Exploring the CAP Theorem

Lætitia Avrot & Vik Fearing

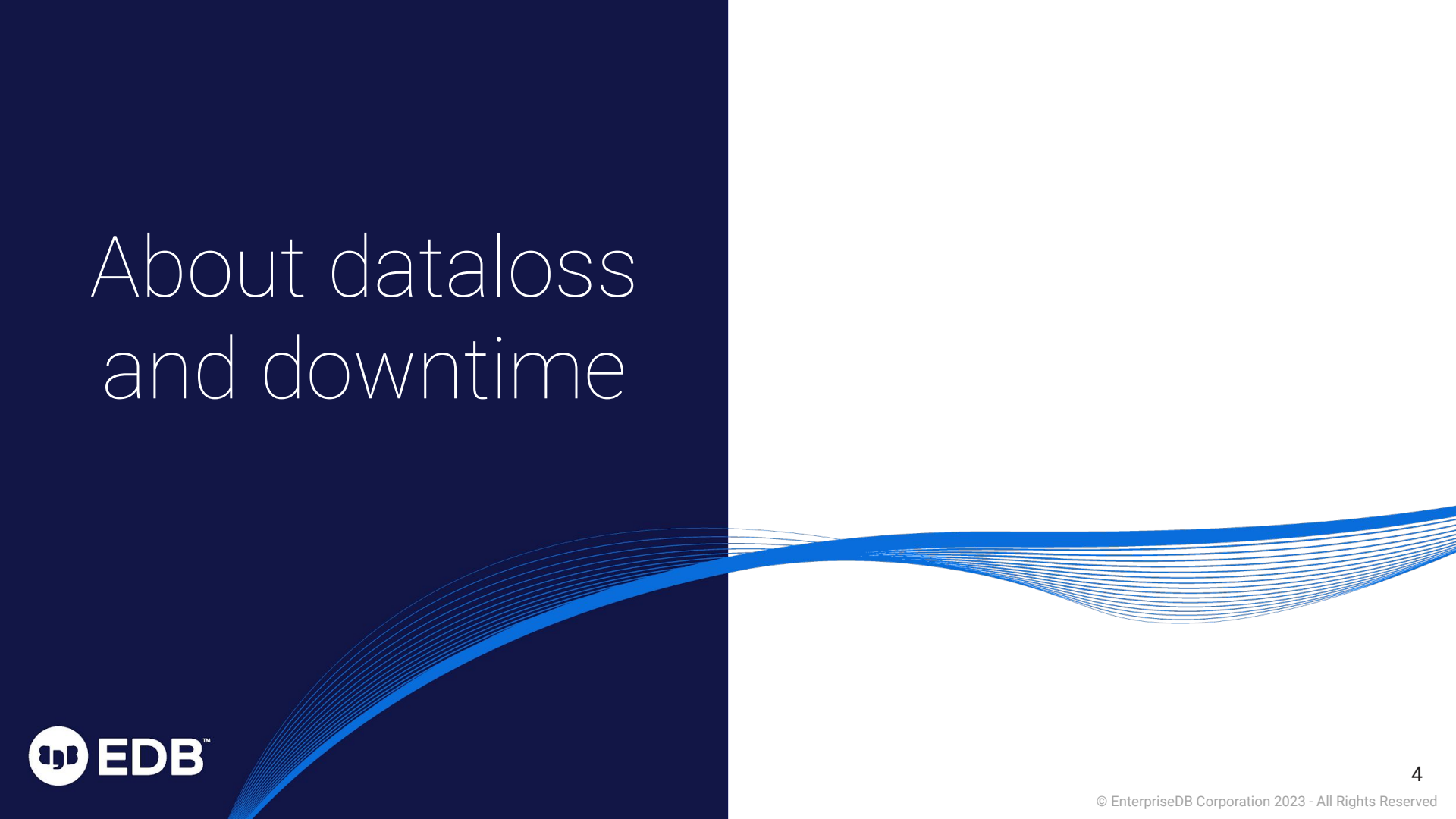
Lætitia Avrot
Field CTO – EDB
PostgreSQL Europe Treasurer
Postgres Women Founder
Recognized PostgreSQL Contributor

Vik Fearing
PostgreSQL Expert – EDB
PostgreSQL Major Contributor
SQL Committee Member
Organizer of several community conferences around the world

Agenda

- About dataloss and downtime
- CAP theorem
- PACELC theorem
- Some typical architectures

About dataloss and downtime



0-Dataloss...

… Does not exist!

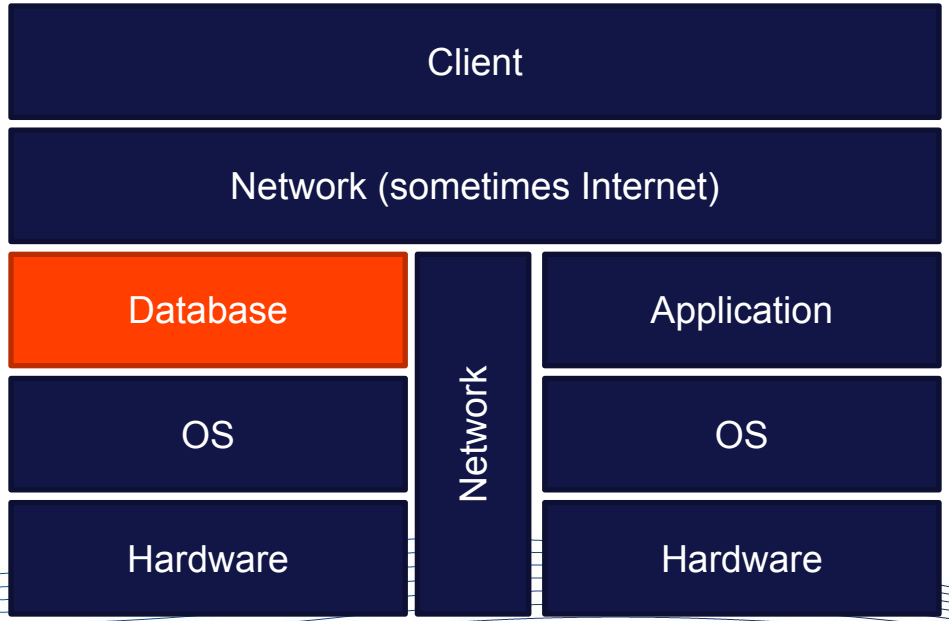
- It’s all about risks and their probability
- 0 risk does not exist
- Near 0 non-acceptable risk does exist

Allowed failure time

Availability needed	daily	weekly	yearly
100%	0	0	0
99.999%	<0.9s	6s	<5min
99.99%	<8s	1min	52min
99.9%	1.5min	10min	8h45min
99.999% during daytime	0.4s	3s	<3min

EDB

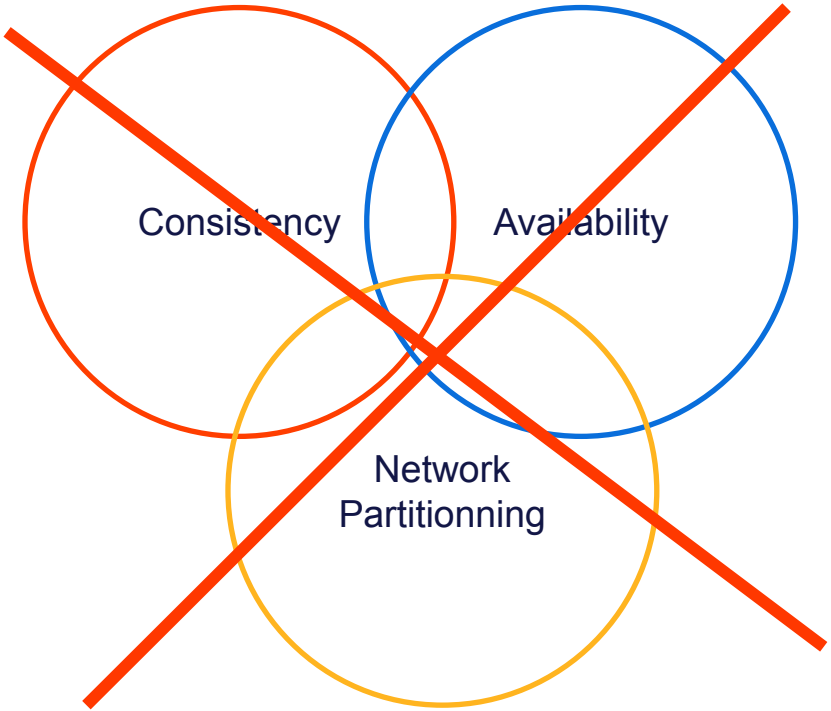
Anythink can break



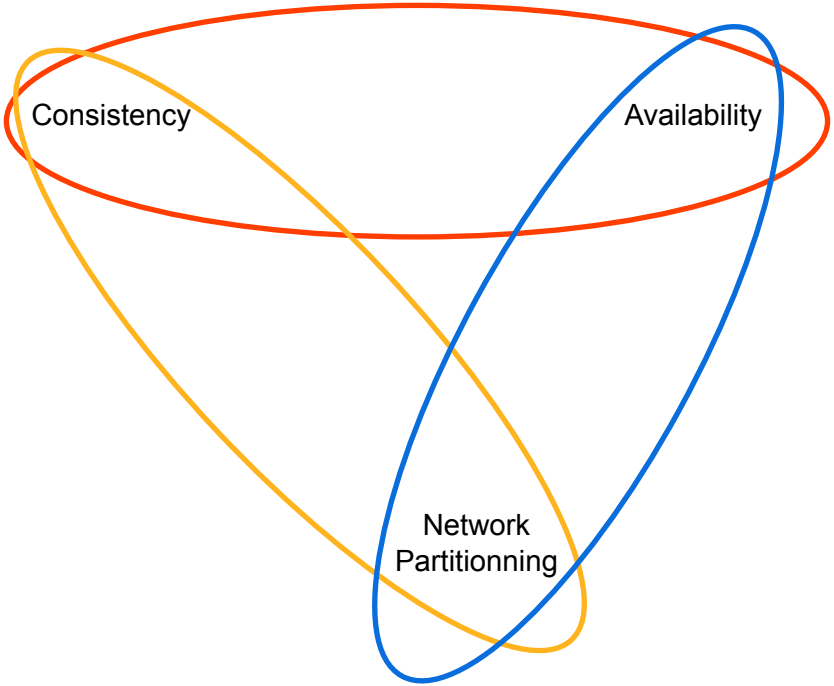
The CAP theorem

When designing distributed web services, there are three properties that are commonly desired: consistency, availability, and partition tolerance. It is impossible to achieve all three.

Graphic representation



Graphic representation



Consistency

Each node will see the last (committed) version of the data.

Availability

The system will always answer something different than an error message.

Network partitioning

The network will be allowed to lose arbitrarily many messages sent from one node to another without altering the system answers.

”Two out of three”

The erroneous interpretation of the CAP theorem.

- Both availability and consistency can be achieved when there is no network partition
- You just need to choose between consistency and availability when there is a network partition

RDBMS are CA

True and False

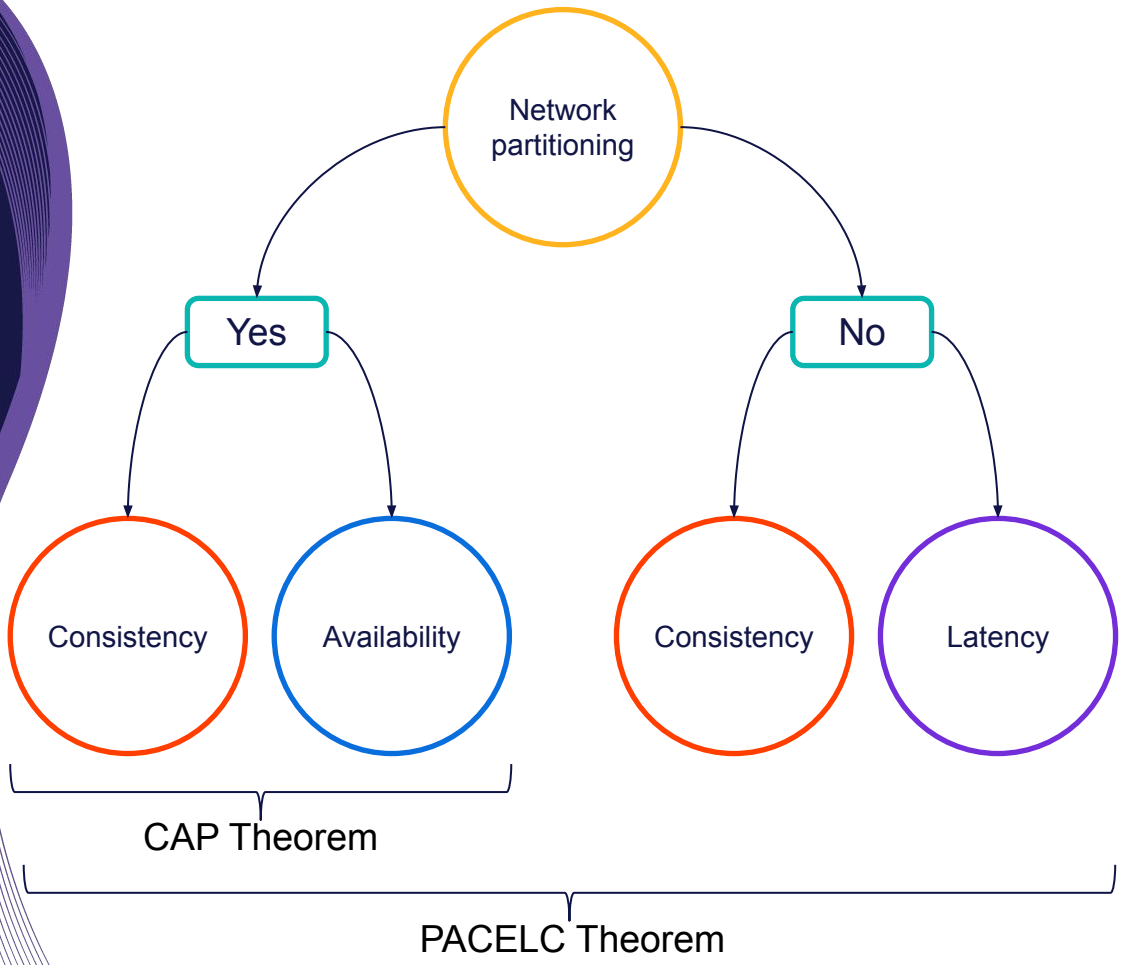
- A standalone RDBMS is CA
- RDBMS with asynchronous replication are AP
- RDBMS with synchronous streaming replication are neither CP, neither AP

The PACELC theorem

In case of network partitioning (P) in a distributed computer system, one has to choose between availability (A) and consistency (C),

but else (E), even when the system is running normally in the absence of partitions, one has to choose between latency (L) and consistency (C).

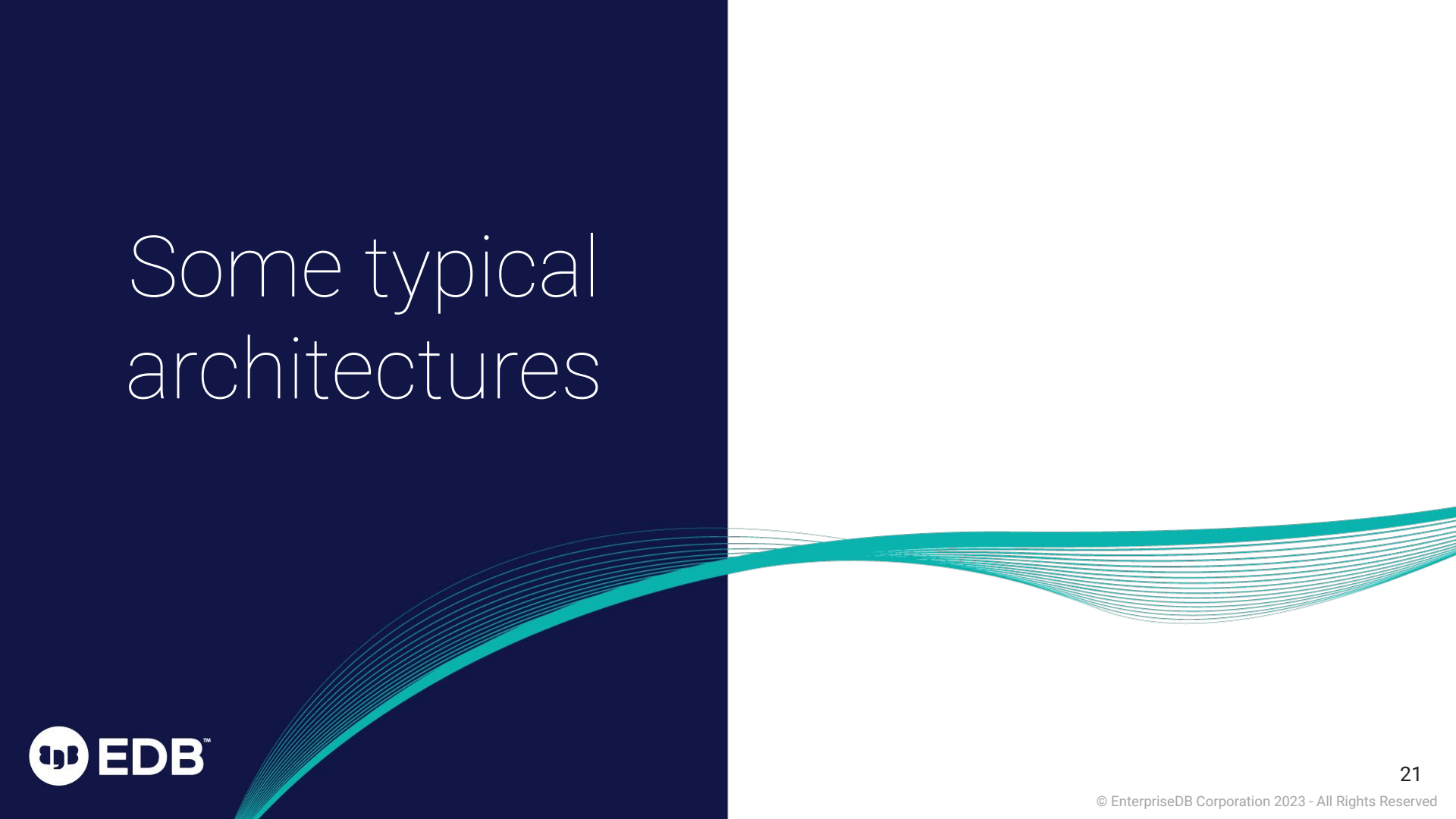
Graphic representation



Latency

Latency is the amount of time the system needs to answer when it's smaller than a request timeout.

Some typical architectures



Dataloss

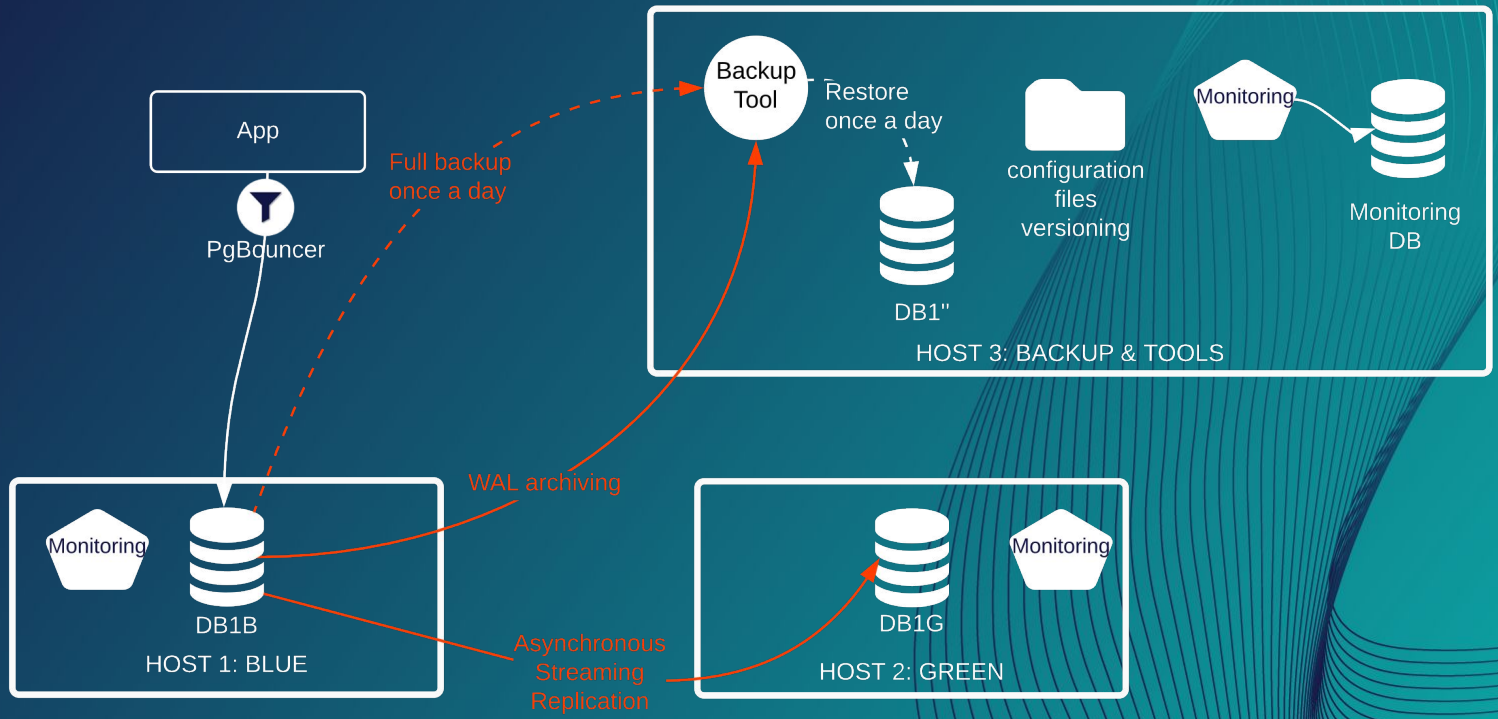
RPO	Architecture
≥ 15min	Physical backup + WALs archiving
≥ 1min	Physical backup + WAL archiving + asynchronous standby
≥ 200ms	Physical backup + WALs archiving+ synchronous standby
0	Not possible 😉

High availability

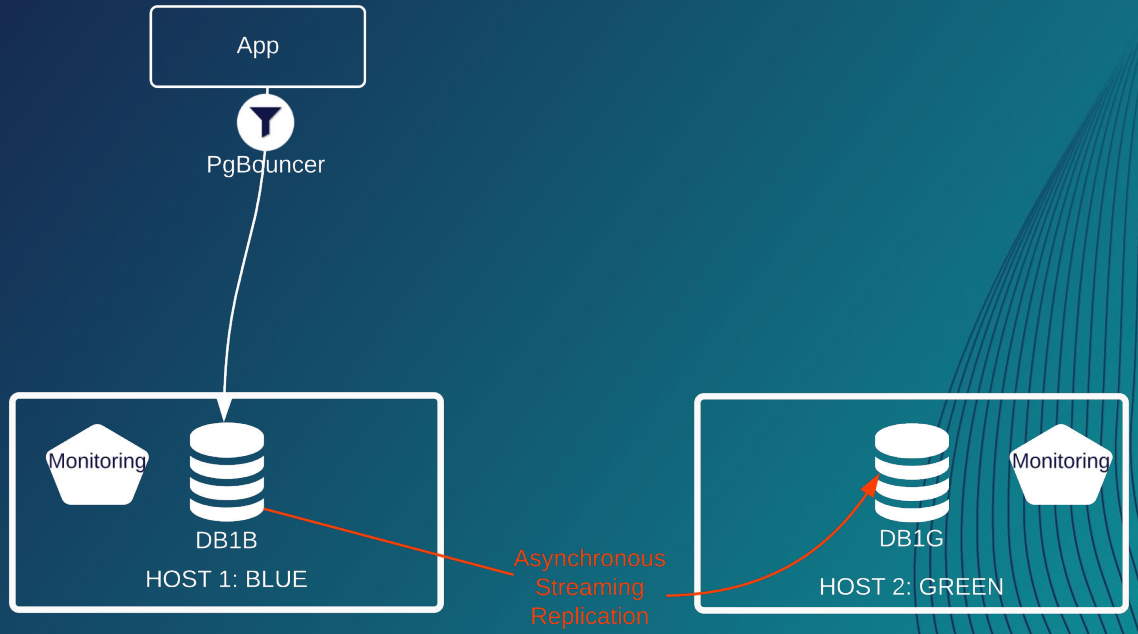
RPO	Architecture
≥ 24h	Physical backup + WAL archiving
≥ 30min	Standby + manual failover
≥ 5min	Standby + automatic failover
≥ 1min/30s	Multi-primary
0	Not possible 😏

EDB

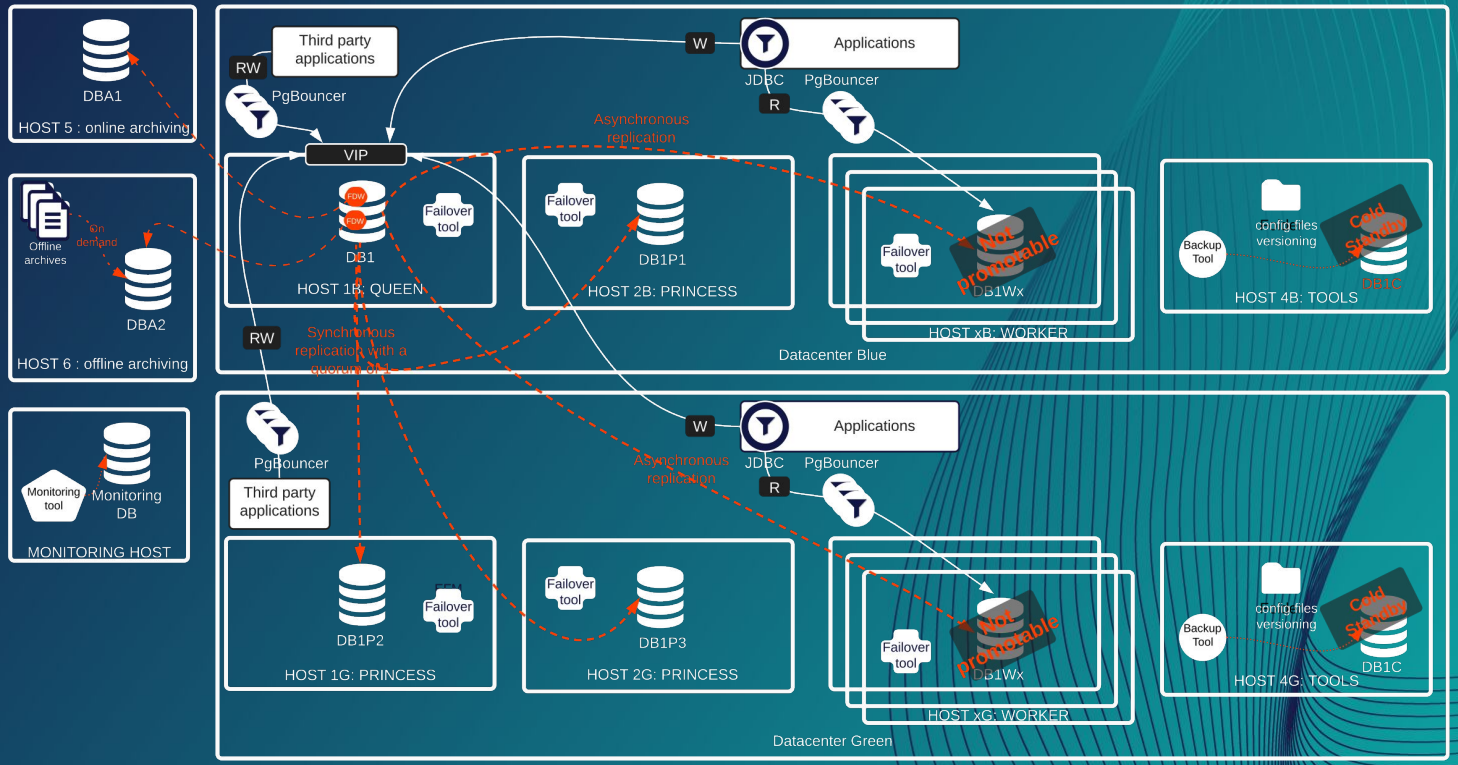
A standard architecture



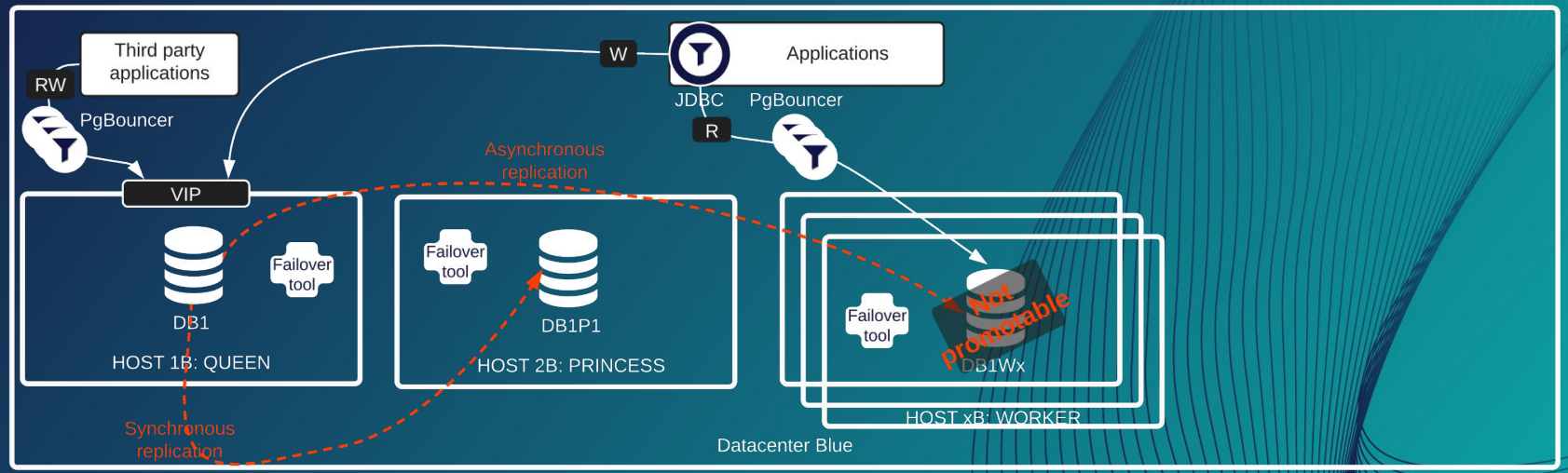
A standard architecture



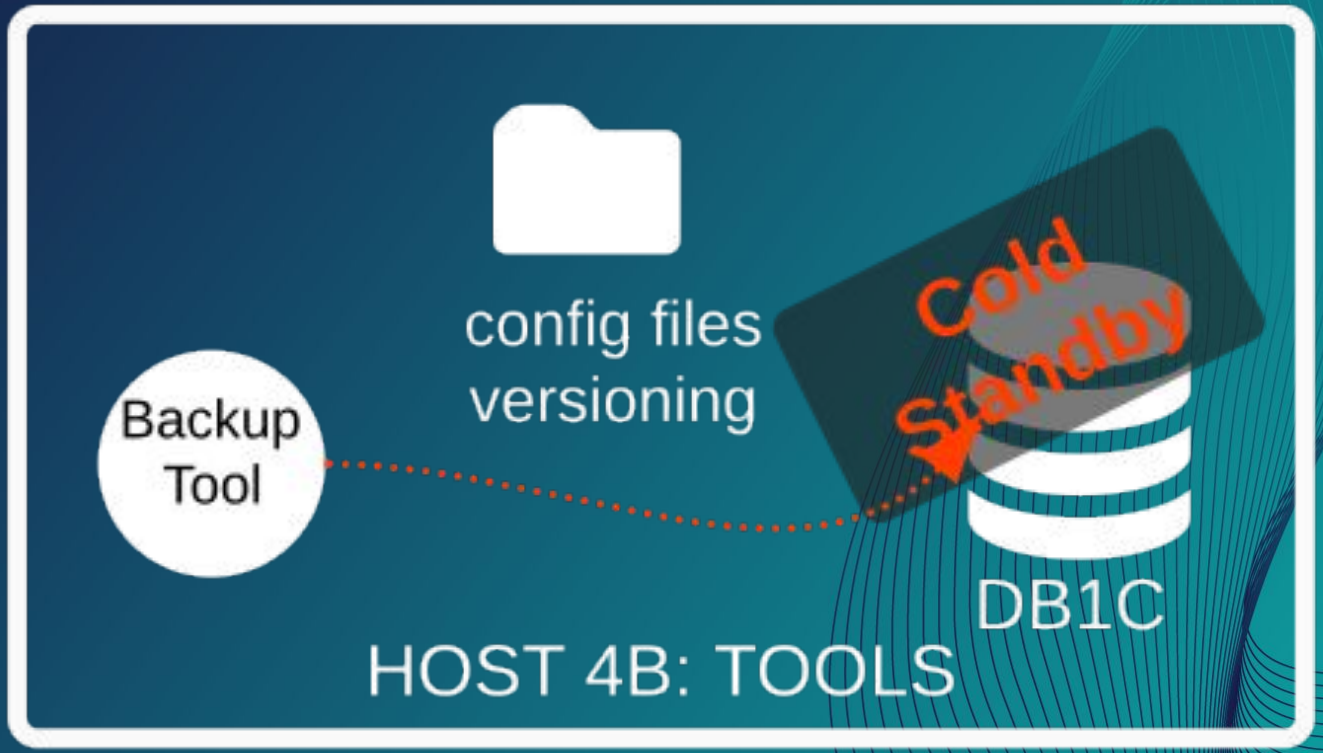
Favoring fast reads



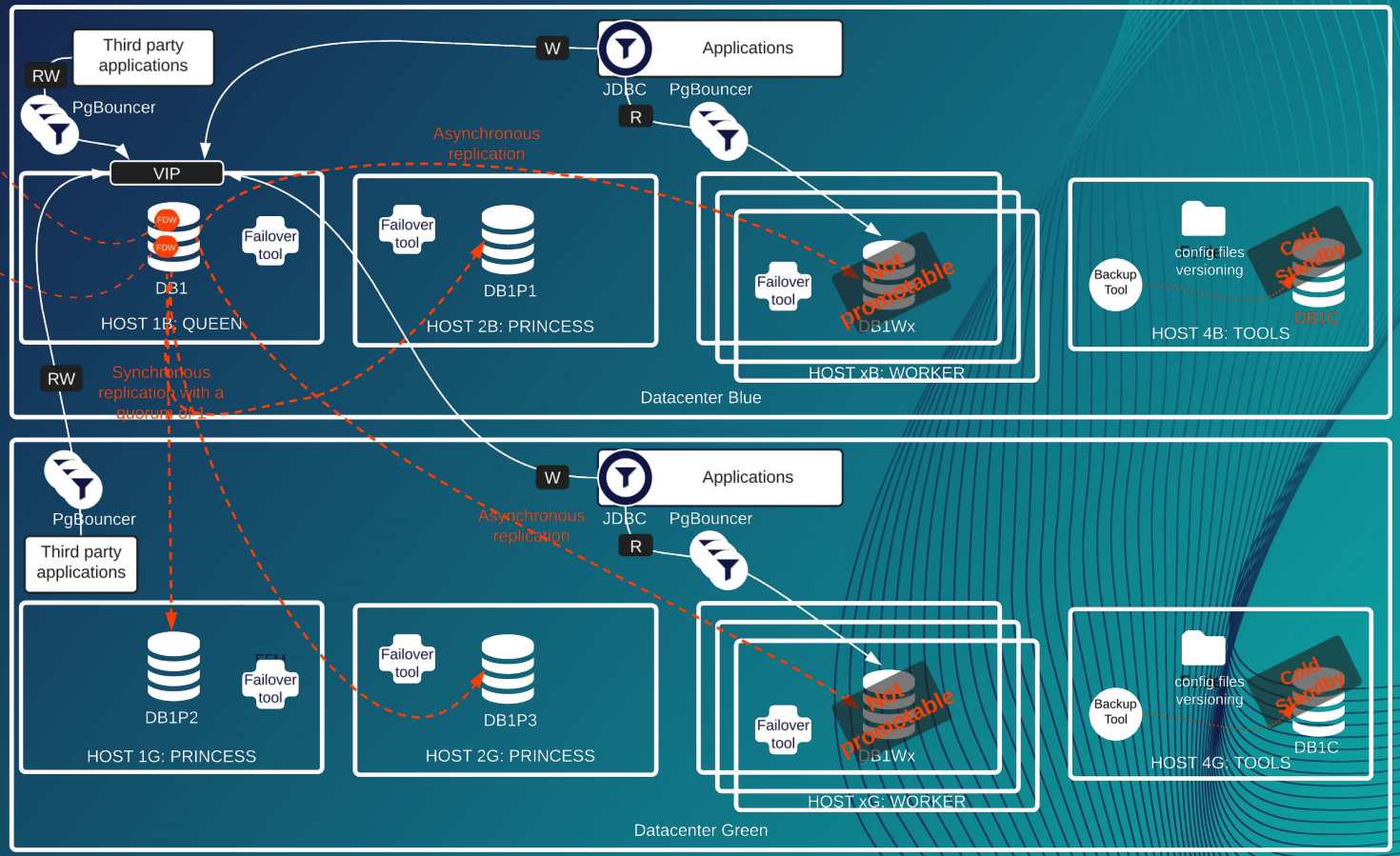
Basic architecture



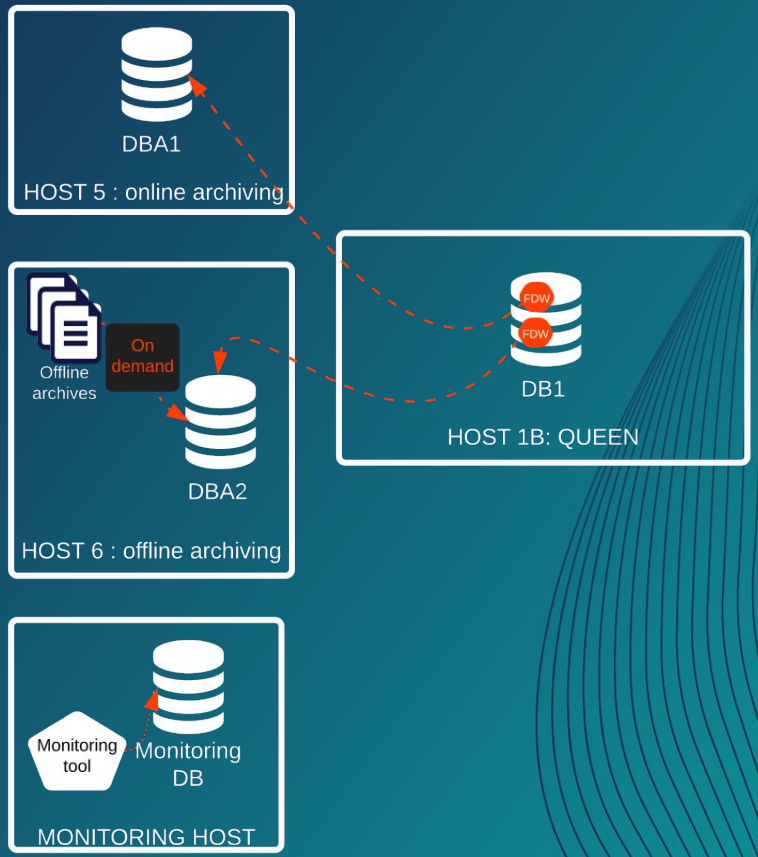
Backups



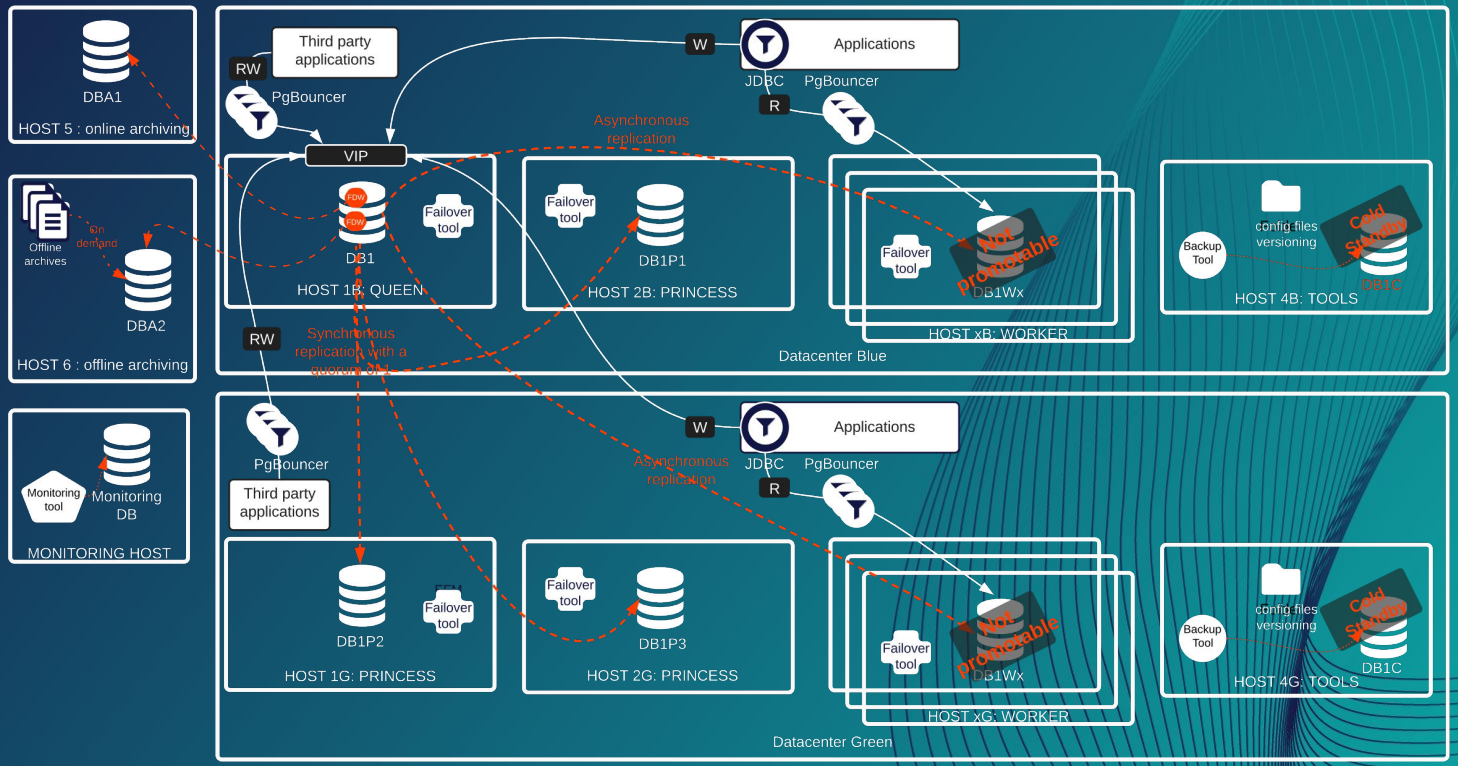
Doubling the datacenters



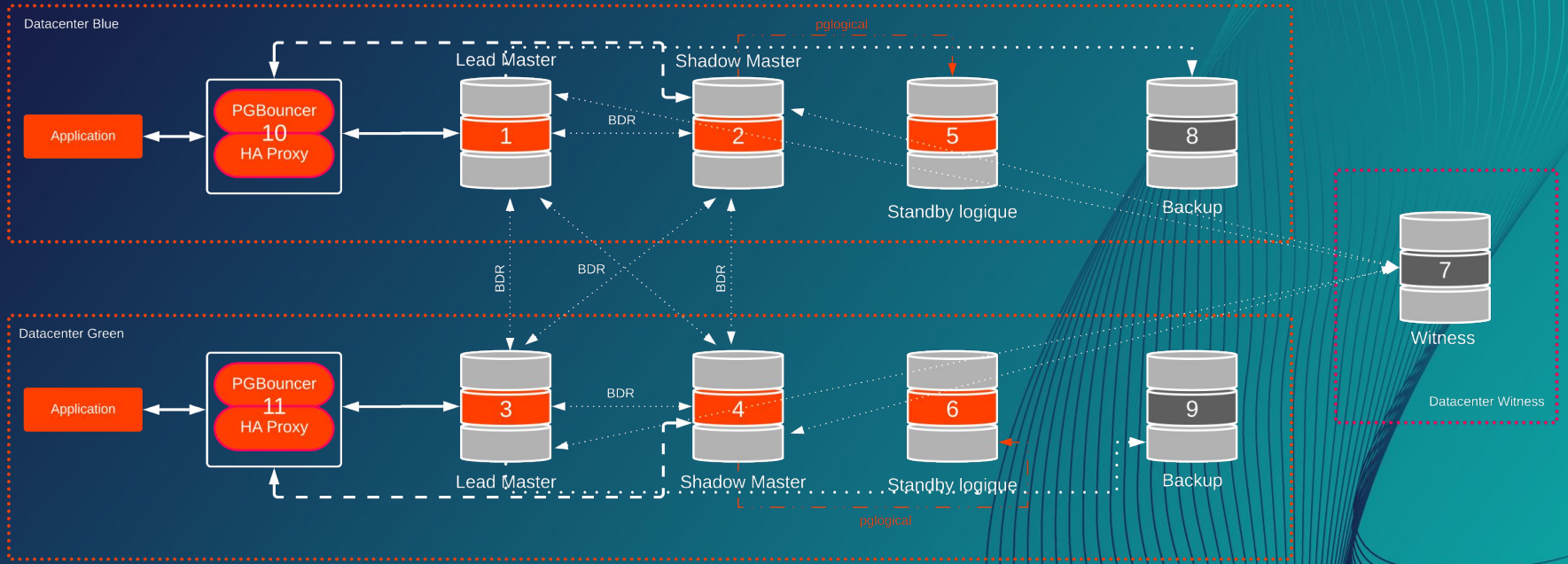
Archiving + monitoring



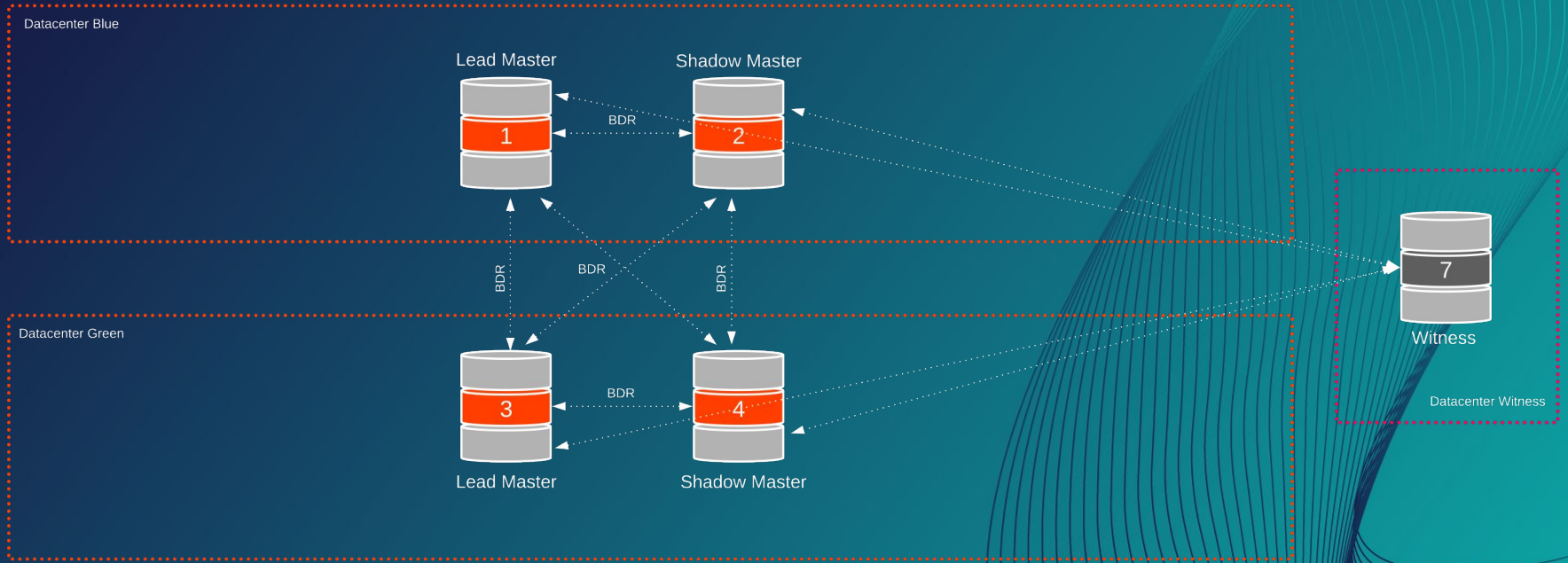
Everything together!



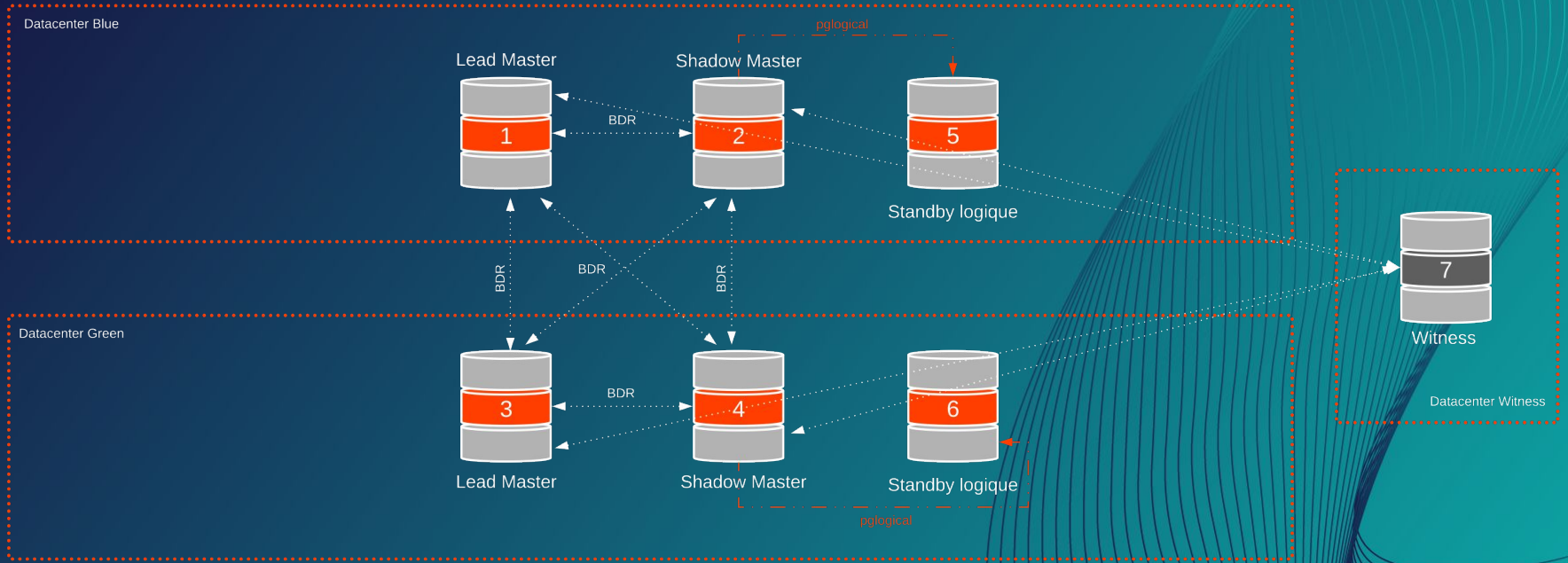
Favour availability



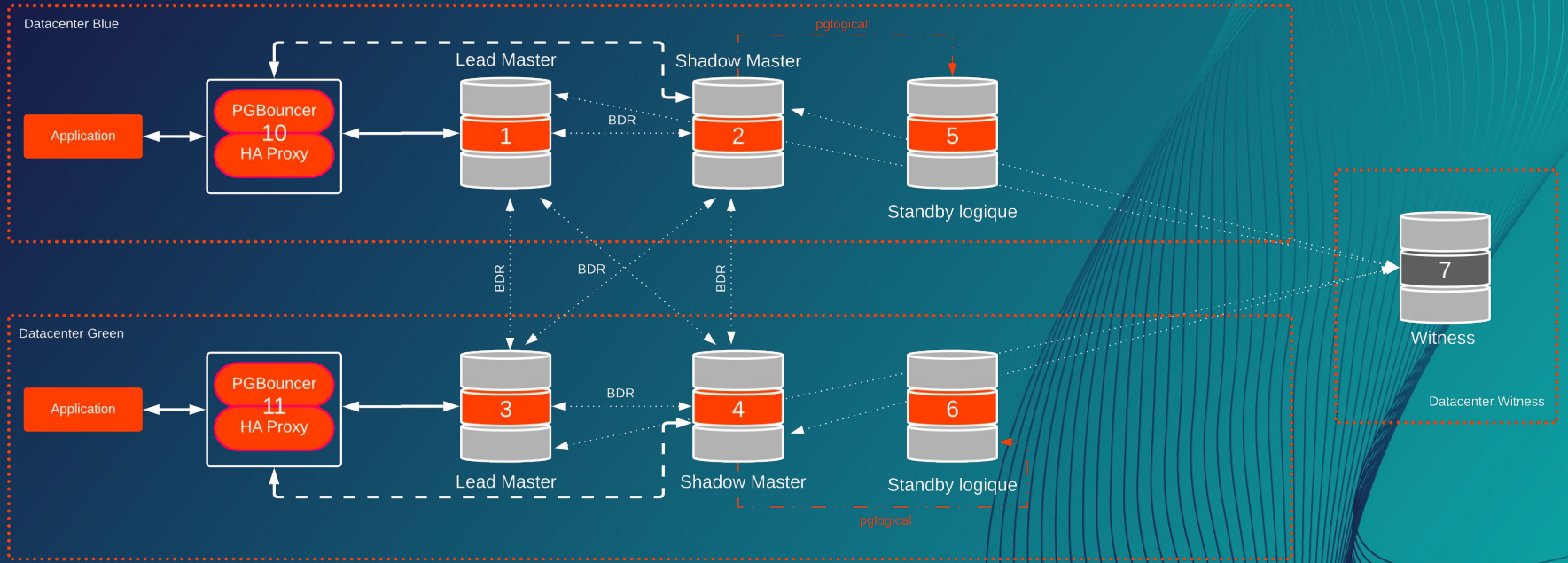
Multi-primary



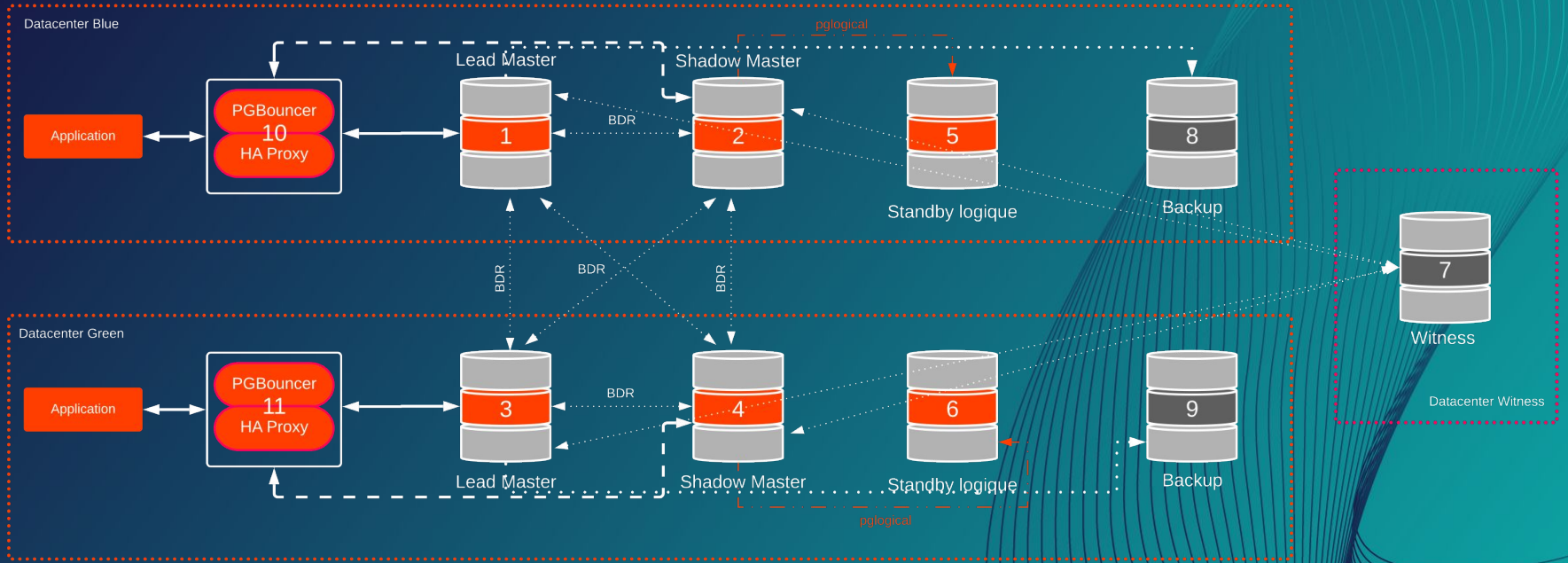
Logical replication



High availability



Adding backups



To conclude

- The database is not the only element that can fail
- Figure out what your RPO is
- Figure out what your RTO is

EDB

EDB™ POSTGRES VISION World Tour

Our flagship user conference is coming to a city near you.

- Dive deeper into Postgres with business and technical sessions
- Learn from EDB and Postgres experts
- Discover fresh insights from industry thought leaders
- Network with Postgres code contributors and power users

JOIN US

NA
DC | June 6
Dallas | June 15
NYC | July 24
Chicago | July 31

EMEA
London | June 6
Dubai | June 14
Amsterdam | June 20

APJ
Seoul | July 11
Jakarta | July 12
Singapore | July 14
Melbourne | July 17
Mumbai | July 21
Tokyo | September 7

Thank you!
Do you
have any
question?