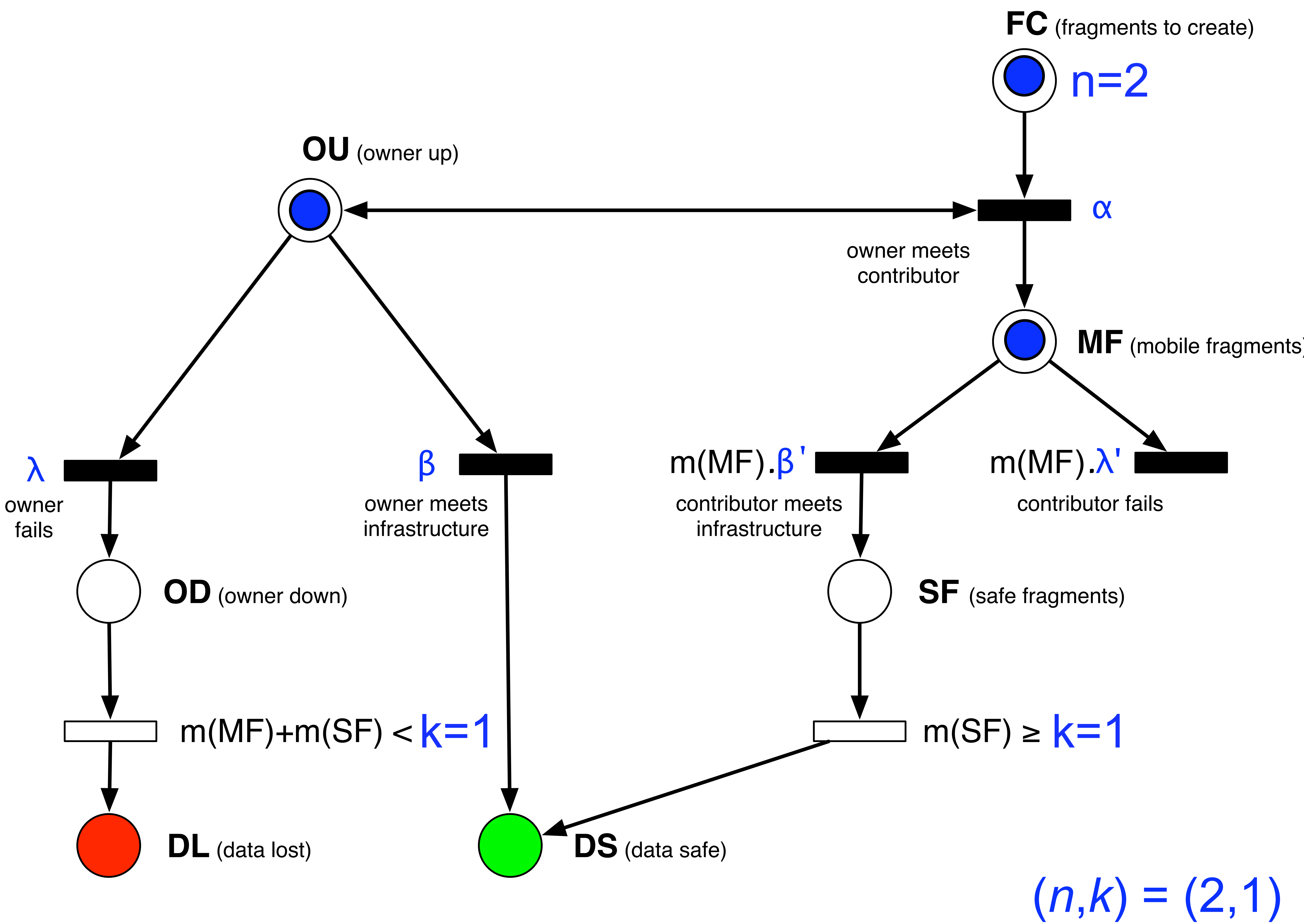
FC (fragments to create)
n=2
OU (owner up)
α
owner meets
contributor
MF (mobile fragments)
λ
owner
fails
β
owner meets
infrastructure
m(MF).β'
contributor meets
infrastructure
m(MF).λ'
contributor fails
OD (owner down)
SF (safe fragments)
m(MF)+m(SF) < k=1
m(SF) ≥ k=1
DL (data lost)
DS (data safe)
(n,k)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

Owner wins: data safe !



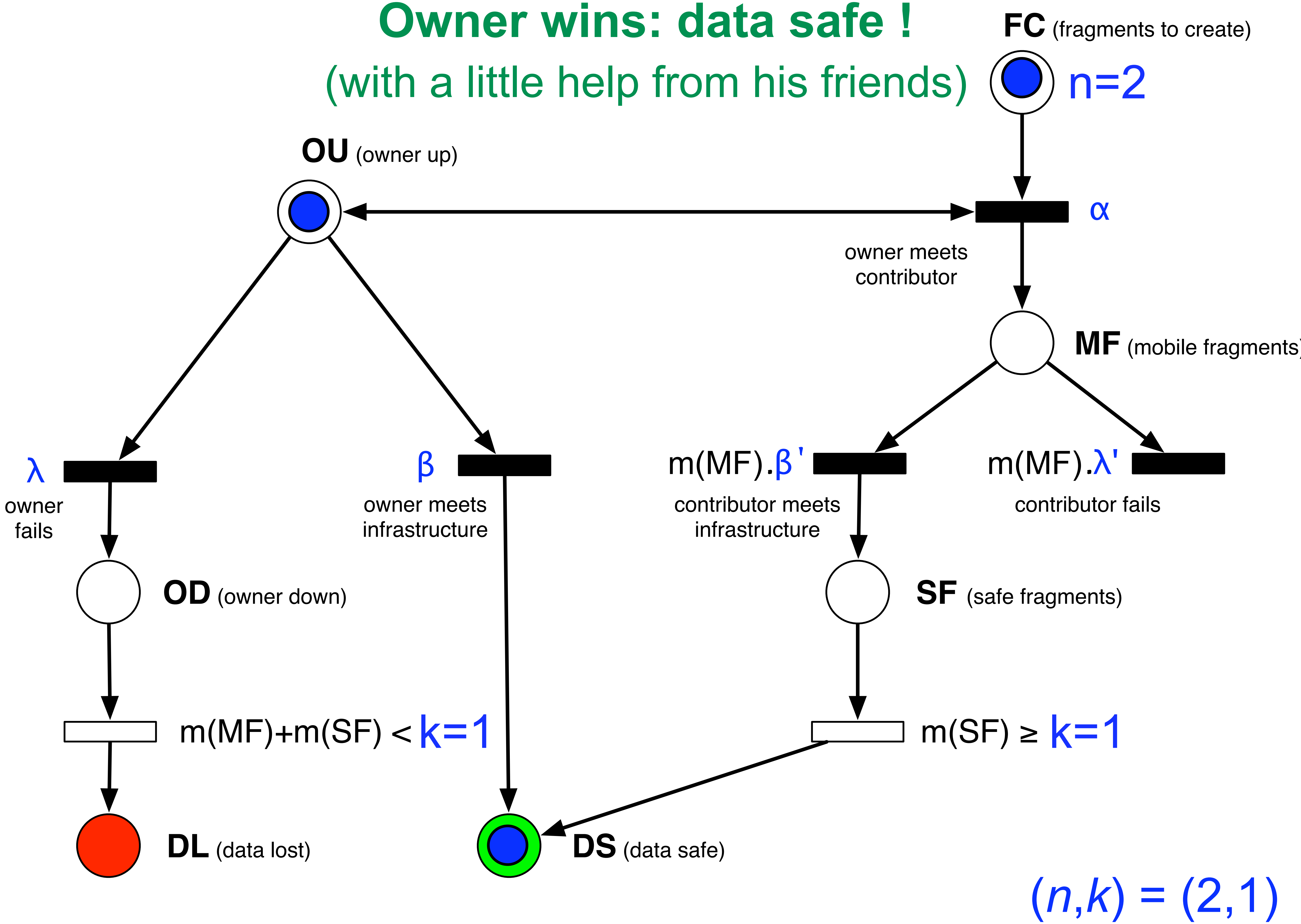


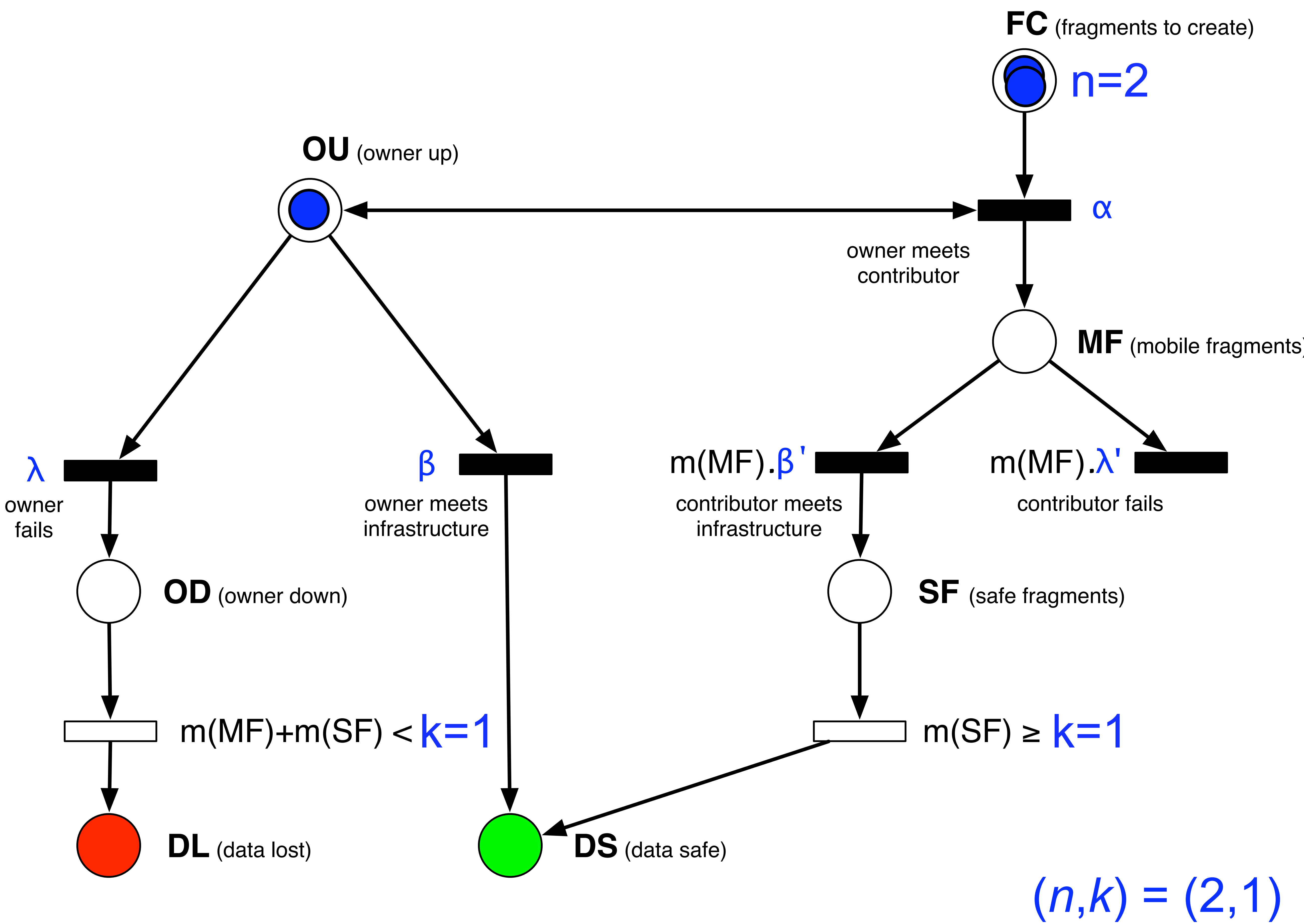
FC (fragments to create)
n=2
OU (owner up)
α
owner meets contributor
MF (mobile fragments)
λ
owner fails
β
owner meets infrastructure
m(MF).β'
contributor meets infrastructure
m(MF).λ'
contributor fails
OD (owner down)
SF (safe fragments)
m(MF)+m(SF) < k=1
m(SF) ≥ k=1
DL (data lost)
DS (data safe)
(n,k)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

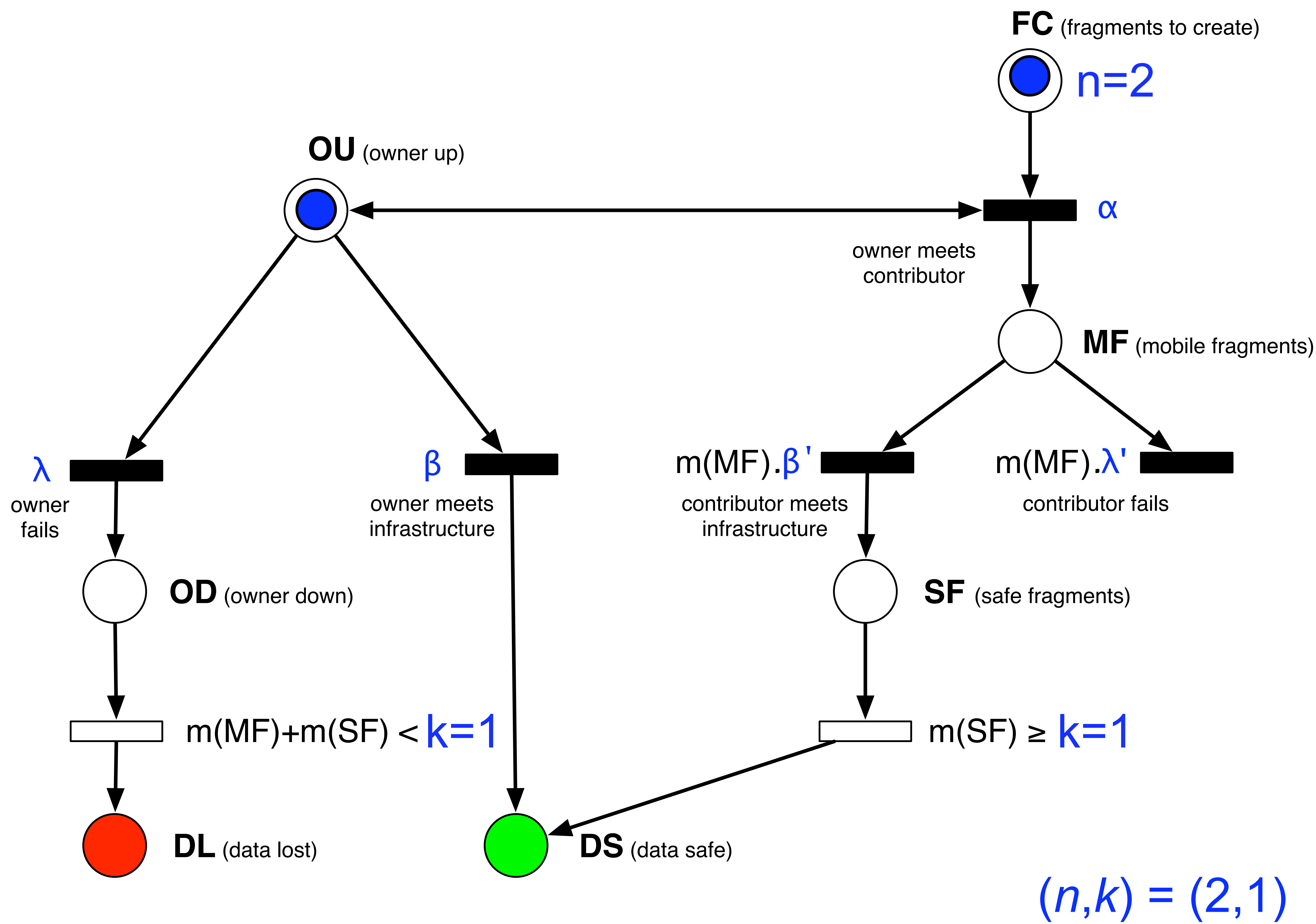

FC (fragments to create)
n=2
OU (owner up)
α
owner meets contributor
MF (mobile fragments)
λ
owner fails
β
owner meets infrastructure
m(MF).β'
contributor meets infrastructure
m(MF).λ'
contributor fails
OD (owner down)
SF (safe fragments)
m(MF)+m(SF) < k=1
m(SF) ≥ k=1
DL (data lost)
DS (data safe)
(n,k)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

Owner wins: data safe !

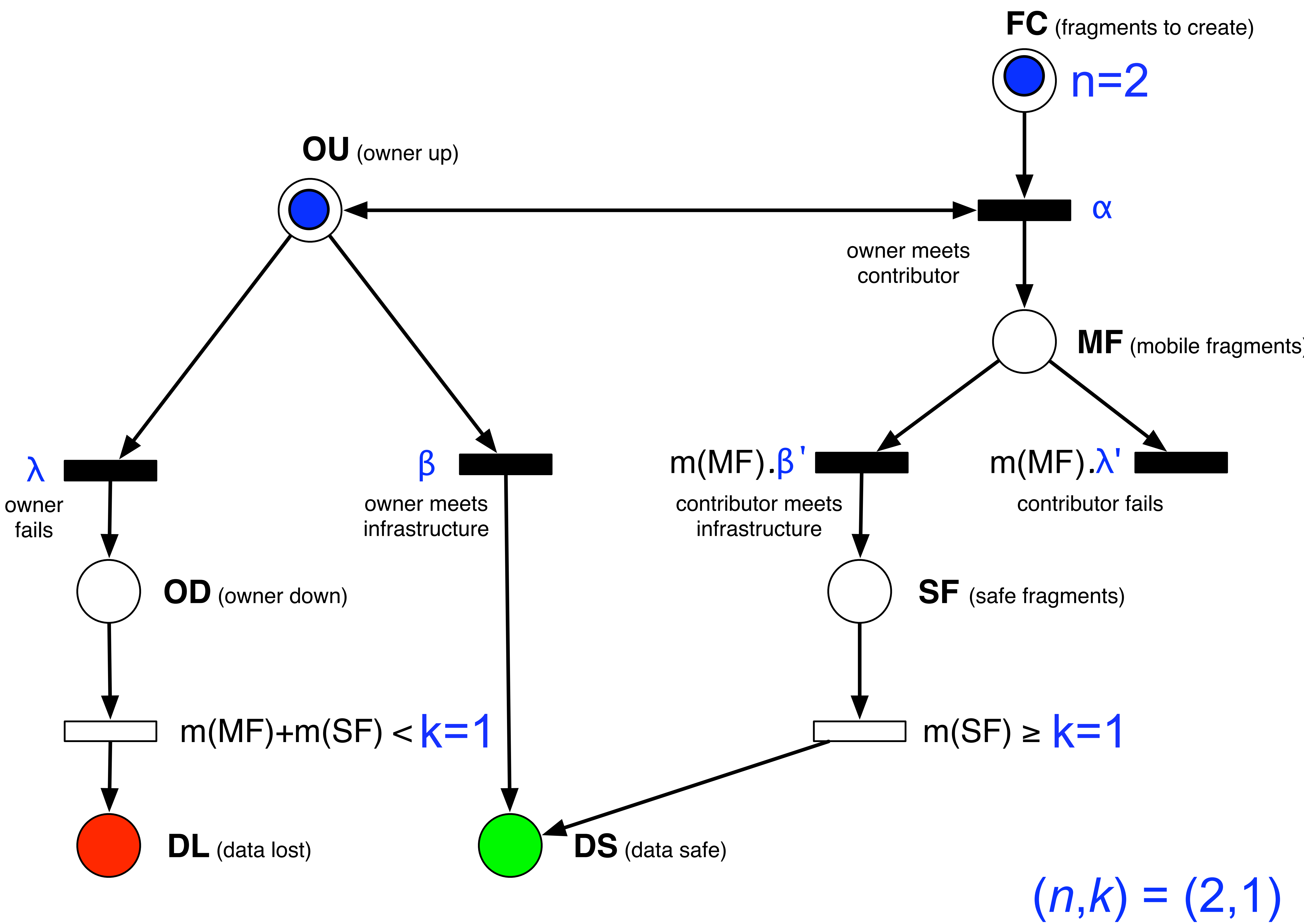
(with a little help from his friends)



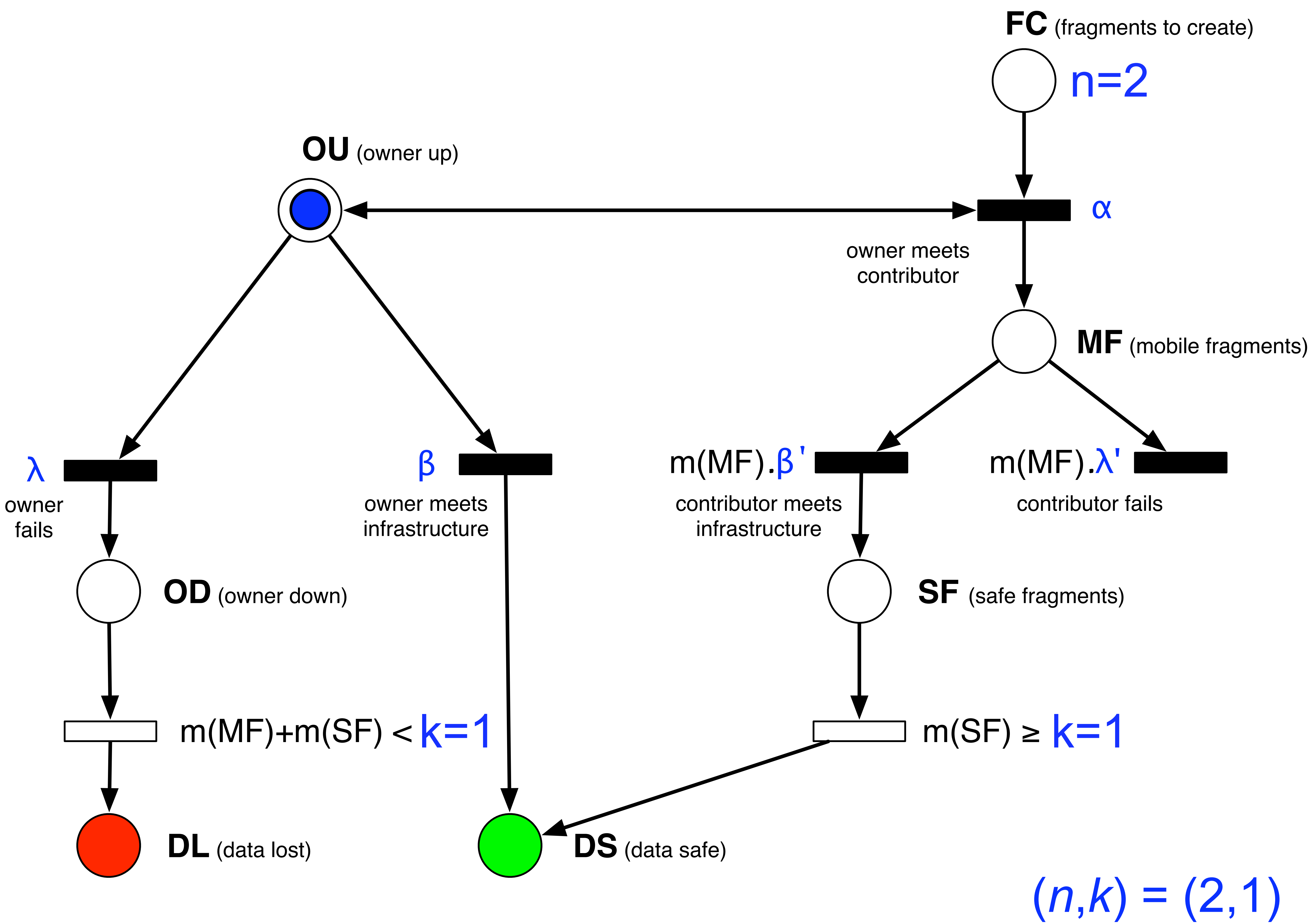

FC (fragments to create)
n=2
OU (owner up)
α
owner meets contributor
MF (mobile fragments)
λ
owner fails
β
owner meets infrastructure
m(MF).β'
contributor meets infrastructure
m(MF).λ'
contributor fails
OD (owner down)
SF (safe fragments)
m(MF)+m(SF) < k=1
m(SF) ≥ k=1
DL (data lost)
DS (data safe)
(n,k)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



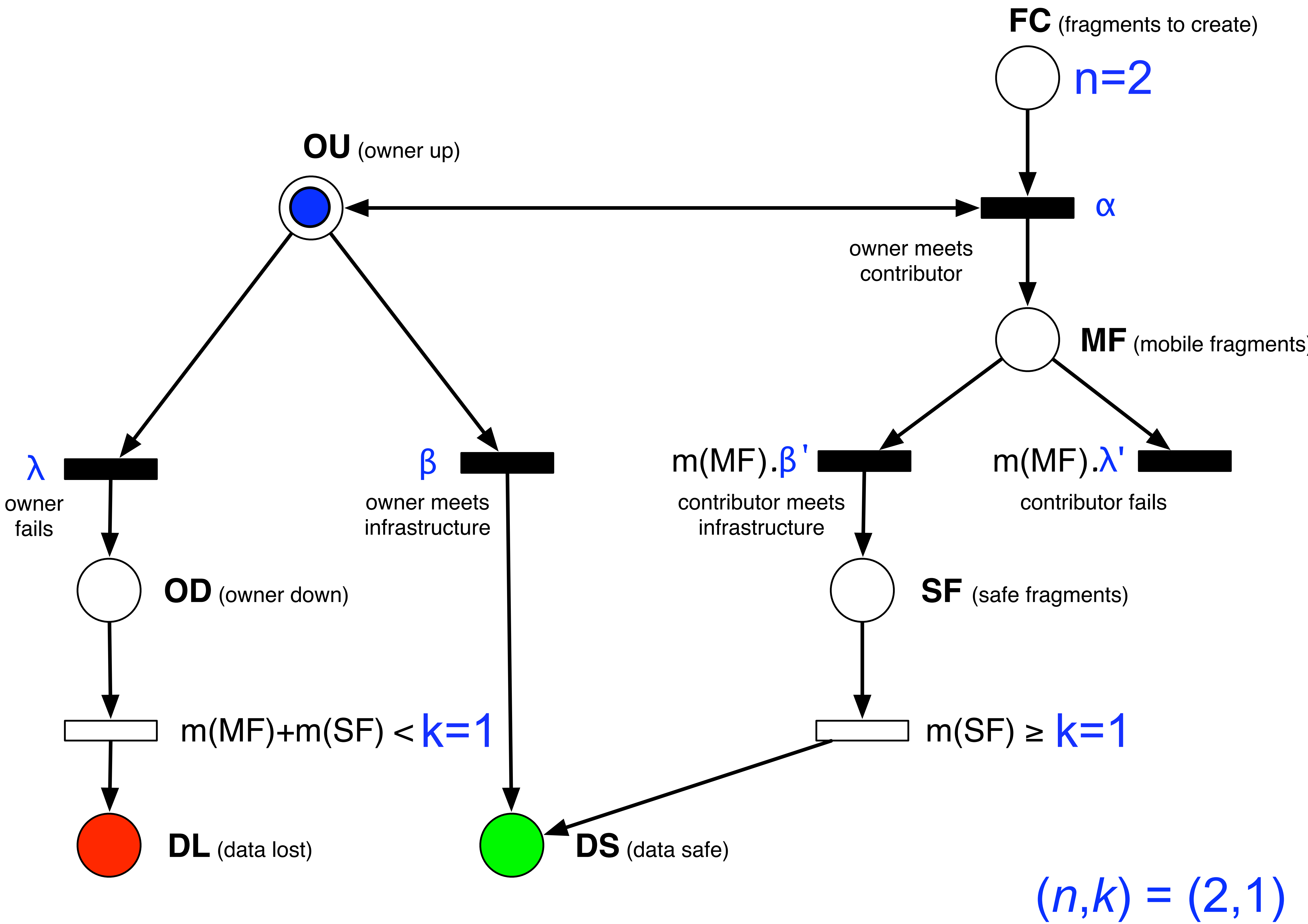
FC (fragments to create)
n=2
OU (owner up)
α
owner meets
contributor
MF (mobile fragments)
λ
owner
fails
β
owner meets
infrastructure
m(MF).β'
contributor meets
infrastructure
m(MF).λ'
contributor fails
OD (owner down)
SF (safe fragments)
m(MF)+m(SF) < k=1
m(SF) ≥ k=1
DL (data lost)
DS (data safe)
(n,k)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



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)



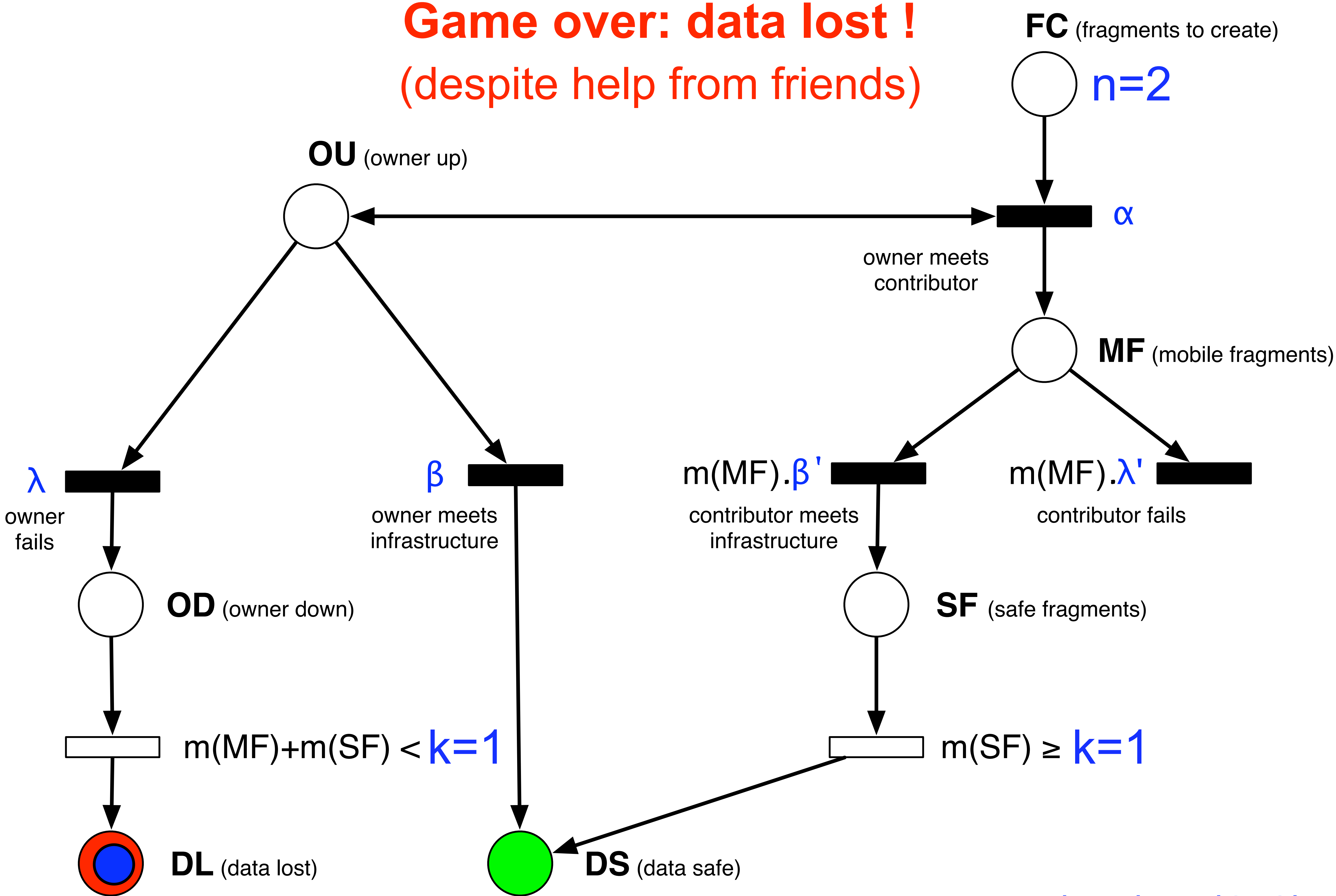
FC (fragments to create)
n=2
OU (owner up)
α
owner meets
contributor
MF (mobile fragments)
λ
owner
fails
β
owner meets
infrastructure
m(MF).β'
contributor meets
infrastructure
m(MF).λ'
contributor fails
OD (owner down)
SF (safe fragments)
m(MF)+m(SF) < k=1
m(SF) ≥ k=1
DL (data lost)
DS (data safe)
(n,k)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



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)

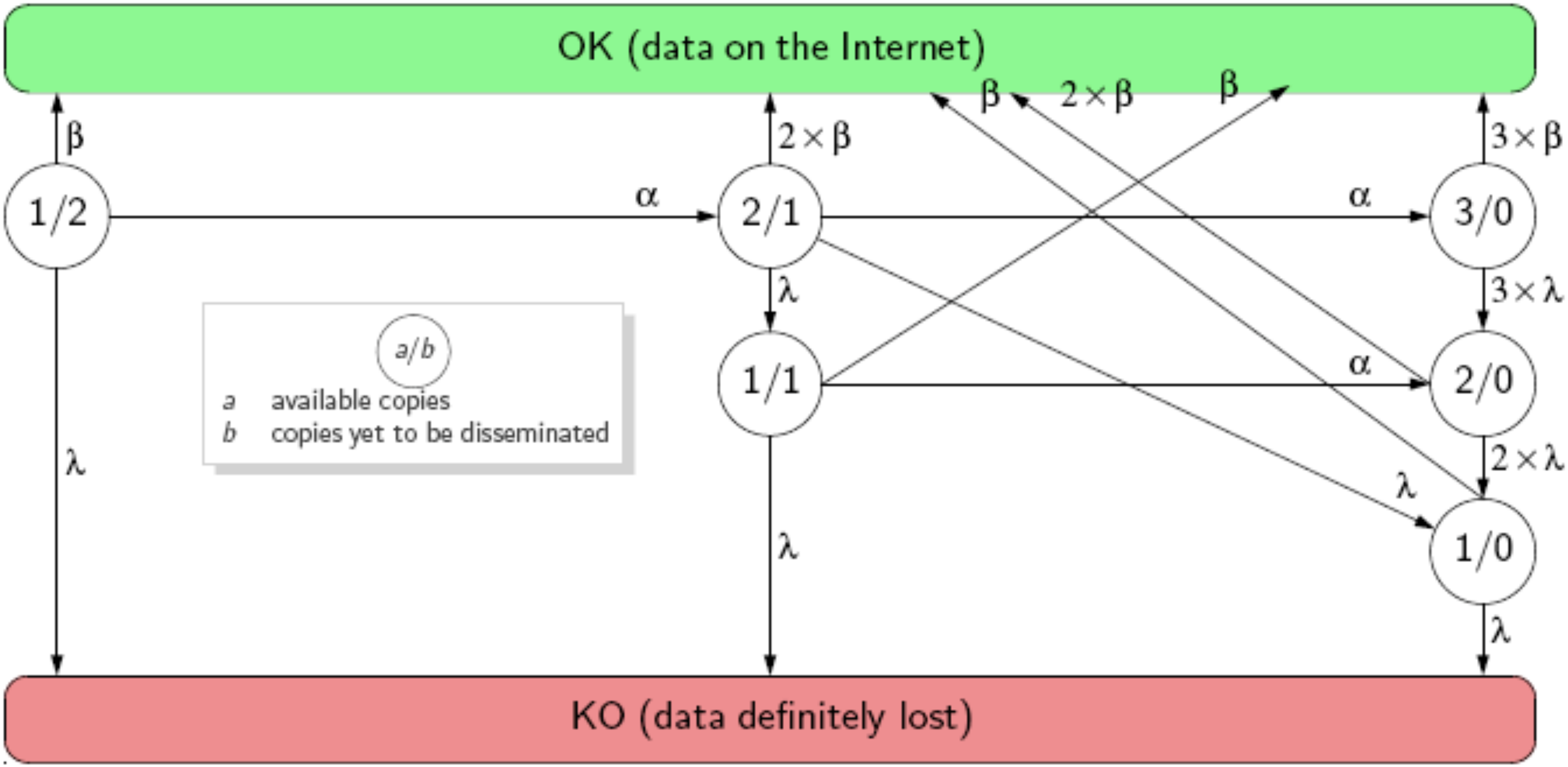
Game over: data lost !

(despite help from friends)



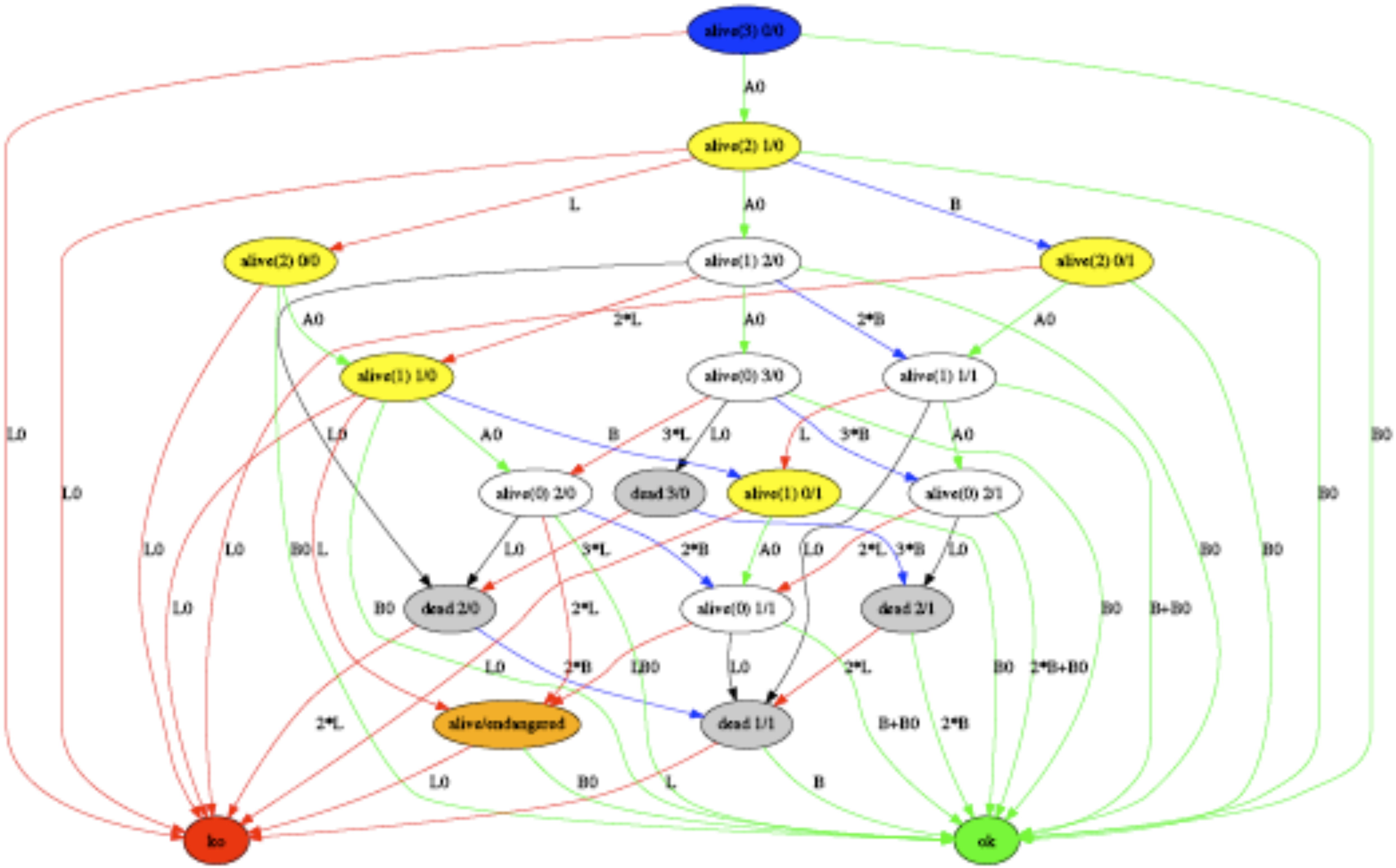
$(n,k) = (2,1)$

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

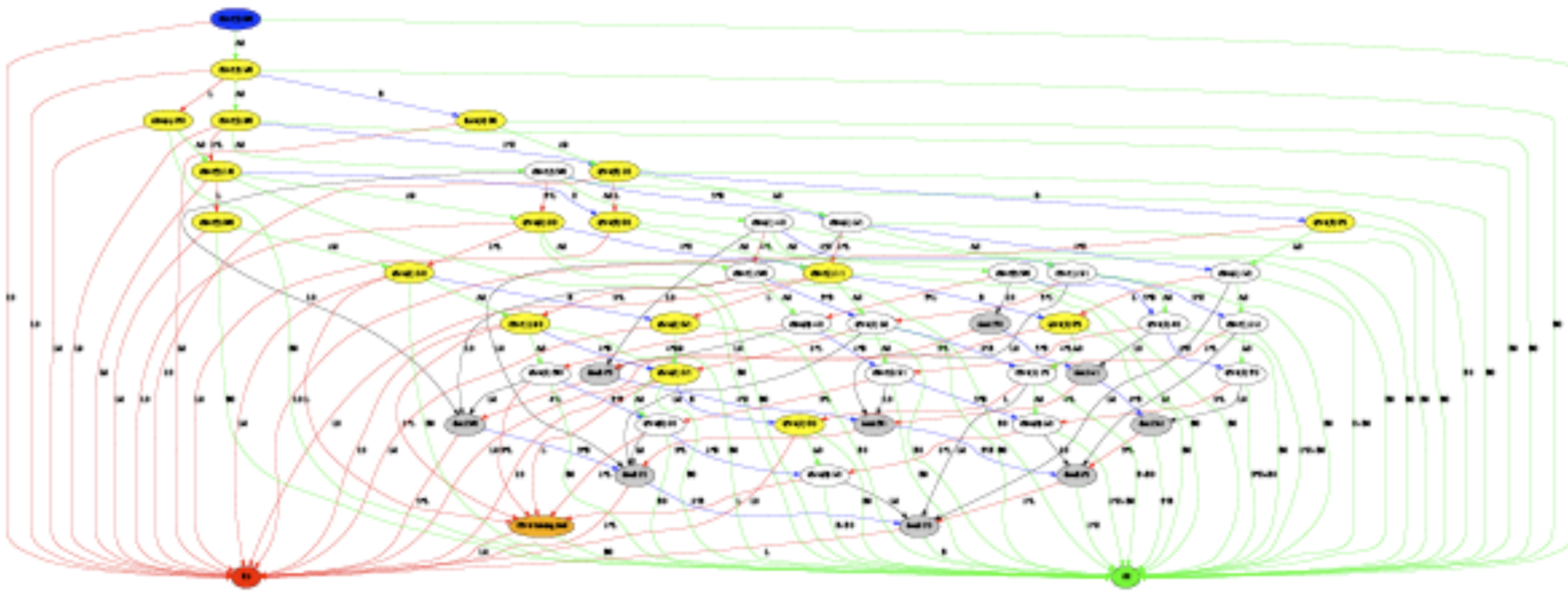


OK (data on the Internet)
β
2 × β
β
3 × β
β
2 × β
β
1/2
α
2/1
α
3/0
λ
3 × λ
a/b
a available copies
b copies yet to be disseminated
1/1
α
2/0
λ
2 × λ
λ
λ
1/0
λ
KO (data definitely lost)

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


alive(3) 0/0
alive(2) 1/0
alive(2) 0/0
alive(1) 2/0
alive(2) 0/1
alive(1) 1/0
alive(0) 3/0
alive(1) 1/1
alive(0) 2/0
dead 3/0
alive(1) 0/1
alive(0) 2/1
dead 2/0
alive(0) 1/1
dead 2/1
alive/endangered
dead 1/1
ko
ok

(n,k) = (5,3)



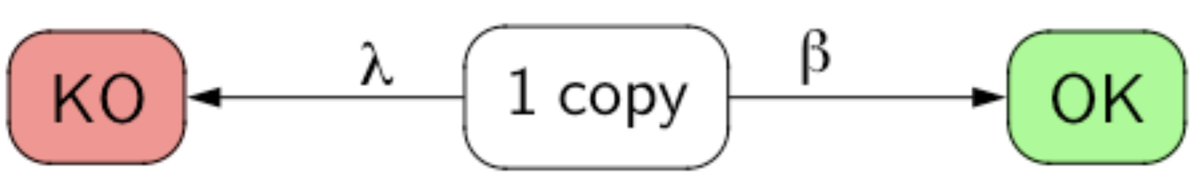
Dependability Measurements

Dependability Measurements

- **PL: probability of data loss**
  - Probability of data owner and contributors failing before sufficient fragments have reached Internet store

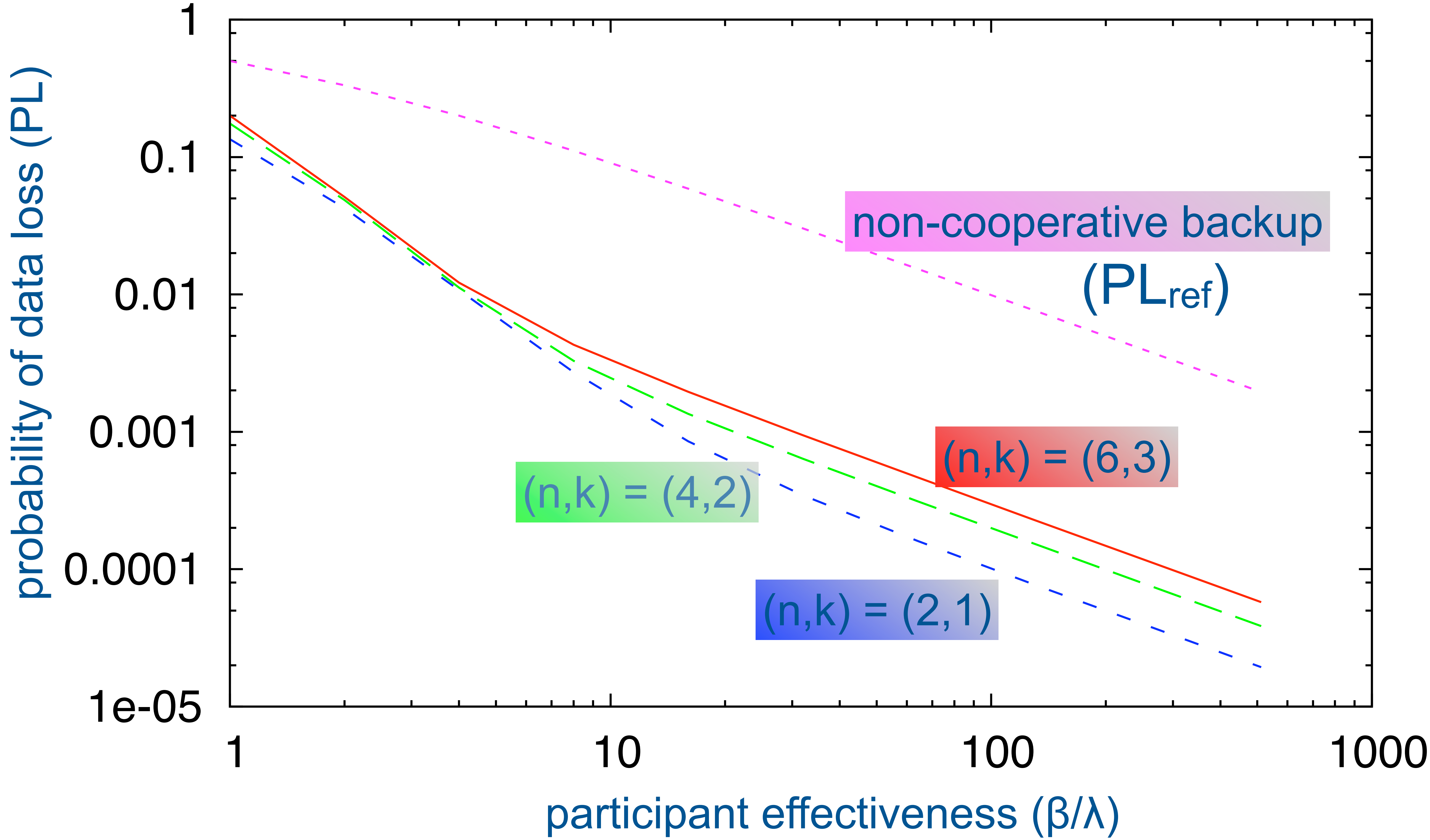
Dependability Measurements

- **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

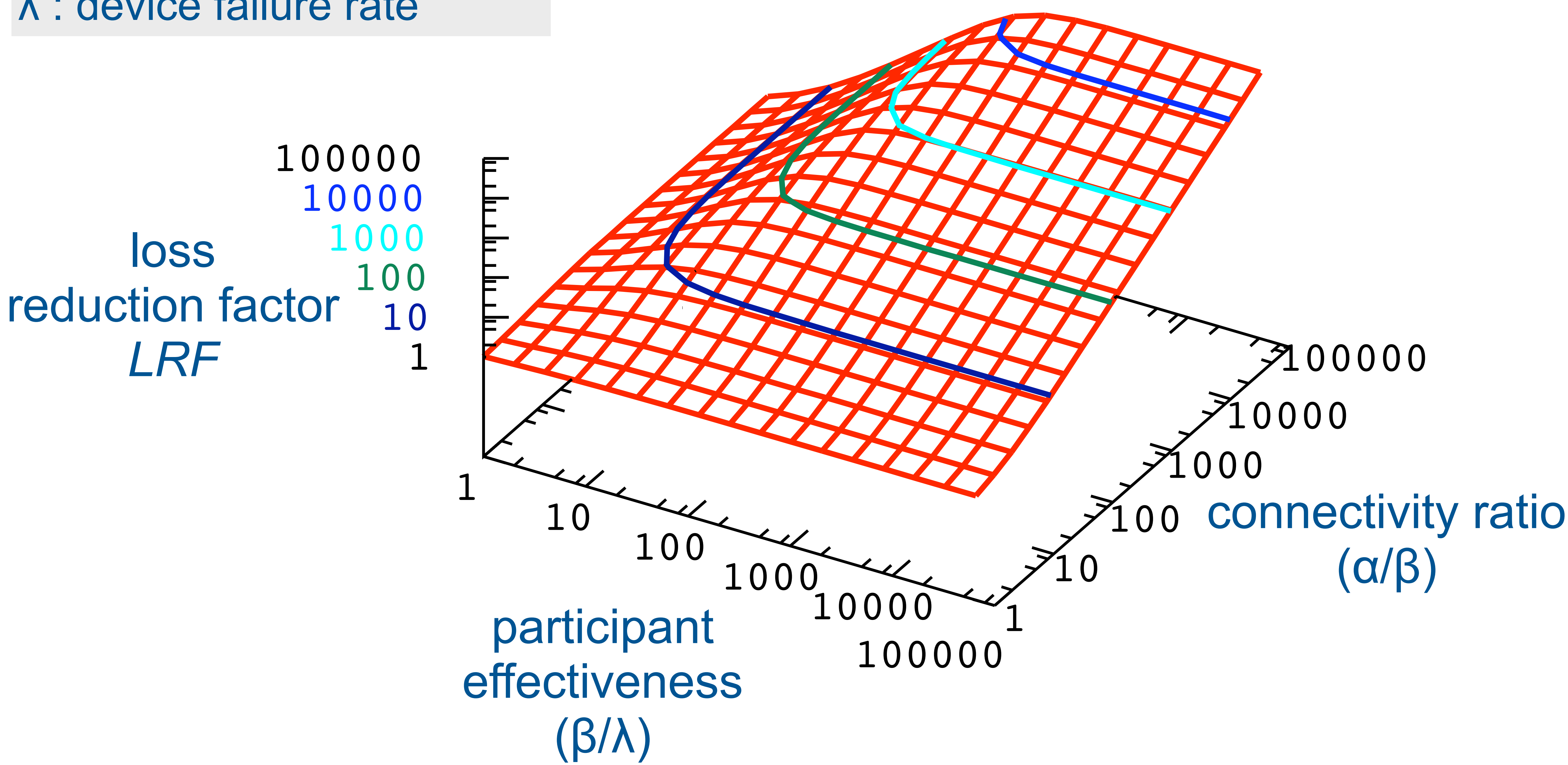
(connectivity ratio α/β = 100)



LRF vs. basic parameters

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

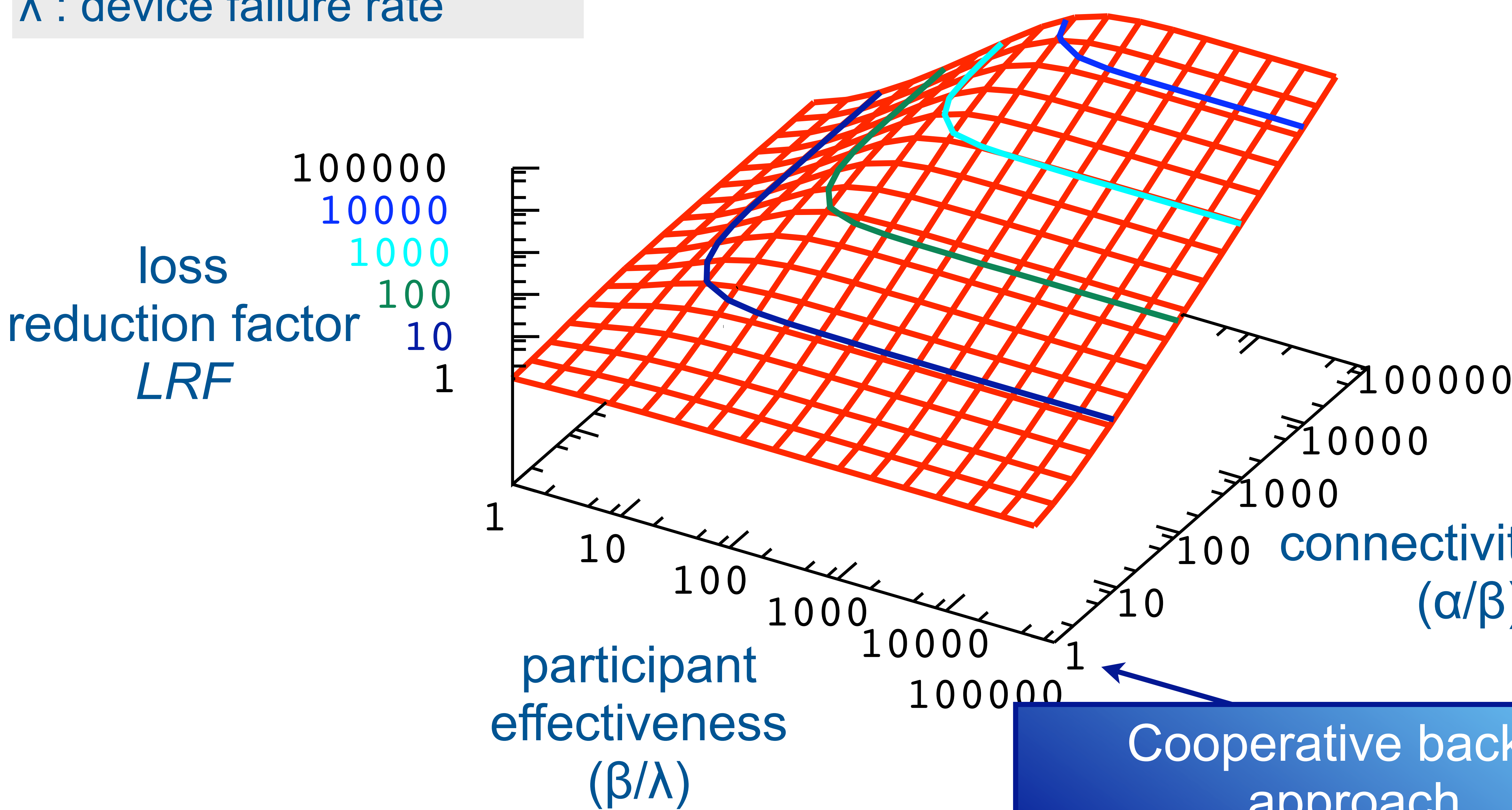
α : device encounter rate
β : internet connection rate
λ : device failure rate



LRF vs. basic parameters

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

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

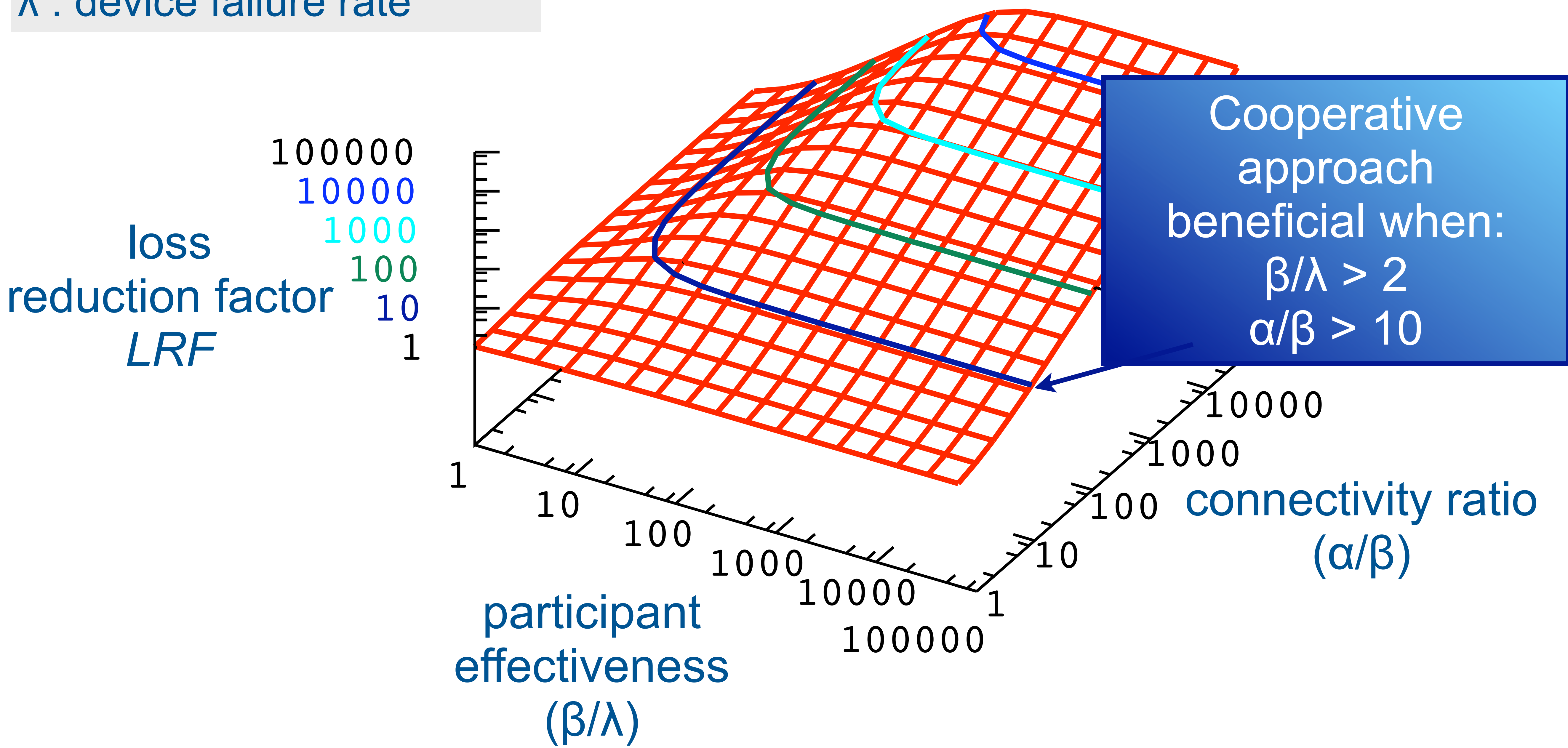


Cooperative backup approach useless when α/β < 1

LRF vs. basic parameters

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

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

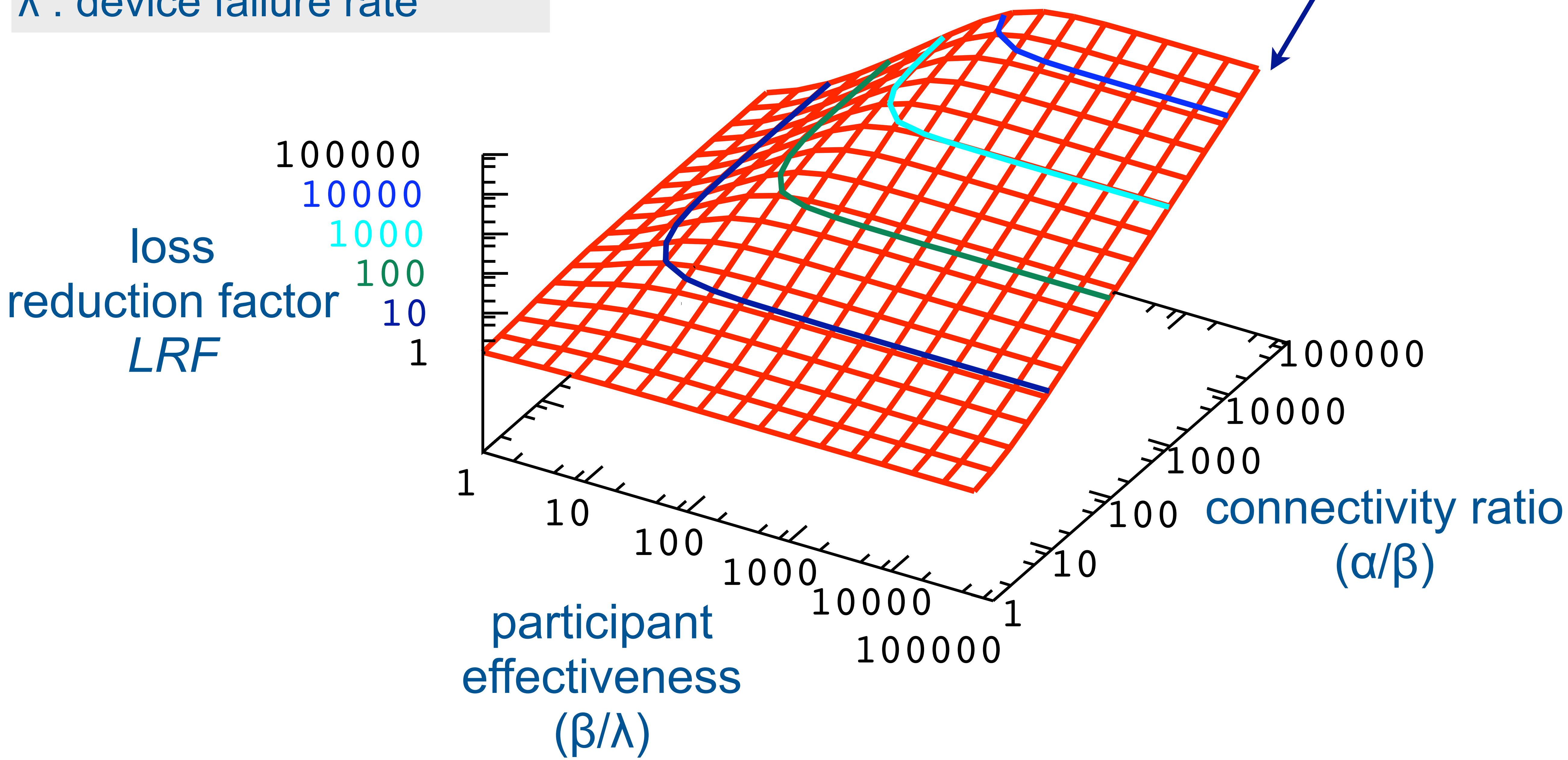


LRF vs. basic parameters

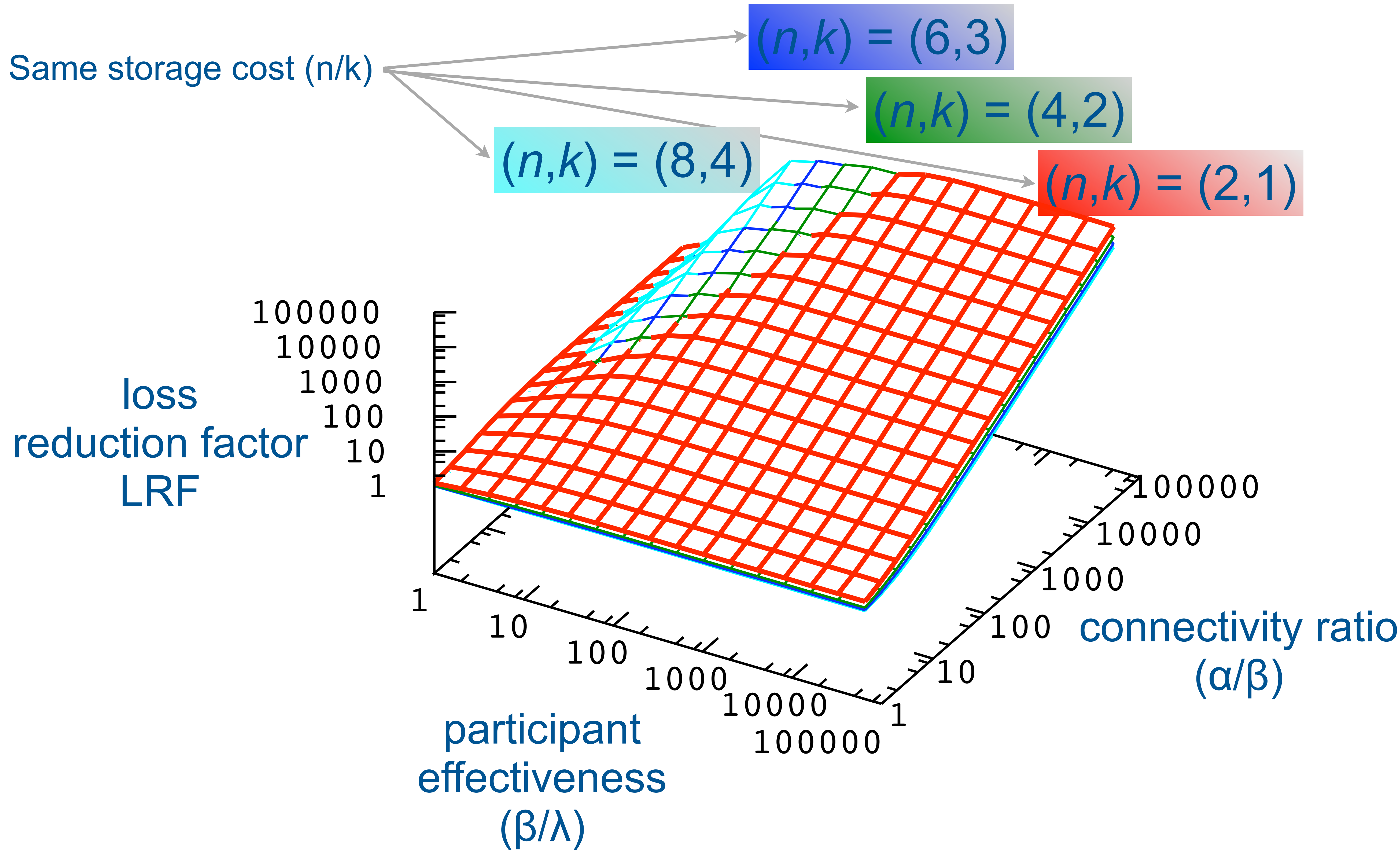
(*n*,*k*) = (3,2)

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

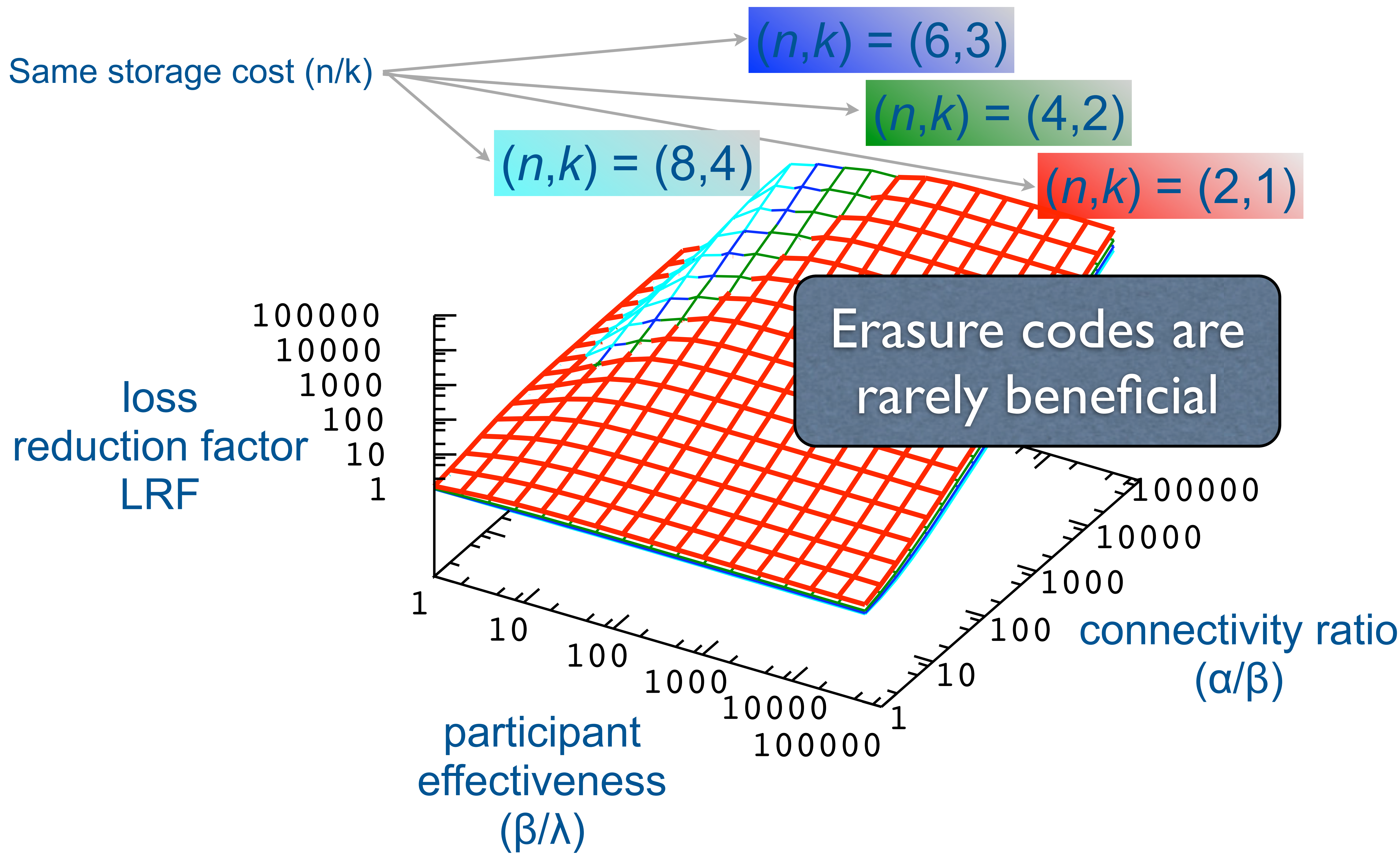
Data loss probability decreased by up to α/β



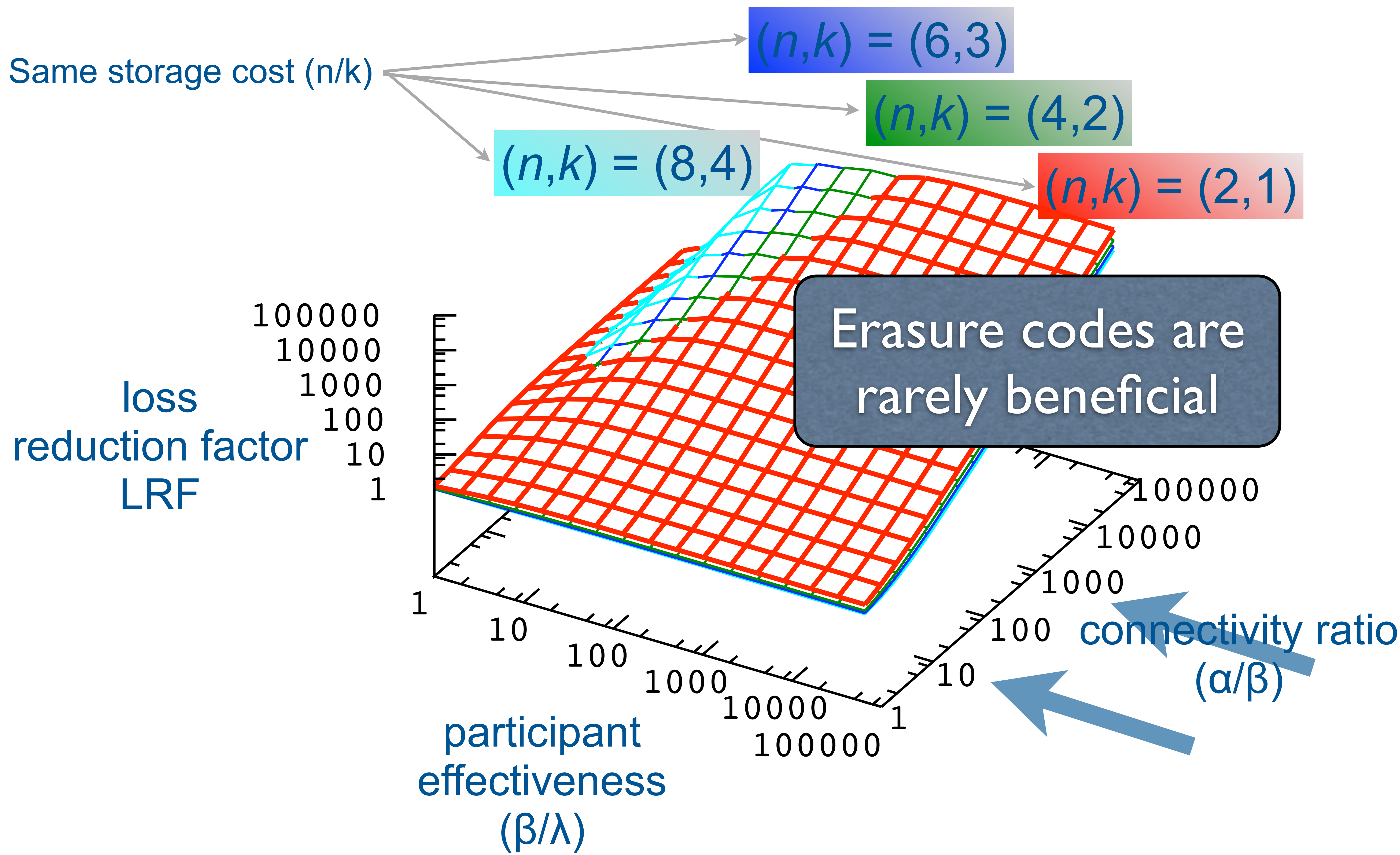
LRF vs. coding parameters



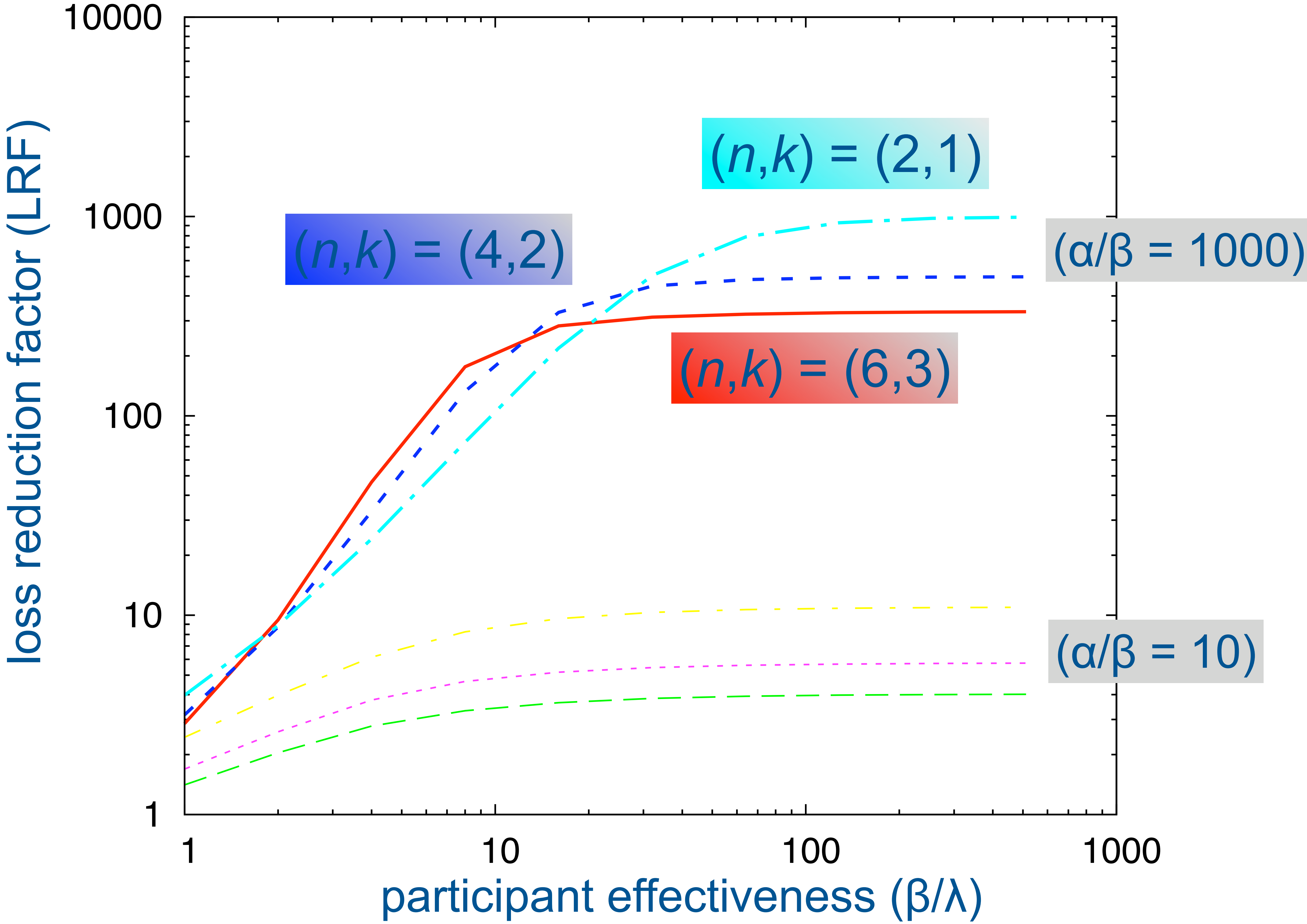
LRF vs. coding parameters



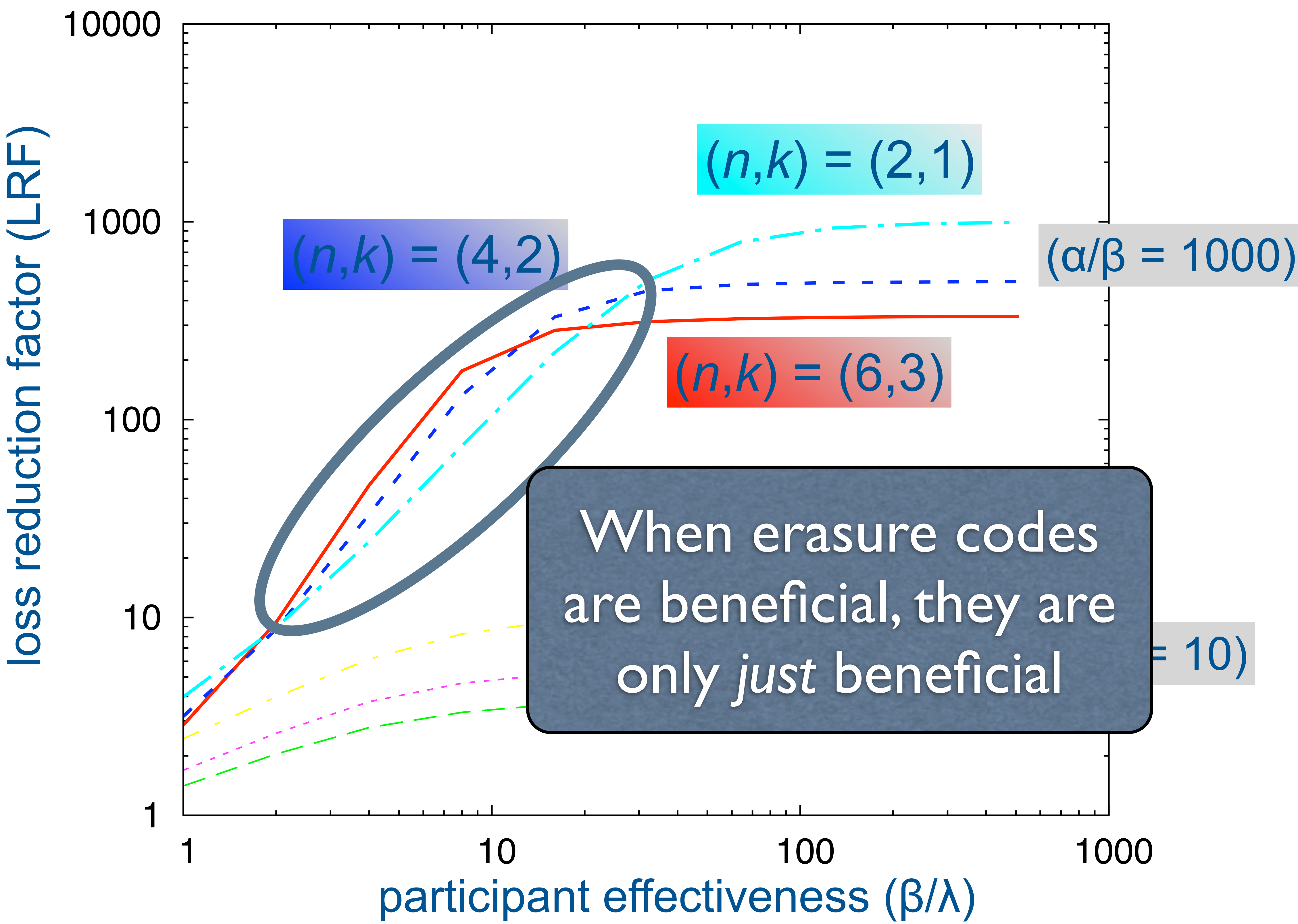
LRF vs. coding parameters



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 $\alpha/\beta>10$ and $\beta/\lambda>2$
- Erasure codes have small advantage over simple replication in only a very narrow domain

Related Work

- **FLASHBACK** [Loo+ 2003]
  - UC Berkeley & Intel Research (USA)
- **UbiStore** [Tan+ 2007]
  - NICTA & Univ. New South Wales (Australia)
- **Swarm-based replication maintenance** [Ball+ 2007]
  - Univ. Kent (GB)
- **Ubiquitous Data Backup** [Aoshima 2007]
  - Hitachi, Ltd. (Japan)
- **Delay- and disruption-tolerant networks** [Fall+ 2003]
  - Intel Research (USA) and others

Future Directions

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

References

MoSAIC

- Killijian+ “Collaborative Backup for Dependable Mobile Applications”, 2nd W/S on Middleware for Pervasive and Ad-Hoc Computing, 2004.
- Courtès+, Storage Tradeoffs in a Collaborative Backup Service for Mobile Devices, EDCC’06
- Courtès+, Security Rationale for a Cooperative Backup Service for Mobile Devices, LADC’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

- Loo+, Peer-to-Peer Backup for Personal Area Networks. Intel, Report, 2003
- Fall, A Delay-Tolerant Network Architecture for Challenged Internets., SIGCOMM'03
- Aoshima, “Ubiquitous data backup”, European Patent Application 1 788 783 A1, 2007
- Ball+, Dependable and Secure Distributed Storage fo Ad Hoc Networks, ADHOC NOW 2007
- Tan+, Ubistore: Ubiquitous and Opportunistic Backup Architecture, PerComW'07