Problems of CAP theorem proof

Petr Škoda

Faculty of Information Technology, Brno University of Technology

LANGUAGE THEORY with APPLICATIONS 2011

14. 12. 2011

Contents

- Introduction
- CAP theorem
- Proof and problems
- PACELC taxonomy
- Enhancing the CAP theorem
- Conclusion

Introduction

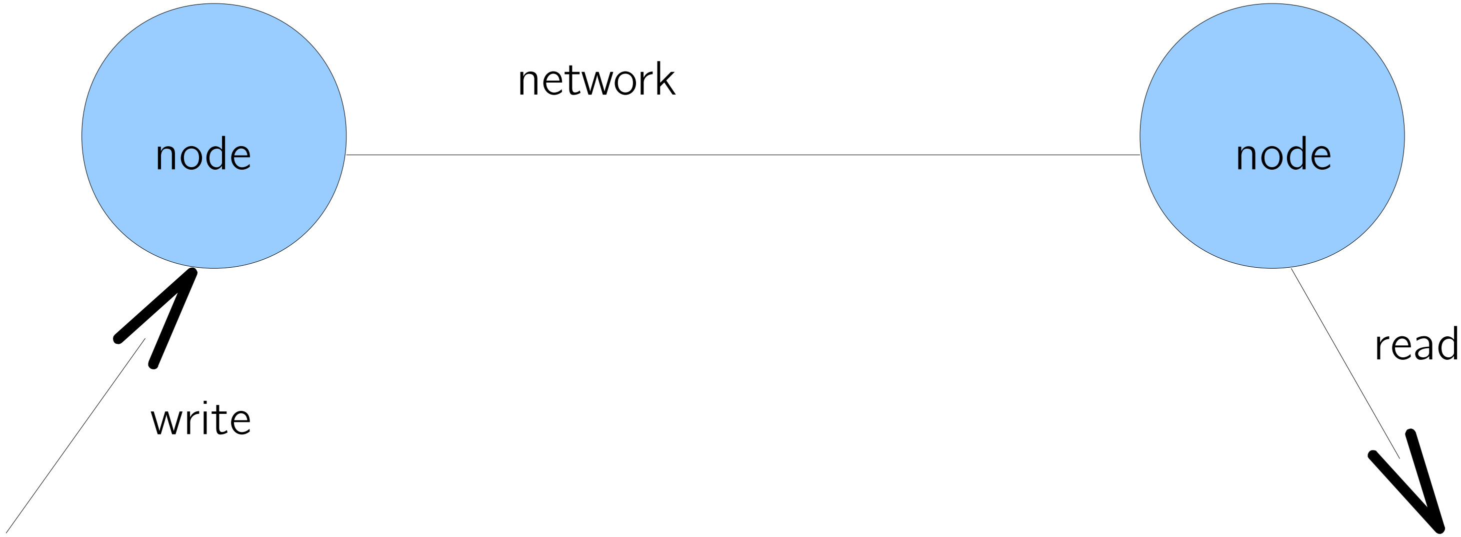
- Relational DBs are “unscalable“
  - Strictly ACID transactions
  - Perfect consistency
  - Best running on one or few machines
- New approach – **noSQL**
  - No transactions
  - Eventual consistency
  - Highly scalable – well run on many machines/in cloud

The CAP theorem

- *“It is impossible for a web service to provide the following three guarantees:* ***C****onsistency,* ***A****vailability and* ***P****artition-Tolerance at the same time.”*
- Originally by Eric Brewer known as “Brewer’s conjecture“
- Usually understood in the following way:
  - You must choose two parameters of the three (CAP)
  - Not chosen parameter cannot be influenced
- This is often criticized
- NoSQL systems are then classified as **CA**, **CP**, **AP**

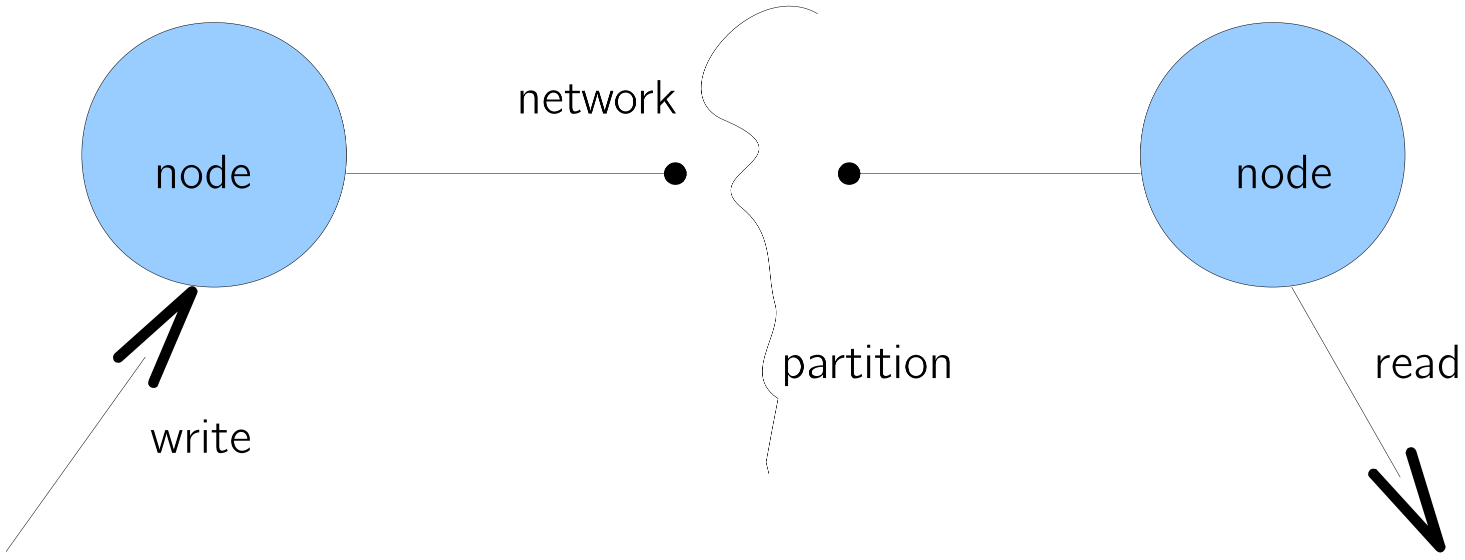
The CAP theorem – proof (1)

- Consider system with only 2 nodes
- Make write on one node, then read from the other
- What would happen while reading?



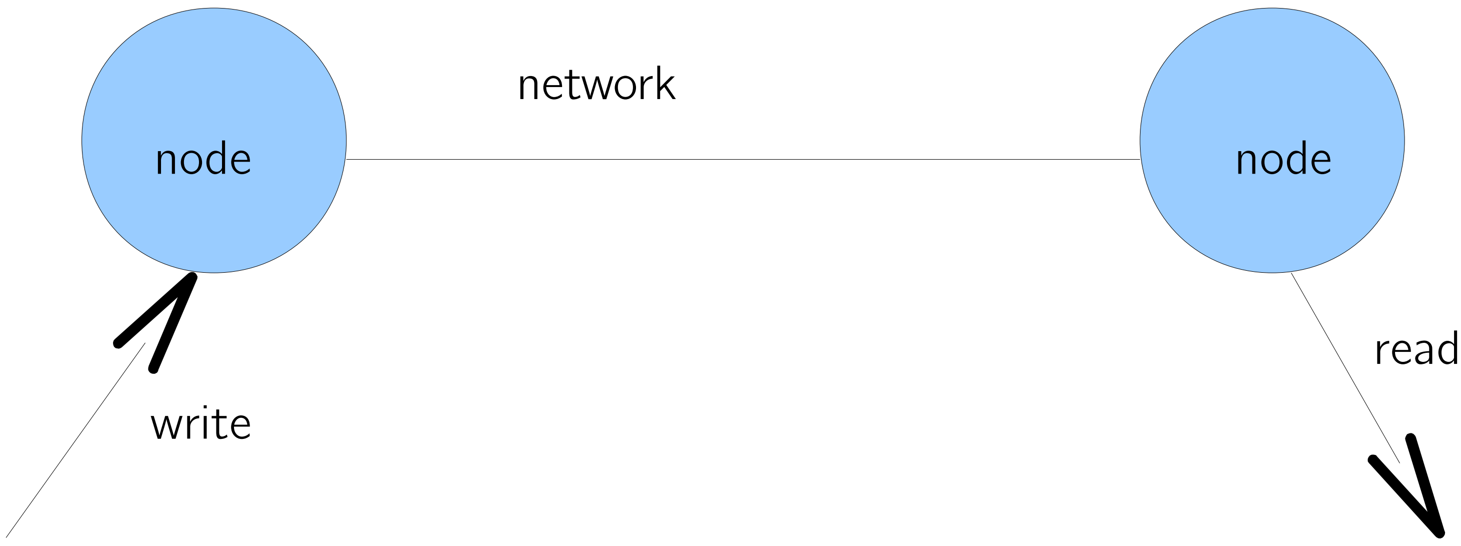
The CAP theorem – proof (2)

- When partition occurs read node could:
  - Return latest known local value –> **not consistent**
  - Wait for latest version –> **not available**



The CAP theorem – proof (3)

- When availability and consistency needed
  - Nodes must communicate – **not partition tolerant**



The CAP theorem – proof (4)

Problems

- Is the system partitioned forever useful anymore?
- Are **CP** and **CA** systems different?
  - When partitioning occurs both systems look unavailable
- What about latency?
  - CAP theorem proof works with no latency at all
- Time may be the key
- The proof is correct but what it proves is **too raw for real world usage**

PACELC taxonomy

- Adding latency “**L**” to CAP
- Classification:
  - PA/EL, PC/EC, PA/EC, PC/EL
  - First part shows behavior in case of **P**artition (**A**vailability or **C**onsistency)
  - Second part shows preferred property when not partitioned (**C**onsistency or **L**atency)
  - PA/EL = When **P**artitioned prefer **A**vailability (over consistency), **E**lse prefer **L**atency (over consistency)
- Looks to be the right direction

Enhancing the CAP theorem $^{(1)}$

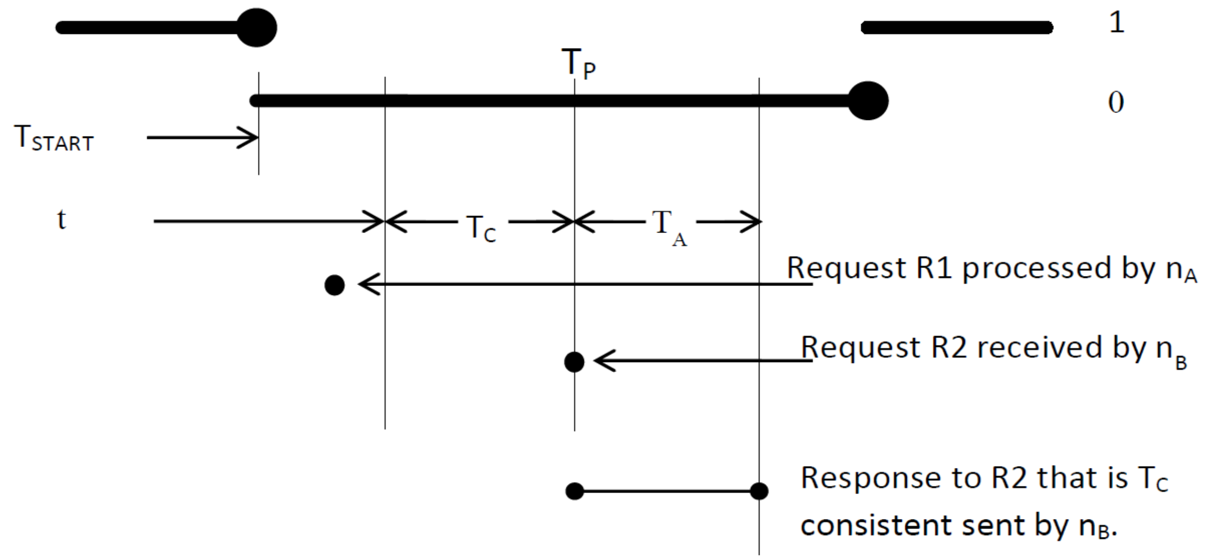
Amrith Cumar, Kenneth Rugg (Oct 2011)

- Precise definitions of *consistency*, *availability* and *partition tolerance* with respect to time
- Think about C, A, P in terms of duration of event
  - Tc – max. time system needs to get consistent after write
  - Ta – max. time between request and response on any node
  - Tp – max. time a group of nodes could be separated
- Conclusion **Tc + Ta ≥ Tp**

Enhancing the CAP theorem [(2)]

Proof

- Consider Tc + Ta < Tp and situation on the picture
- Then we should be able to find a time "*t*" such that:
  - $T_{START} < t < T_{START} + Tp$ **AND**
  - $T_{START} < t + Tc + Ta < T_{START} + Tp$
  - *Which is not possible.*



1
0
T_P
T_{START}
t
T_C
T_A
Request R1 processed by n_A
Request R2 received by n_B
Response to R2 that is T_C consistent sent by n_B.

Conclusion

- CAP theorem is mostly misunderstood
- PACELC is only interesting for classification
- Thinking about CAP with respect to time could show what real systems are able to
- Only practical usage of distributed systems can prove what is really correct
- Some people still argue, that the SQL could be scaled as noSQL is

Questions

Thank you for listening.
Questions?

References

- E. Brewer. “*Towards Robust Distributed Services*”. PODC 2000 Keynote
- S. Gilbert, N. Lynch. “*Brewer’s Conjecture and the Feasibility of Consistent, Available, Partition-Tolerant Web Services*”. ACM SIGACT News. Volume 33 Issue 2 (2002), pg. 51-59
- A. Kumar, K. Rugg, “*Brewer’s Conjecture and a characterization of the limits, and relationships between Consistency, Availability and Partition Tolerance in a distributed service*”. ParElastic: Notes on CAP Theorem October 2011. <http://www.parelastic.com/database-architectures/an-analysis-of-the-cap-theorem/>
- M. Perry. “*The CAP Theorem*”. qedcode.com. [video]. <http://www.youtube.com/watch?v=Jw1iFr4v58M>
- D. Weinreb. “*Improving the PACELC Taxonomy*”. Dan Weinreb’s blog. 12. 1. 2011. [online]. <http://danweinreb.org/blog/improving-the-pacelc-taxonomy>