
CERN

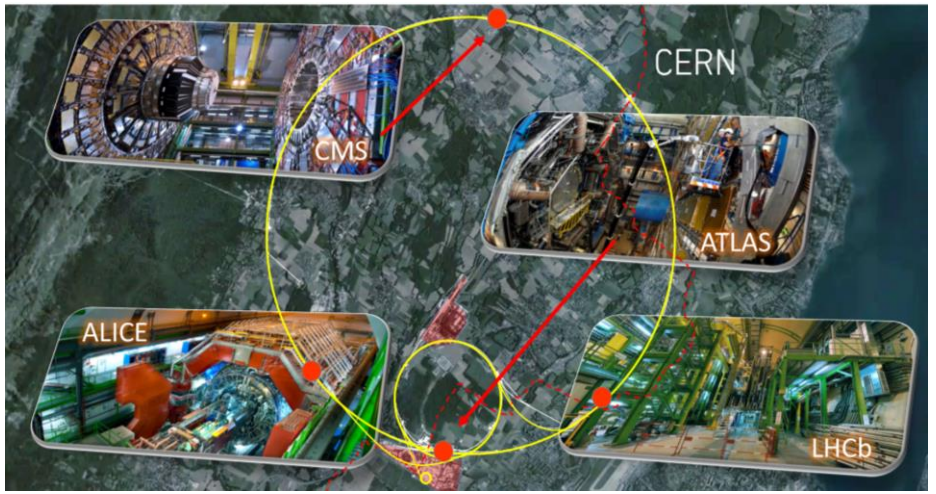
Preserving Large Quantities of Data and Maintaining Digital Sovereignty

Alberto Pace (alberto.pace@cern.ch)

CERN, Geneva, Switzerland

How Large is “Large Quantities of Data” ?

- Today, some sciences are dealing with hundreds of petabytes, heading for Exabyte scale in 1, max 2 years
  - Weather forecast, Earth observations, biology, medicine, astronomy, high energy physics, …
- Computing is the main strategy for many research fields
  - Investments in computing have the highest return

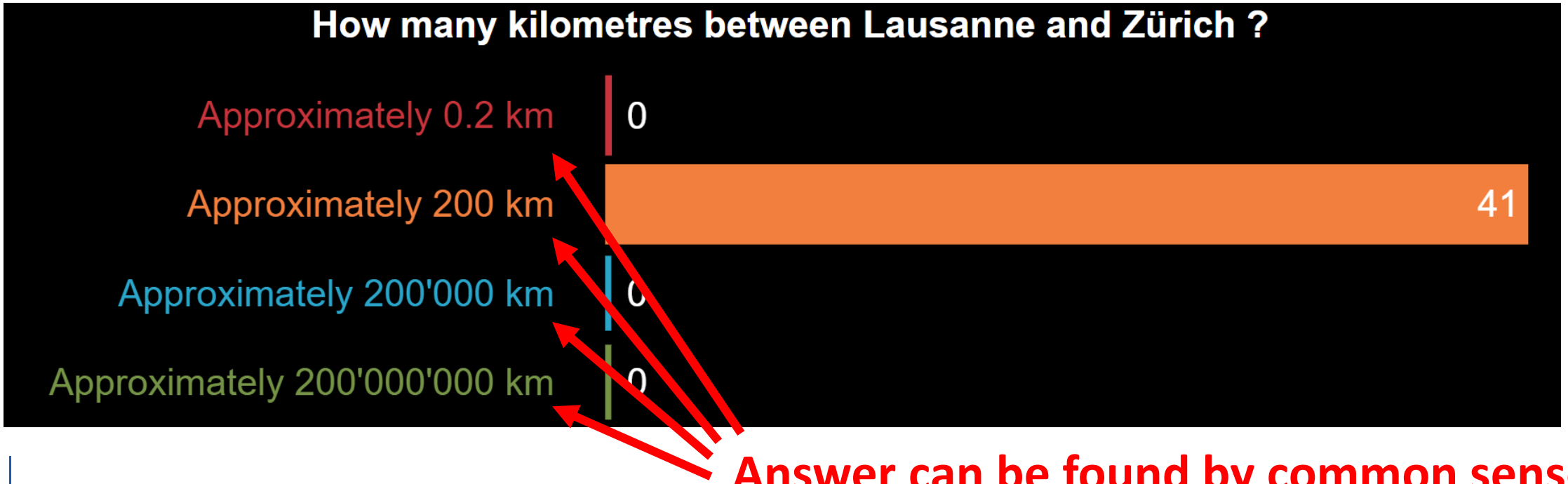


We are just at the beginning

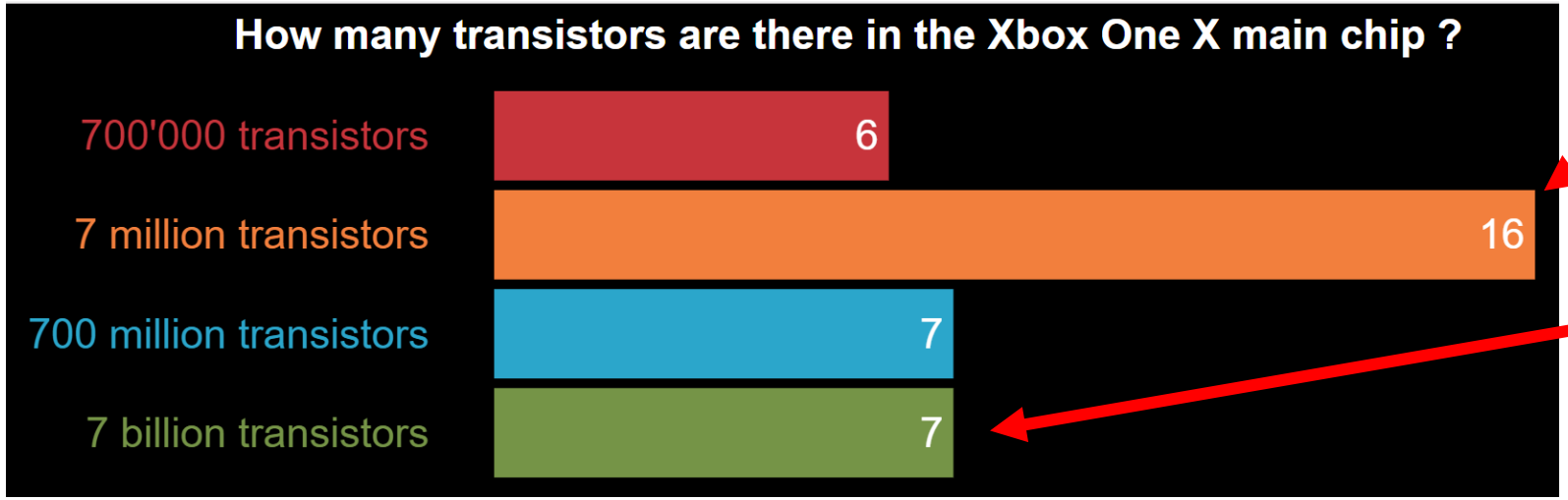
- Statistics and Mathematics are sciences, they do not change …
- But computers have never improved so fast:
  - The explosion of data processing possibilities
    - CPU performance (10^3 increase) and number of CPUs available (10^3 increase)
  - New storage possibilities
    - From few GB to many PB (10^6 increase) – Big Data
  - The possibility of collect / transfer / store these data in a distributed environment
    - From few Mbit/s to 100s of Gbit/s (10^5 increase)

The general population is not aware of these changes

- An example of a survey made with Master students from University of Lausanne:

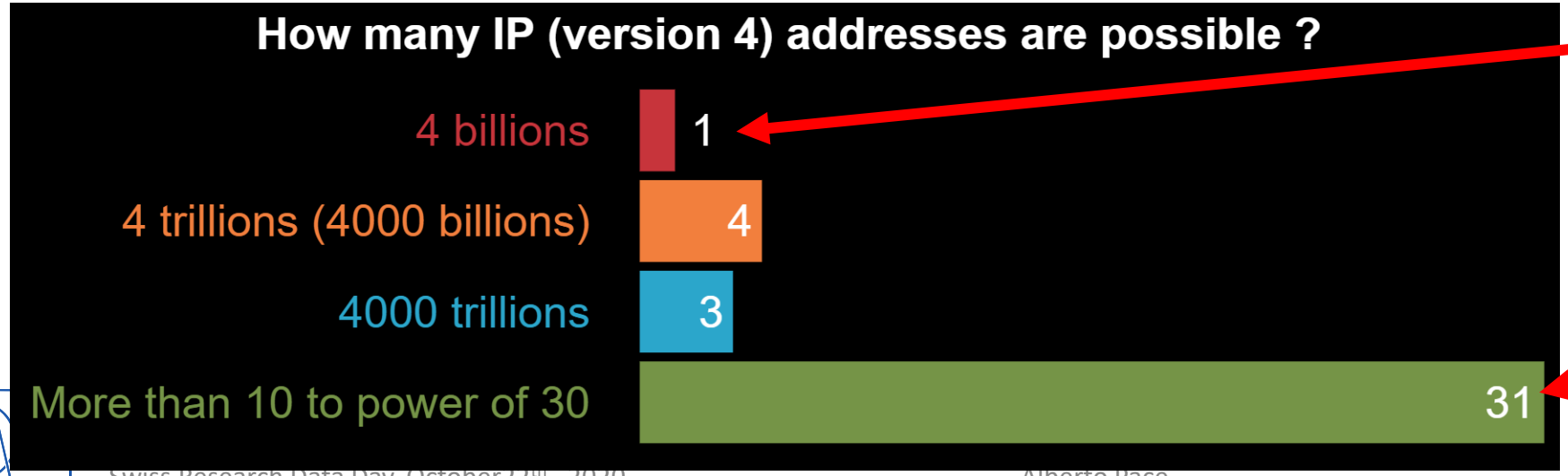


but … what about questions on computing ?



most popular answer
has 3 orders of magnitude error

less than 1/5
correct answers !



only 1 answer correct

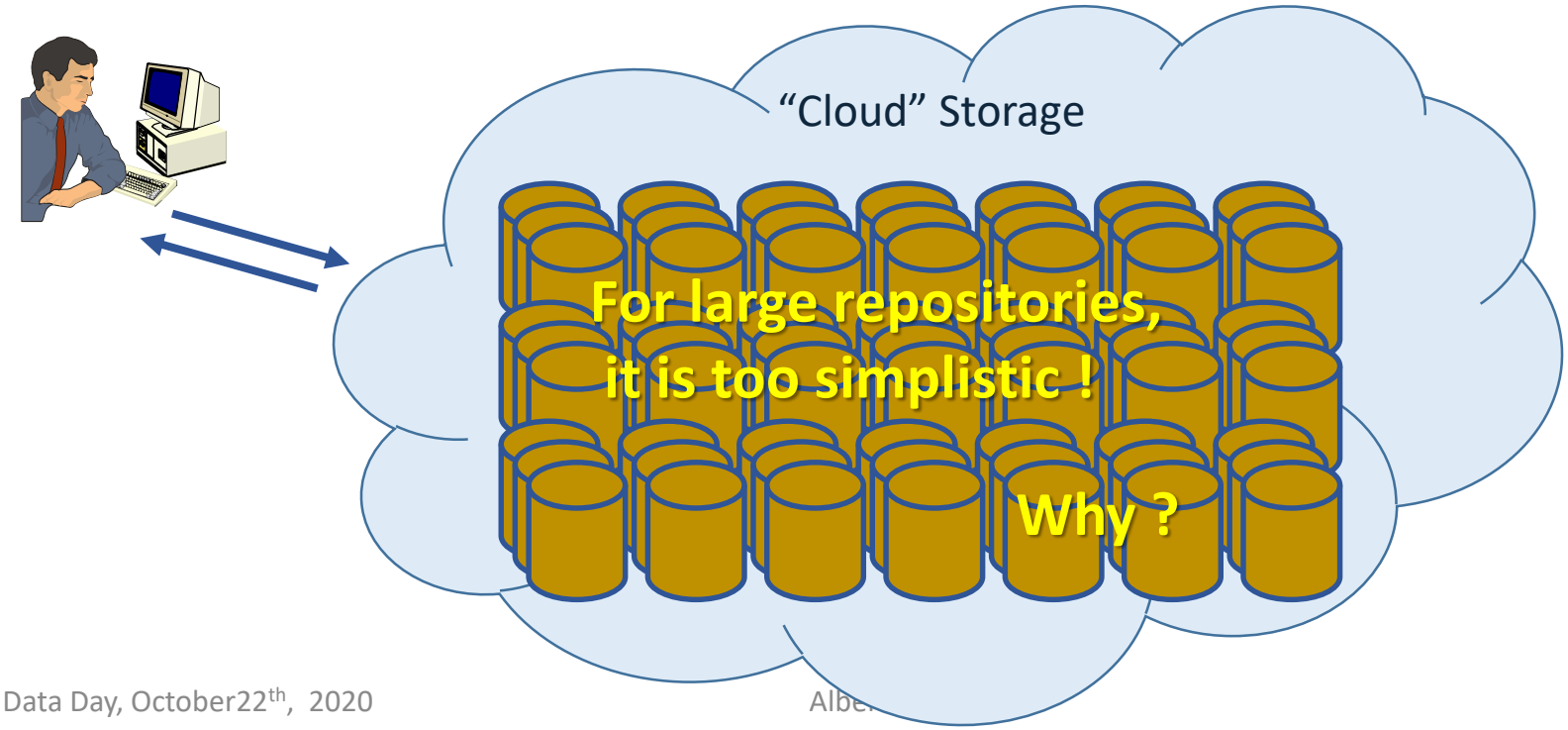
20 orders of
magnitude error

“Why” data management ?

- Data Management solves the following problems
  - Data reliability
  - Access control
  - Data distribution
  - Data archives, history, long term preservation

Can we make it simple ?

- A simple storage model: all data into the same container
  - Uniform, simple, easy to manage, no need to move data
  - Can provide sufficient level of performance and reliability



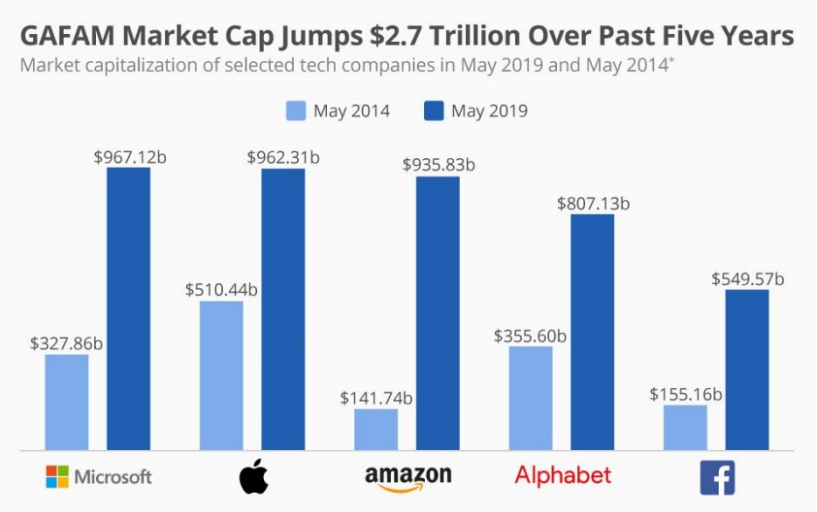
Why multiple pools and quality ?

- Raw data that need to be analyzed
  - Need high performance, High reliability, can be expensive (small sizes)
- Raw data that has been analyzed and archived
  - Must be low cost (huge volumes), High reliability (must be preserved), performance not necessary
- Derived data used for analysis and accessed by thousands of nodes
  - Need high performance, Low cost, minimal reliability (derived data can be recalculated)

History repeats itself !

- Same story already heard:
  - 1990's Software: "is so flexible that can be done in the last minute …"
  - 2010's Data: "Why care about data? Just put It in the cloud (for free)"
- This introduces the need for data sovereignty
  - Vendor lock-in has a huge impact: access to your data is at stake
    - Companies fails
    - Contract fails
    - Law changes
    - Subject to remote jurisdictions
    - Loss of your own data, not just access to software
  - Cannot be bought !

Data is more and more strategic



- Can this be done in the cloud ?
  - There are multiple (big) vendors competing
  - There are established standard interfaces and protocol that ensure interoperability
- Ok for small amounts of data … but:
  - All major vendors offer free data ingestion, but expensive data retrieval. This breaks the model of interoperability and, in fine, data ownership.
  - All major vendors are under US jurisdiction, and this is a political risk

*: source www.cnbc.com, www.statista.com

When funded with public money …

- The public sector is particularly risk-exposed as contracts are granted on the lowest bid …
  - Interoperability is required in the initial contract, but as the product evolves, it can be deliberately removed
- If interoperability is not guaranteed, it is easy to win public contracts by bidding at very low prices (big organization are even offered free services)
  - Vendors' expectation is to achieve a lock-in, and increase prices at contract renewal.
- Costs of change can be huge, and are not provisioned in contracts.
  - They represent a debt
  - Debt removes the freedom of choice, it removes sovereignty

Digital Sovereignty

- Is related to identifying which (digital) activities can be outsourced while maintaining the authority to self-govern
- Is two-folds, as it applies to
  - Data
  - Software

Today, things have changed for storage

- Software in storage
  - In many scenario software can be the most strategic component
  - When you are 'big', software should have a fixed-cost only
- Storage Hardware
  - If the "software" problem is correctly handled, the Hardware + Energy is where variable-costs are concentrated – scale out is possible at minimal ‘marginal’ cost.
- With this approach …
  - the cost of adding a PB of storage is limited to the cost of a PB of HW
  - the cost of operating an additional PB of storage is limited to the cost of the required energy and hardware amortisation
  - If you do not have the critical mass …. cooperation is the solution

Data at CERN

- CERN has the critical mass to build its own data centre
  - planning to reach exabyte scale in 2 years
- For storage, CERN has its own software stack (CERNBOX, EOS, FTS, …) built on top of established and interoperable open source solutions (Linux, Virtualization, Containers, Block Storage, Sync and Share, …)
  - The stack ensures Storage, Backup/Archival, Data Transfers, Sync and Share
- This approach requires important investments in terms of infrastructure
  - but the marginal cost of storage is limited to the media and energy cost, which is one order of magnitude lower than cloud storage

Software is strategic

- It took many years to acknowledge the importance of software
  - Software is flexible and that's where you invest !
  - The more you invest, the more complex it becomes, and that's where you need top level skills
- Three choices:
  - Write your own software, closed or open source
    - This gives an economic advantage as competitors may not have access to your software. You pay all the cost.
  - Use open source, customize the software to your needs
    - You pay only the customization cost. Competitors have access to your software, therefore economic advantage must come from better products, skills, support, services, ... i.e. your business
  - Using proprietary, licensed software
    - Costs are easy to estimate and customization can be also purchased. Competitive advantage comes while your license remains cheap
- Using proprietary software is outsourcing
  - You outsource something that is standard and not important to your business, in order to reduce cost

When is it effective to outsource ?

- 3 requirements (apply to all fields, not just computing):
  - The activity is not strategic / not core business
  - The activity has clear established standard (interfaces / protocols)
  - There are multiple independent vendors implementing these standards.
- If any of these 3 requirements is not satisfied, you are exposed to problems
  - Lock-in, business or service failure, blackmailing

Licensed software cost explodes

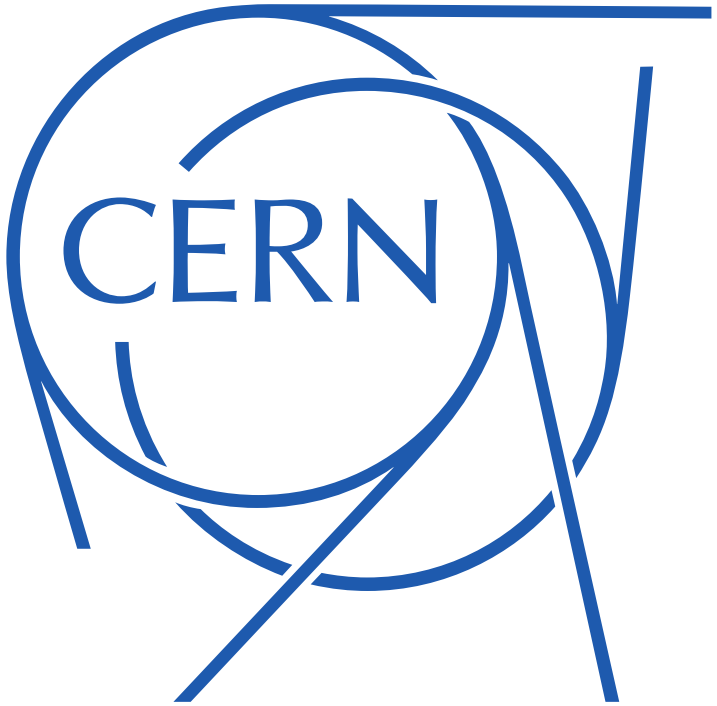
- Every vendor plans on 10 % revenue increase / year
- This is incompatible with a fixed budget organization
- Licenses cost constantly on the rise, with all vendors
  - Majors: Microsoft, Oracle, …
  - Specialized: Financial, CAD, Engineering, Support workflows
- CERN has started in 2019 a project (MALT) to reduce significantly dependencies on licensed software
  - It will take years and important investments …

Findings so far …

- For Storage, the open source approach is working very well since several years
- The CERNBOX Sync & Share portal to access storage is becoming also an access point to corporate recommended applications in the MALT project
- The critical mass to build a successful infrastructure is beyond what a small institute can afford:
  - The **consortium** / open source approach is the best practice to collaborate on well focused projects that guarantee maintaining ownership of your critical activities at a minimum cost

Conclusions

- The more critical a component is to your business, the more marketing pressure you will receive to outsource it (i.e. use proprietary sw)
- Outsource standard activities, well defined and interoperable
  - Do not outsource your own business
- Insource what is specific to you, and your critical activities
- The open source approach is the best practice to insource your critical activities at a minimum cost
  - This will guarantee fixed cost for software.
  - No license cost proportional to data volumes (or number of nodes, or cores, or disk, or data transferred).


CERN