

Verifying Recursive Active Documents with Rewriting Rules

Blaise Genest

CNRS: IPAL, Singapore

IRISA, Rennes, France

Anca Muscholl LaBRI, Univ Bordeaux

Zhilin Wu Chinese Academy of Science, Beijing

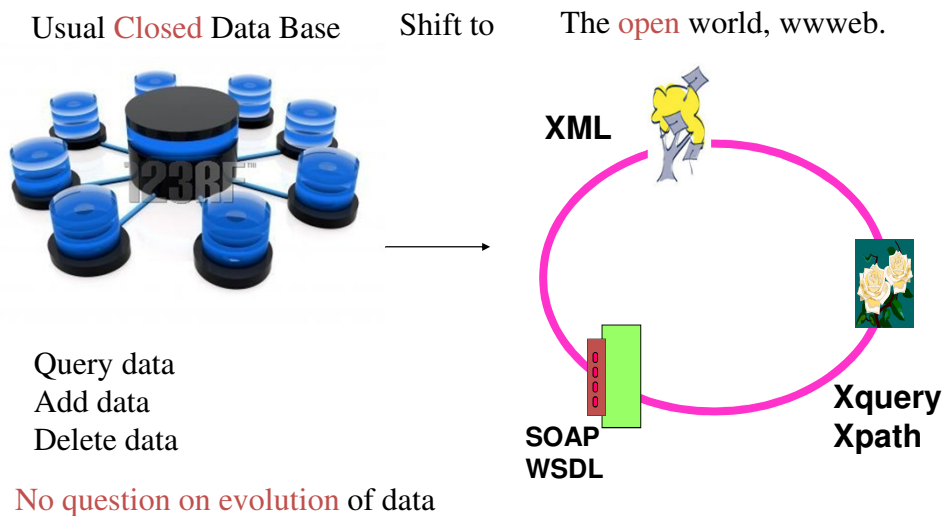
Frederic Blanqui?

INRIA, LIAMA, Beijing (Tsinghua -> CAS)

Shaofa Yang?

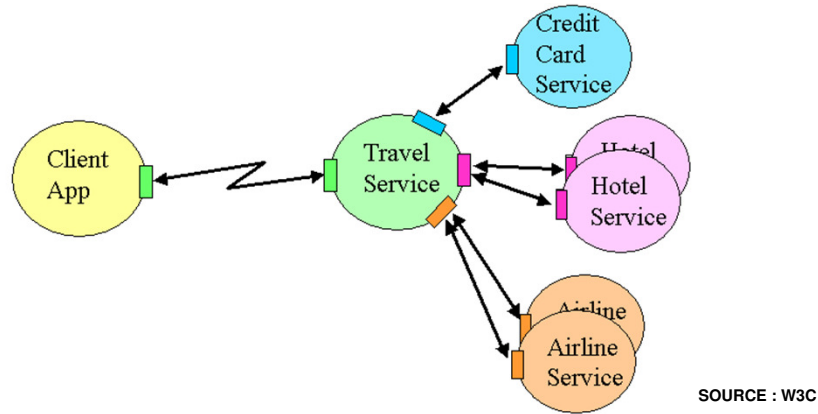
CAS, SIAT -> Beijing?

DataBase World Shift



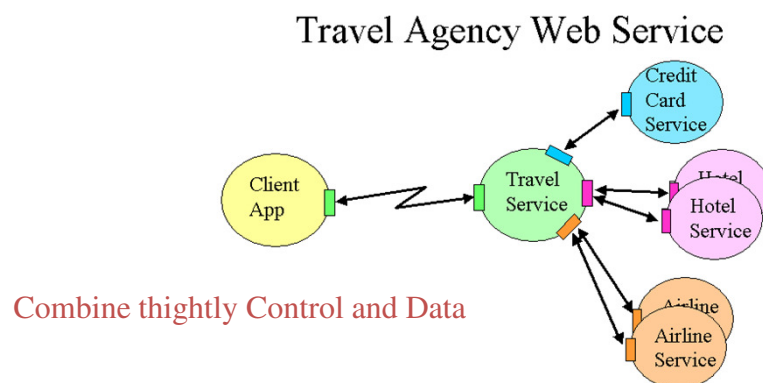
Open World: Web Services

Travel Agency Web Service



Composite (web) services, sharing parts of the same **data document**.
How these services combine? Unexpected **behaviors**?

Model and Algorithm for documents

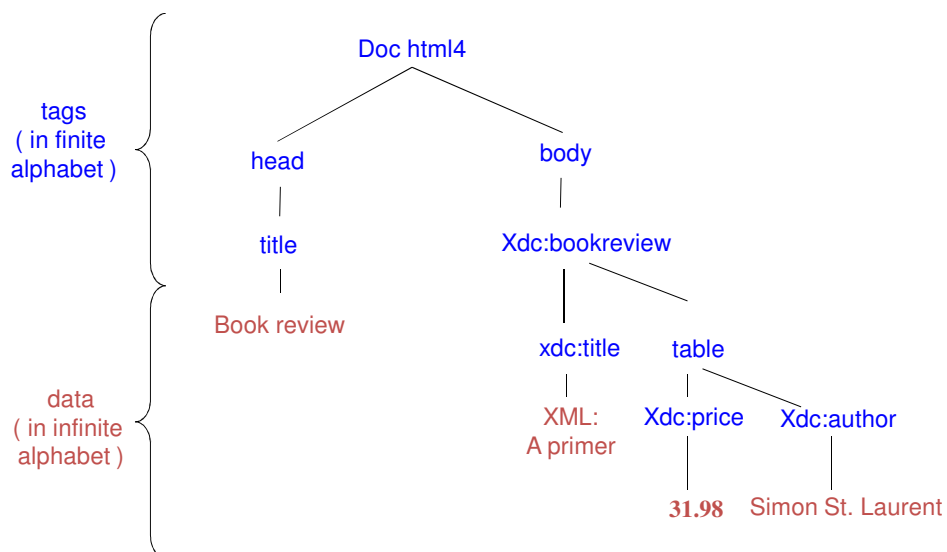


- I We need **models** to describe evolution of documents
- II We need **algorithms** on these models (**decidability** issue)
(to be used by designers of services)

Rich SemiStructured Data = XML

```
<html xmlns="http://www.w3.org/HTML/1998/html4"
      xmlns:xdc="http://www.xml.com/books">
  <head><title>Book Review</title></head>
  <body>
    <xdc:bookreview>
      <xdc:title>XML: A Primer</xdc:title>
      <table>
        <tr align="left">
          <td><xdc:author>Simon St. Laurent</xdc:author></td>
          <td><xdc:price>31.98</xdc:price></td>
        </tr>
      </table>
    </xdc:bookreview>
  </body>
</html>
```

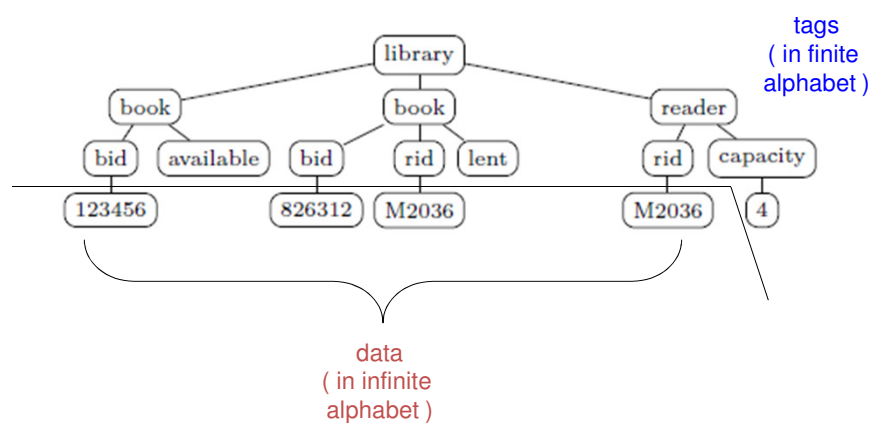
XML Documents modeled as Trees



DTPRS: Data Tree Pattern Rewrite System

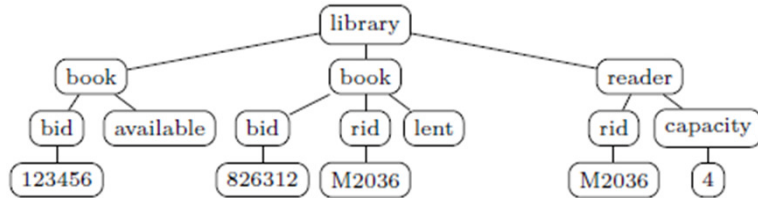
close in spirit to Guarded AXML
Abiteboul et al.
ANR MDCA « Docflow »

Example: Library



Here: 2 books (123456 and 826312), 1 reader (M2036).
One book available (123456) and one book is lent to reader M2036.
Reader M2036 can still borrow 4 other books (capacity).

Modeling evolution of document?



Evolution:

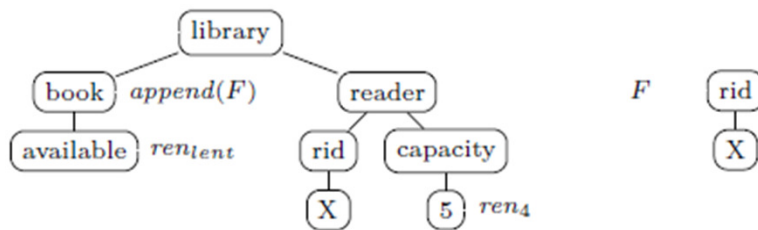
For tags:

- add (new book, new reader)
- delete (reader unsubscribe)
- change (available -> lent -> available)

For data:

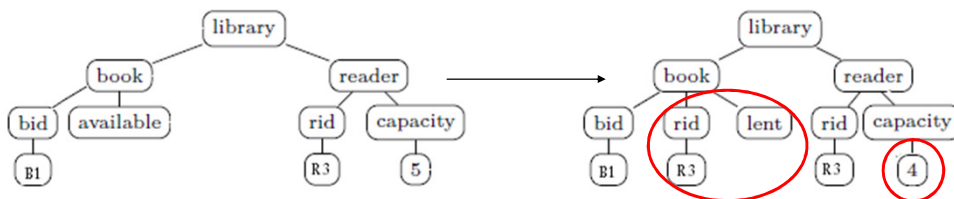
- copy (when reader borrows a book, its rid is copied)
- produce fresh data (bid/rid when new book arrives or new reader subscribes)

Example of DTPRS transition rules



Locator of transition-rule « reader-borrow-book »

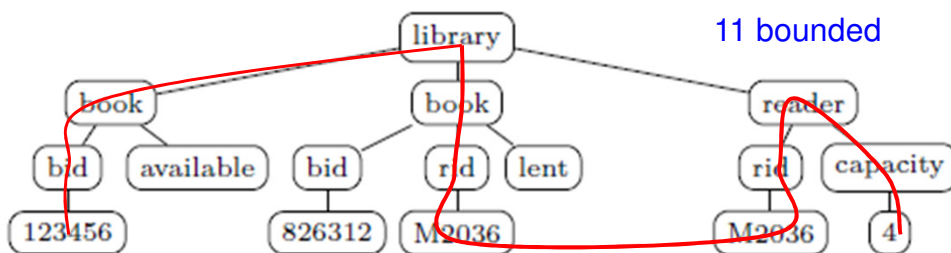
Evolution of the document when rule reader-borrow-book used.



Positive-Bounded DTPRS

Positive-bounded DTPRS if following restrictions :

- **positive** (no negation: cannot say something is not here)
- there is a **bound B** on the size of the **simple** paths on **every reachable document**, using parent/child relations and same data relation (implies depth B bounded).



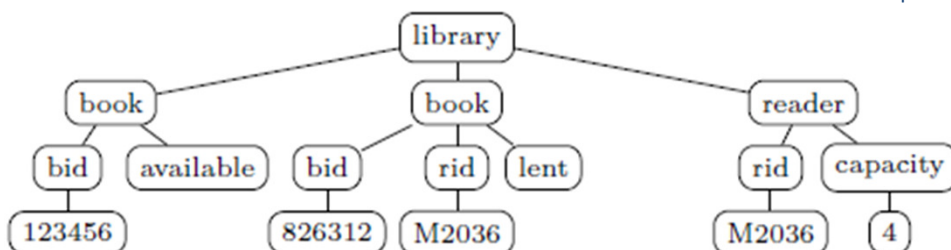
Results for Positive-Bounded DTPRS

Decidability: Termination, Reachability of a state/Tree Pattern

Undecidability : Confluence, Boundedness

Resort to theorem proving, Sat solving modulo theory

=> Frederic Blanqui



Conclusion

Future: Make it applicable!

- 1) Symbolic computation, way of representing infinitely many config
- 2) need good abstraction, refinement only when needed etc...
- 3) Use **complementary** techniques from **other** communities (AUR@)

