CS 856 Web Data Management

PicoDBMS: Scaling down database techniques for the smartcard

Philippe Pucheral, Luc Bouganim, Patrick Valduriez, Christophe Bobineau

Discussion Presented by:
Hesham Fahmy

Initial Thoughts

- Paper well written
- Motivation for solution developed well with good background.
- Solution well presented and ideas are relatively new (while not in base idea but in innovative application to solution)

Why DBMS on Smart Card?

- “Supporting database management on the card itself rather than on an external device is the only way to achieve very high security, high availability (anywhere, anytime, on any terminal),and acceptable performance.” --- Is this necessarily true?
- “The Smart Card is the unique trusted part of the system, access rights and transaction management cannot be delegated to an untrusted terminal.” ---This is a design restriction that can be overcome.

Why DBMS on Smart Card?

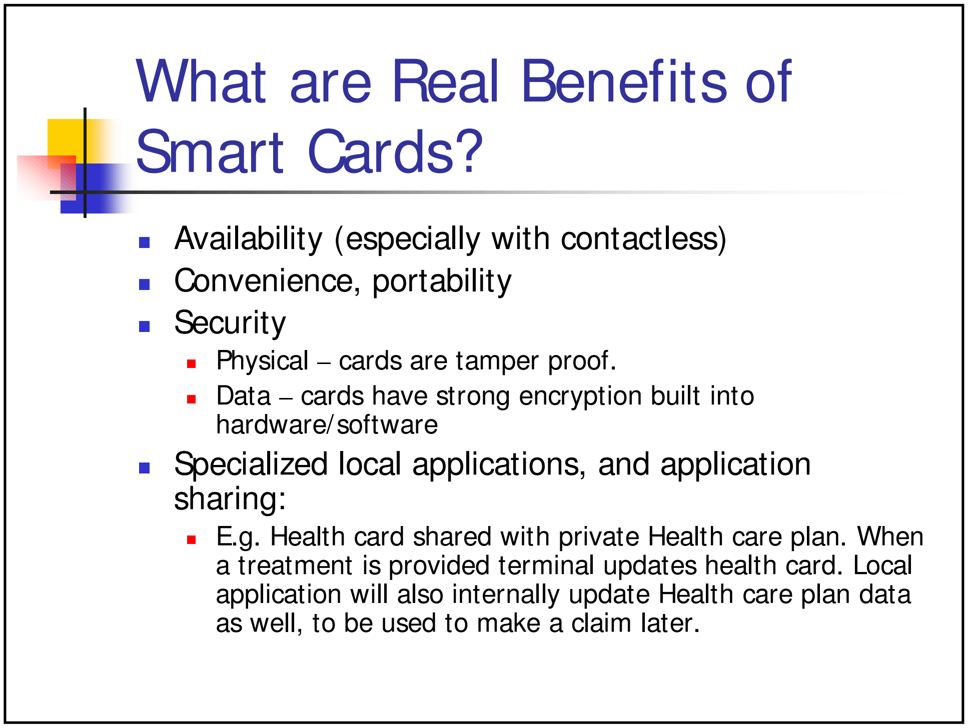
- Availability on any terminal handled by interoperability standards (ISO, JavaCard, etc).
- Having a system with data only on the Smart Card is not feasible either:
  - What happens when card is damaged or lost? Information needs to be backed up somewhere.
  - Data is not always relevant only to the card holder. E.g. Health system; gov’t needs to manage and administer the system and cannot do so if info is only on smart cards. Need to have some centralized view of the overall data to do this.
  - Hence, real world applications will most likely rely on replication/synchronization between distributed versions of data and the smart card. As such the Smart Card is not the sole keeper of trust.

Why not Trusted Terminals?

- Eventually data will be presented to the terminal anyways. Some trust must exist (access rights in PicoDBMS) so why not enhance it?.
- A different design paradigm can better solve this problem by introducing the notion of trusted terminals (not a radically new idea):
  - Newer versions of smart cards use external memory modules, and security is still maintained outside the scope of the physical smart card.
  - Paper already argues that for transaction durability a “trustee server” must be used to store coordinator log. (note that the paper also argues that durability only effective when terminal reconnects to the network. Another argument for a network copy of DB)
  - Paper also recommends delegating result sorting to terminal.

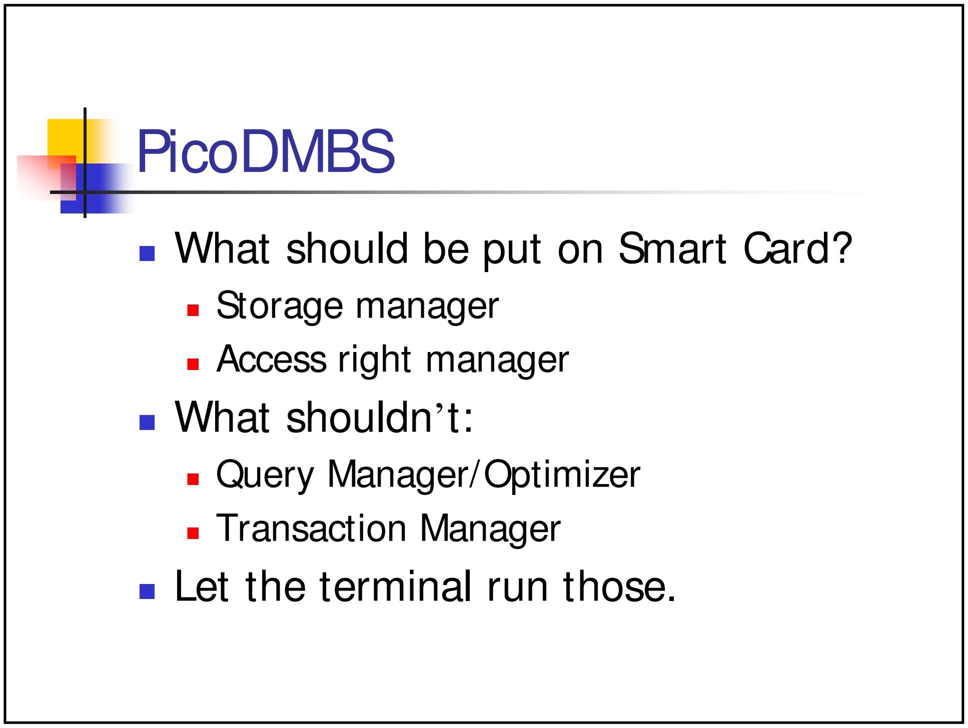
Trusted Terminal

- Use a hybrid PKI scheme to access data.
- All smart card data encrypted with appropriate read/write keys.
- Terminal must use correct keys to use, and make sense of, the data. Thus a trusted terminal can run the DBMS.
- Removes RAM, storage write delay, and processing speed restrictions from DBMS solution.



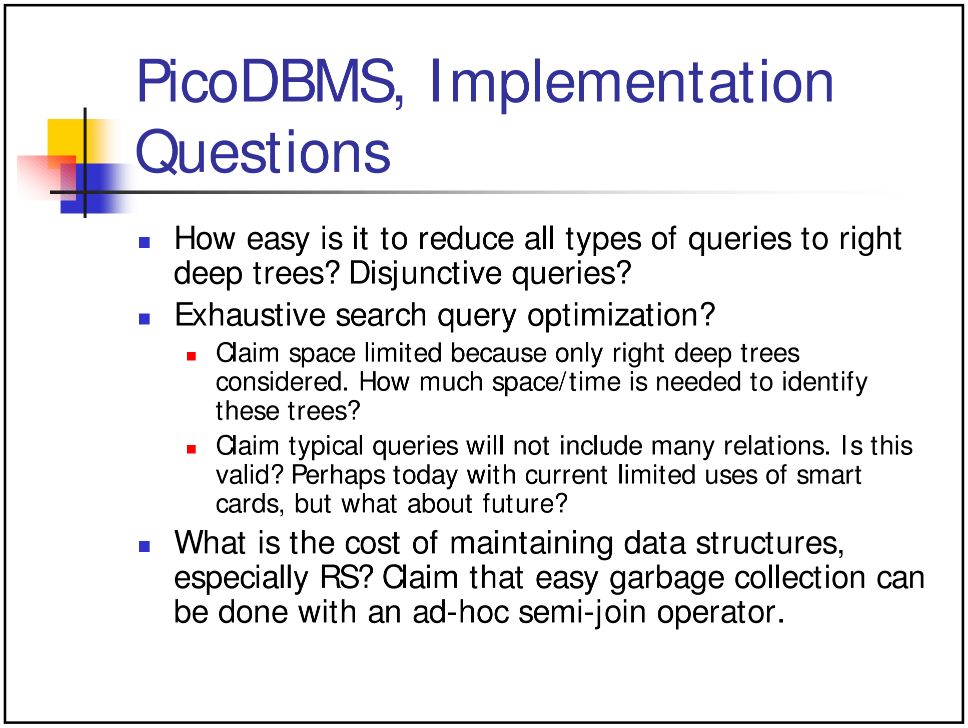
What are Real Benefits of Smart Cards?

- Availability (especially with contactless)
- Convenience, portability
- Security
  - Physical – cards are tamper proof.
  - Data – cards have strong encryption built into hardware/software
- Specialized local applications, and application sharing:
  - E.g. Health card shared with private Health care plan. When a treatment is provided terminal updates health card. Local application will also internally update Health care plan data as well, to be used to make a claim later.



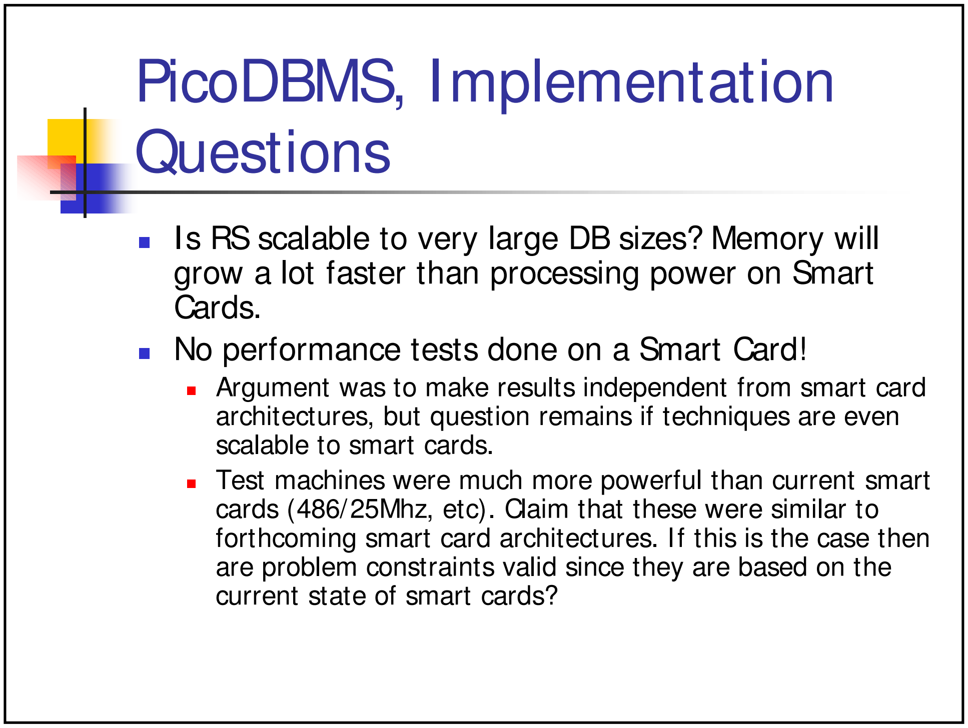
PicoDMBS

- What should be put on Smart Card?
  - Storage manager
  - Access right manager
- What shouldn’t:
  - Query Manager/Optimizer
  - Transaction Manager
- Let the terminal run those.



PicoDBMS, Implementation Questions

- How easy is it to reduce all types of queries to right deep trees? Disjunctive queries?
- Exhaustive search query optimization?
  - Claim space limited because only right deep trees considered. How much space/time is needed to identify these trees?
  - Claim typical queries will not include many relations. Is this valid? Perhaps today with current limited uses of smart cards, but what about future?
- What is the cost of maintaining data structures, especially RS? Claim that easy garbage collection can be done with an ad-hoc semi-join operator.



PicoDBMS, Implementation Questions

- Is RS scalable to very large DB sizes? Memory will grow a lot faster than processing power on Smart Cards.
- No performance tests done on a Smart Card!
  - Argument was to make results independent from smart card architectures, but question remains if techniques are even scalable to smart cards.
  - Test machines were much more powerful than current smart cards (486/25Mhz, etc). Claim that these were similar to forthcoming smart card architectures. If this is the case then are problem constraints valid since they are based on the current state of smart cards?

Discussion?