Frontiers of Tractability for Typechecking Simple XML Transformations

Wim Martens Frank Neven

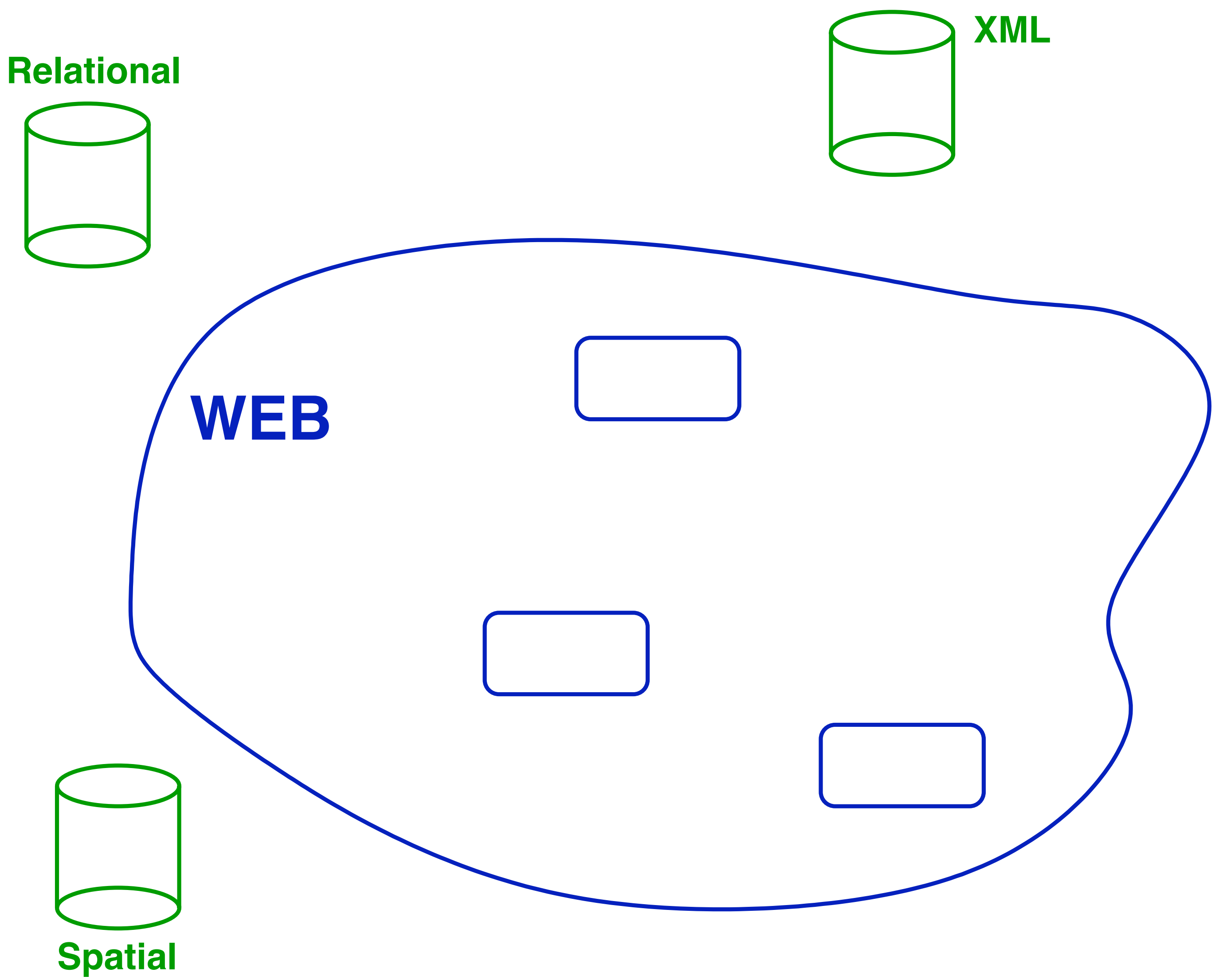
Limburgs Universitair Centrum

Belgium

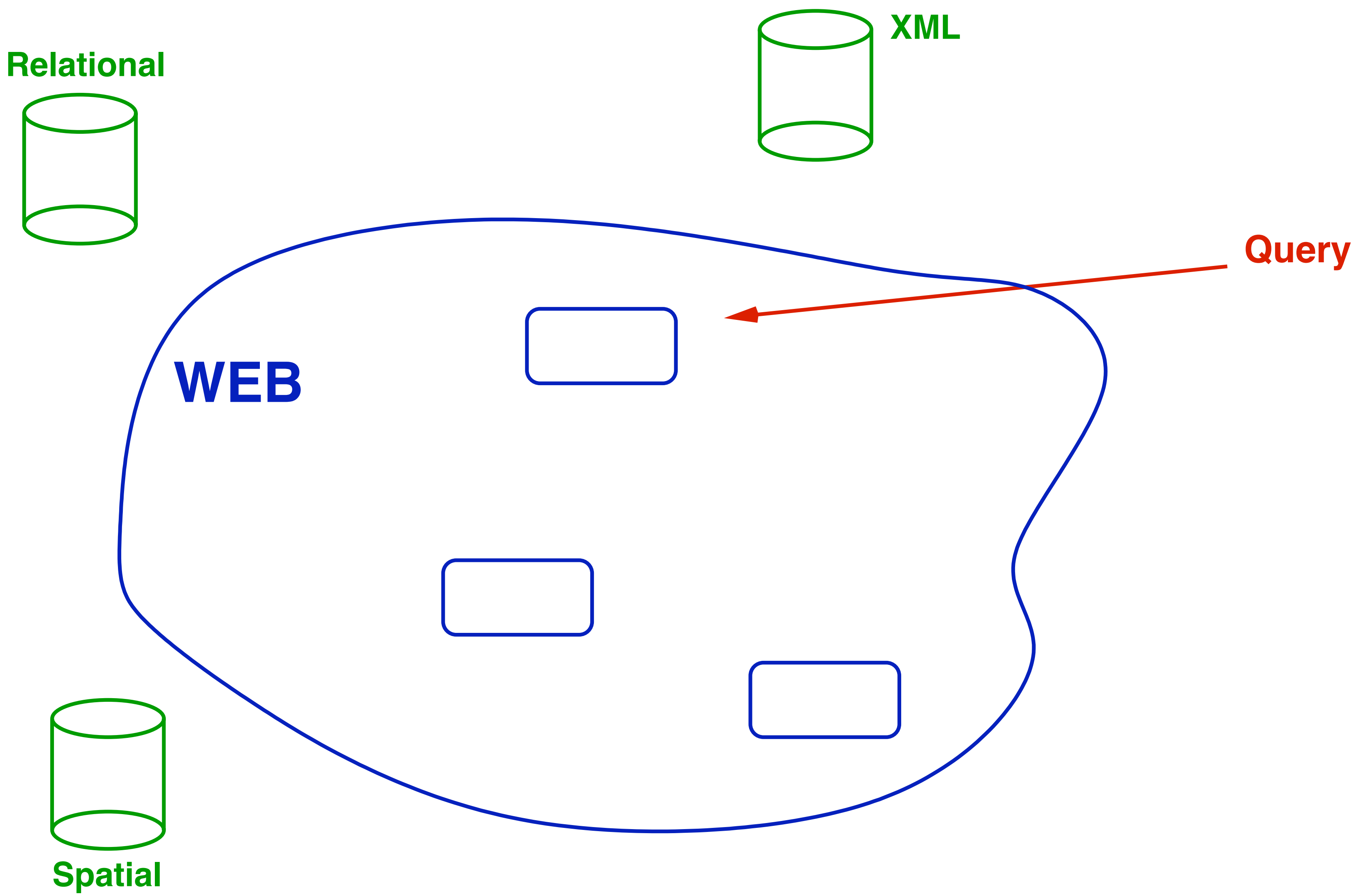
Overview

- Introduction
- Tree Languages
- Tree Transformations : XSLT
- The Typechecking Problem
- Tractable Deleting Transformations
- Tractable Copying Transformations
- Conclusion

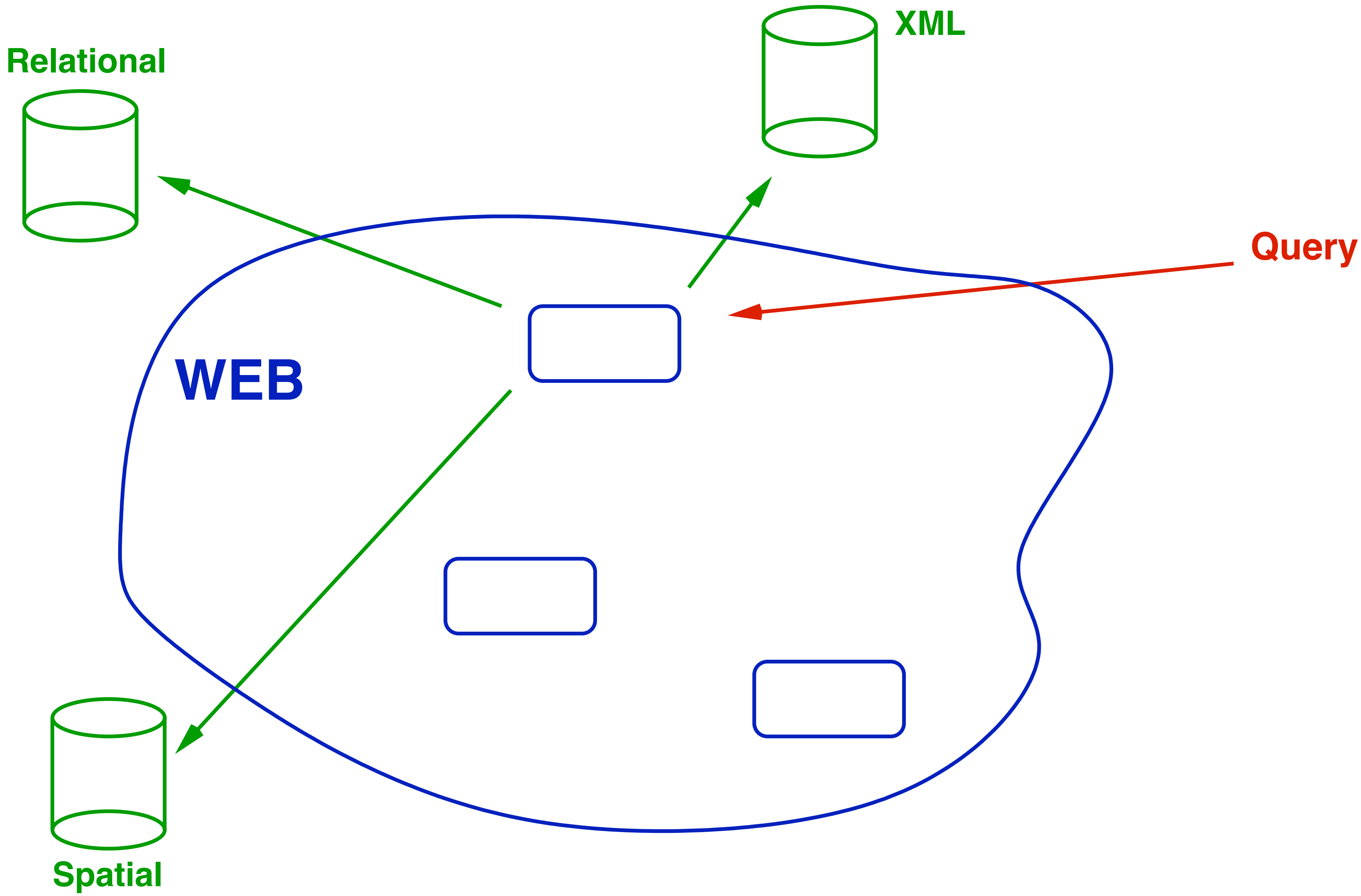
Data Integration on the Web



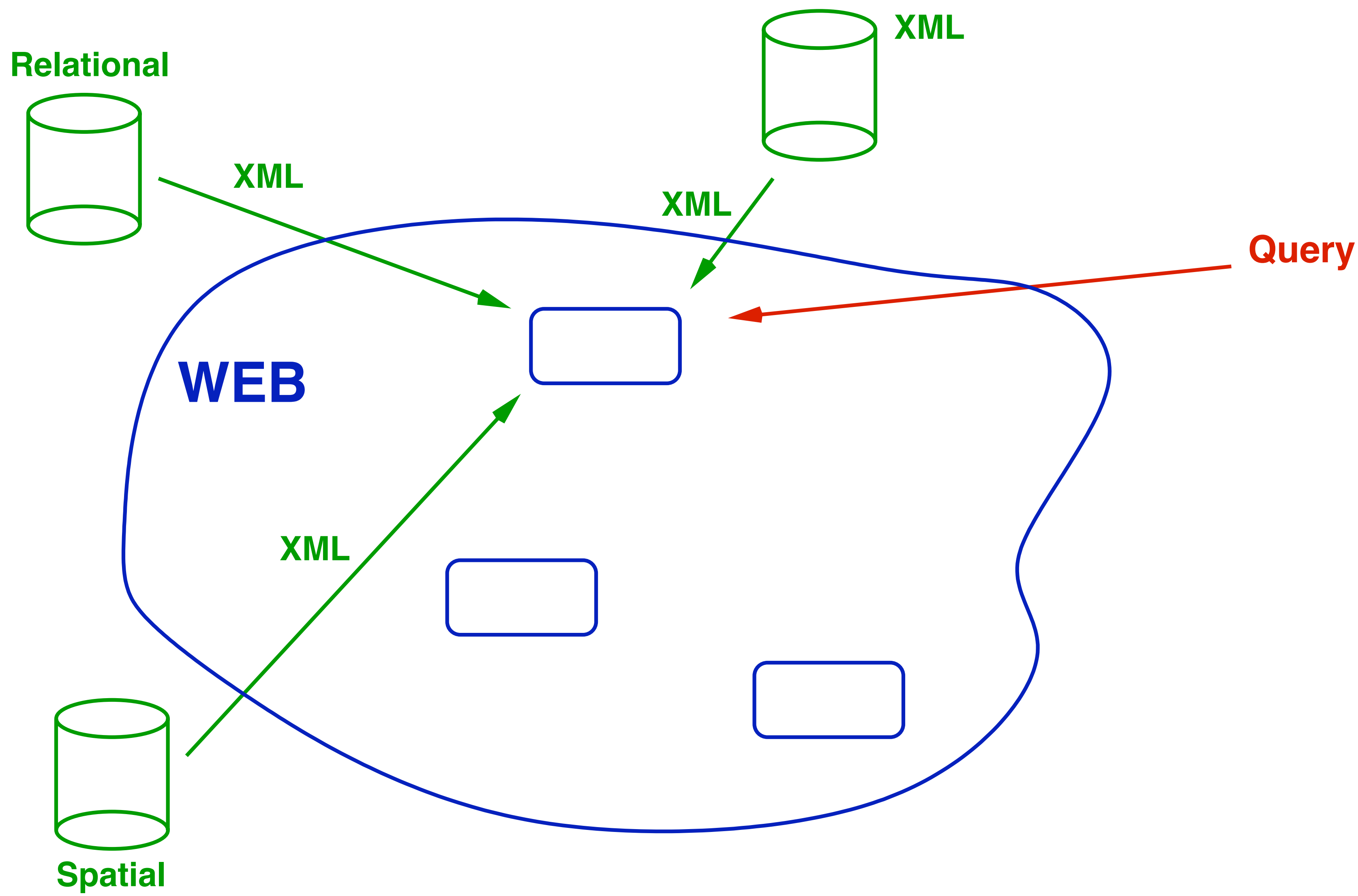
Data Integration on the Web



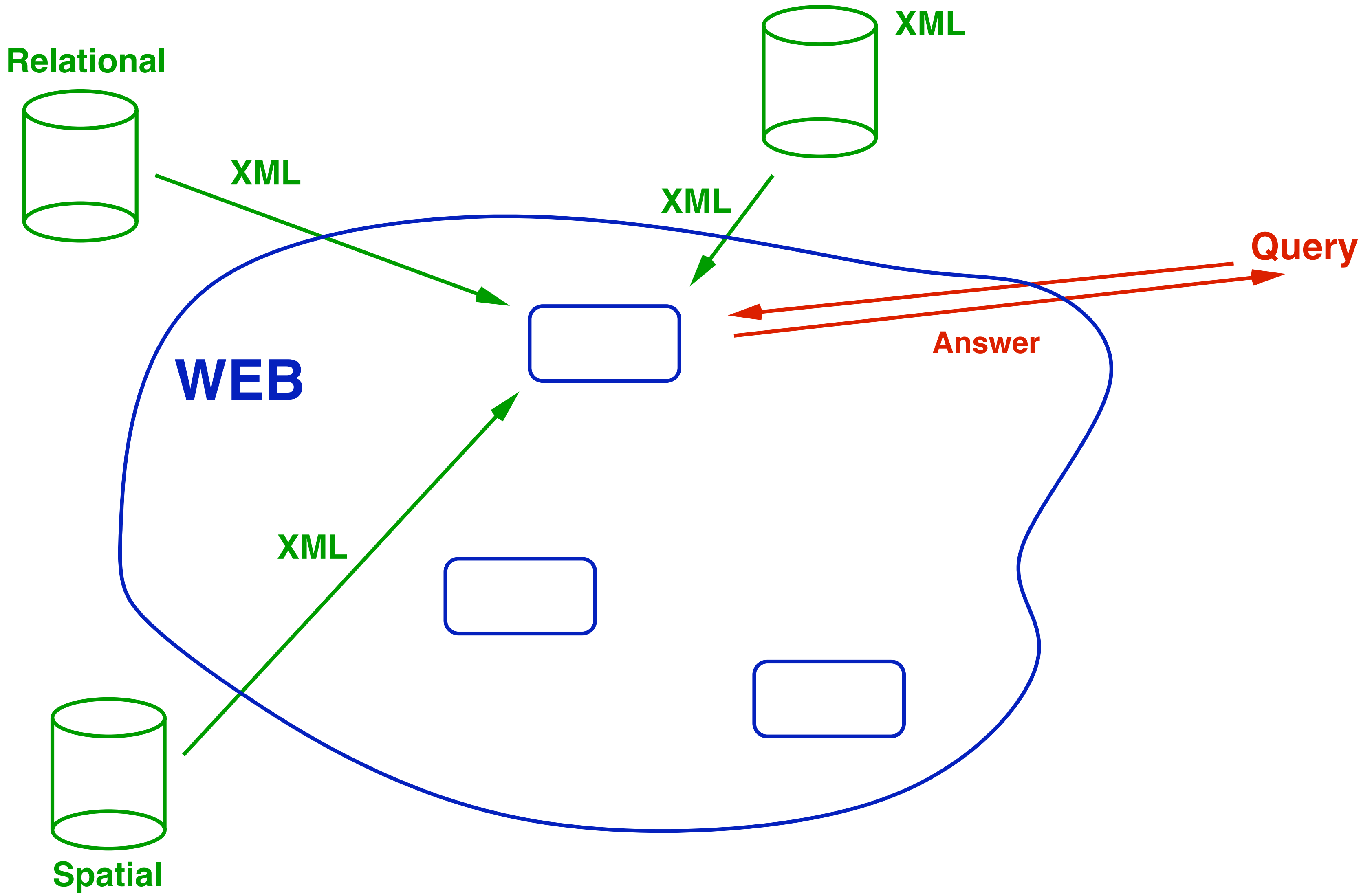
Data Integration on the Web



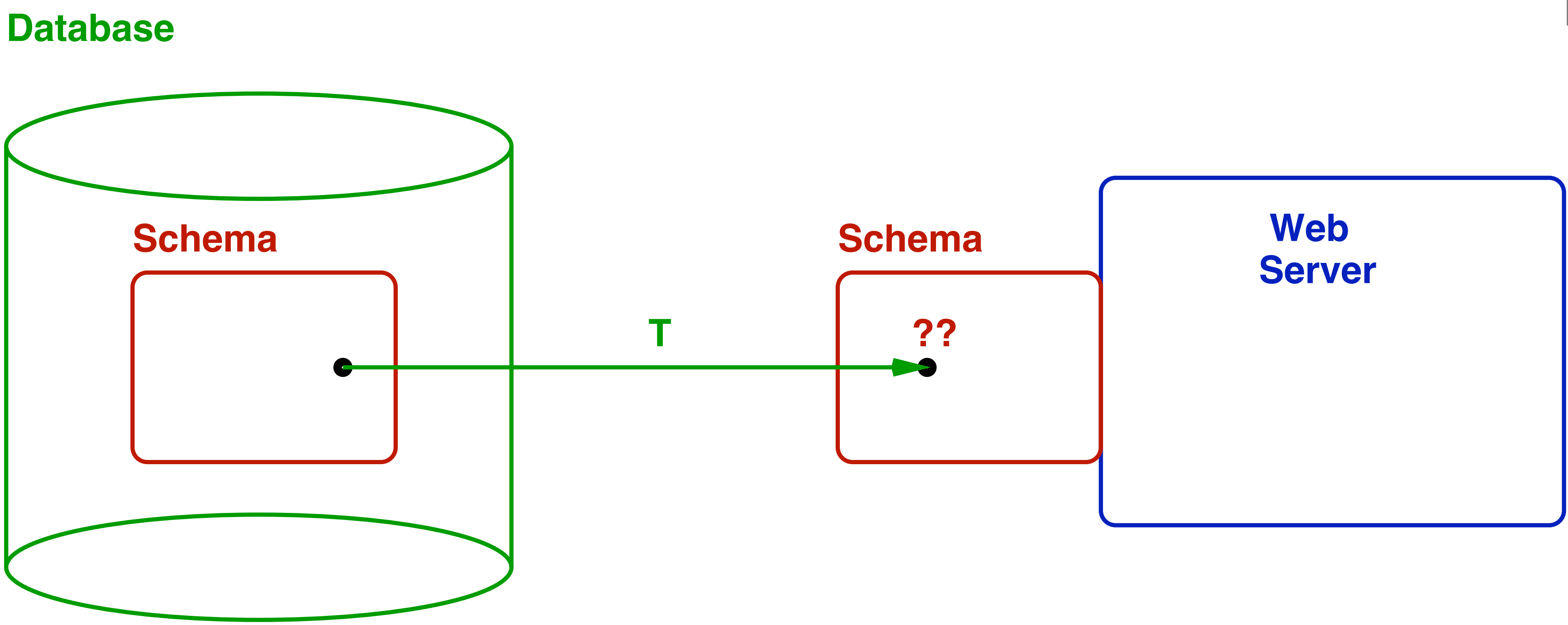
Data Integration on the Web



Data Integration on the Web



What is Typechecking?



Our Focus

XML to XML transformations

Transformation/query languages:

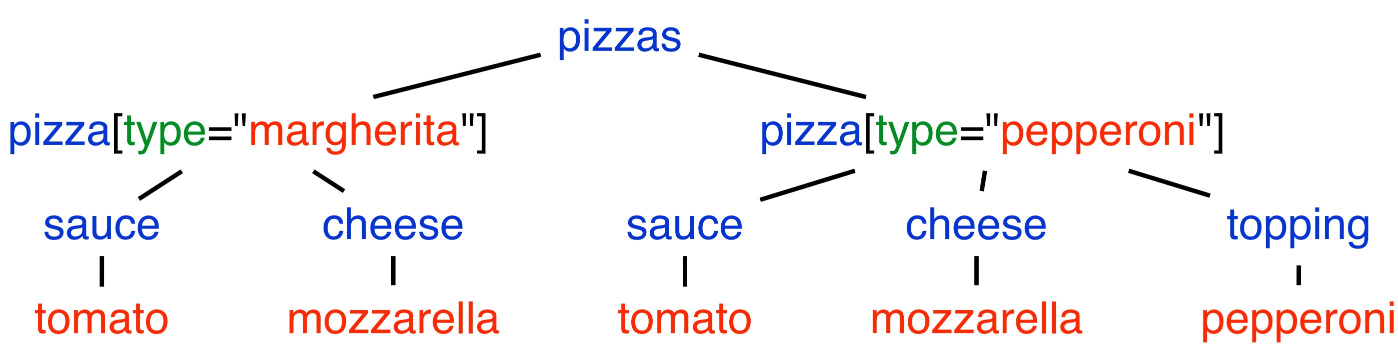
- XQuery
- XSLT

We focus on structural top-down XSLT transformations

(i.e. simple restructuring/filtering transformations)

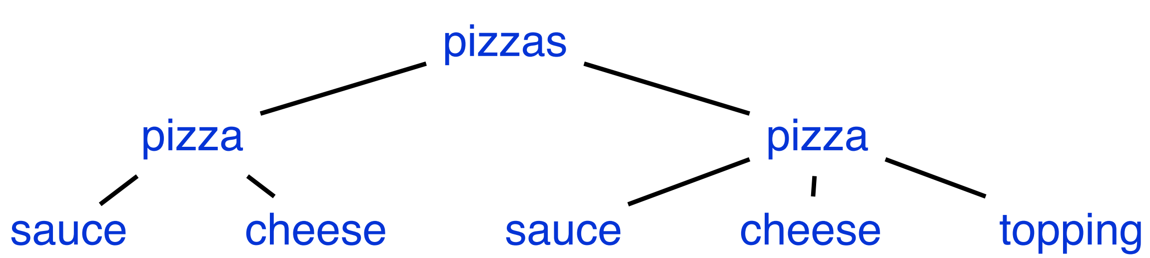
XML Documents are Trees

```
<pizzas>
    <pizza type="margherita">
        <sauce> tomato </sauce>
        <cheese> mozzarella </cheese>
    </pizza>
    <pizza type="pepperoni">
        <sauce> tomato </sauce>
        <cheese> mozzarella </cheese>
        <topping> pepperoni </topping>
    </pizza>
</pizzas>
```



XML Documents are Trees

```
<pizzas>
    <pizza type="margherita">
        <sauce> tomato </sauce>
        <cheese> mozzarella </cheese>
    </pizza>
    <pizza type="pepperoni">
        <sauce> tomato </sauce>
        <cheese> mozzarella </cheese>
        <topping> pepperoni </topping>
    </pizza>
</pizzas>
```



Previous Results on XML Typechecking

- Alon et al (2001) :
  - Typechecking quickly turns undecidable when data or attribute values are incorporated.
- Milo, Suciu, Vianu (2000) :
  - When only looking at structural properties of trees, typechecking is decidable for a large fragment of tree transformations (formalized by k-pebble transducers).
  - Complexity is high (non-elementary).

Overview

- Introduction
- Tree Languages
- Tree Transformations : XSLT
- The Typechecking Problem
- Tractable Deleting Transformations
- Tractable Copying Transformations
- Conclusion

Tree Languages

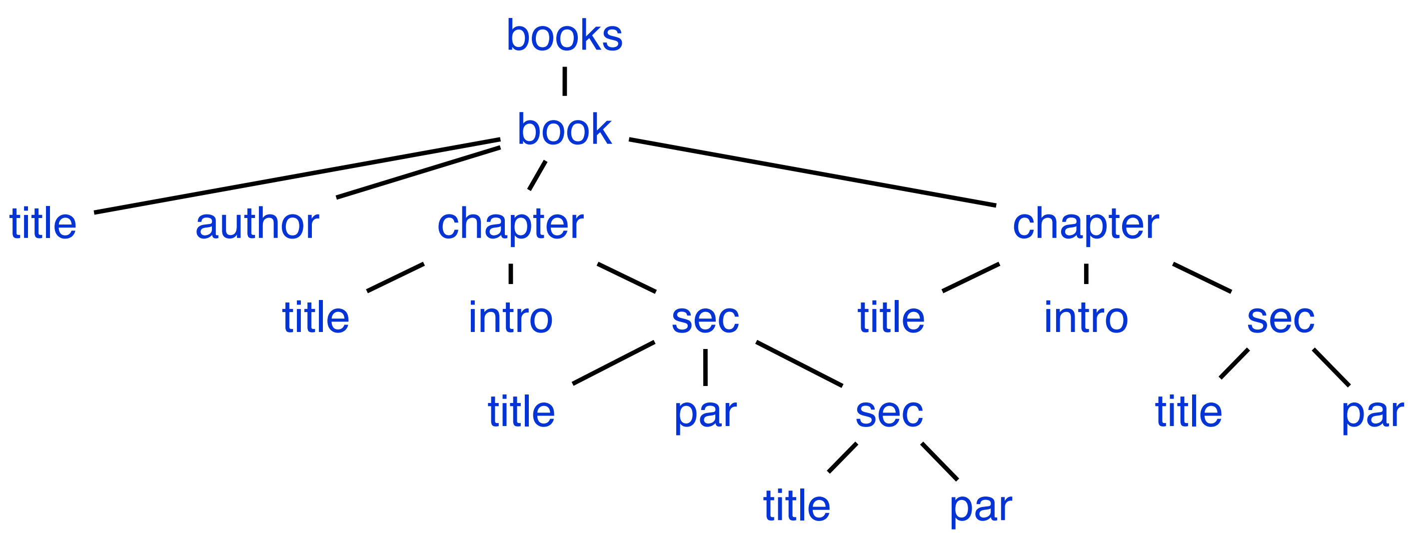
- DTDs:

$$\begin{array}{lll} \texttt{books} & \rightarrow & \texttt{book}^* \\ \texttt{book} & \rightarrow & \texttt{title},\ \texttt{author}^+,\ \texttt{chapter}^+ \\ \texttt{chapter} & \rightarrow & \texttt{title},\ \texttt{intro},\ \texttt{sec}^+ \\ \texttt{sec} & \rightarrow & \texttt{title},\ \texttt{par}^+,\ \texttt{sec}^* \end{array}$$

Tree Languages

- DTDs:

$$
\begin{array}{lcl}
\texttt{books} & \rightarrow & \texttt{book}^{*} \\
\texttt{book} & \rightarrow & \texttt{title},\ \texttt{author}^{+},\ \texttt{chapter}^{+} \\
\texttt{chapter} & \rightarrow & \texttt{title},\ \texttt{intro},\ \texttt{sec}^{+} \\
\texttt{sec} & \rightarrow & \texttt{title},\ \texttt{par}^{+},\ \texttt{sec}^{*}
\end{array}
$$



Overview

- Introduction
- Tree Languages
- Tree Transformations : XSLT
- The Typechecking Problem
- Tractable Deleting Transformations
- Tractable Copying Transformations
- Conclusion

XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \longrightarrow \begin{array}{c}\text{books}\\ | \\ q_0\end{array}$

$(p,\text{chapter}) \longrightarrow \text{chapter}\ p$

$(p,\text{title}) \longrightarrow \text{title}$

$(p,\text{sec}) \longrightarrow p$

$(q_0,\text{book}) \longrightarrow \begin{array}{ccc} & \text{book} & \\ / & & \backslash \\ p & & q\end{array}$

$(q,\text{chapter}) \longrightarrow \text{chapter}\ q'$

$(q',\text{title}) \longrightarrow \text{title}$

$(q',\text{intro}) \longrightarrow \text{intro}$

XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \to \text{books}(q_0)$

$(q_0,\text{book}) \to \text{book}(p\ q)$

$(p,\text{chapter}) \to \text{chapter}\ p$

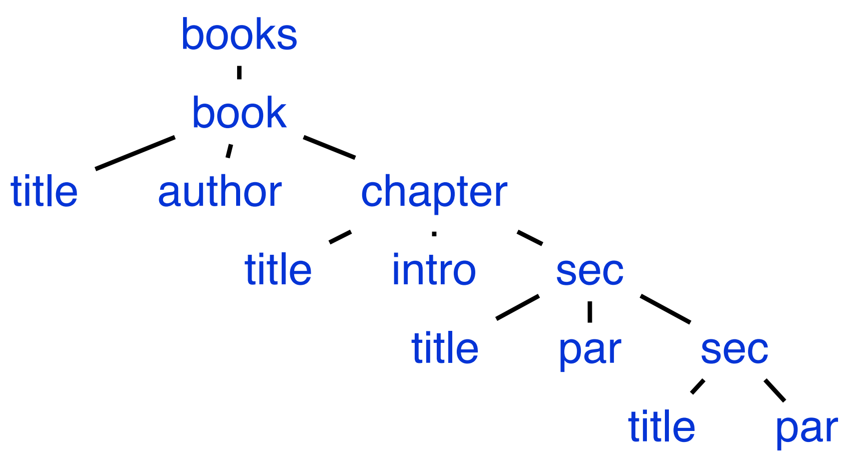
$(q,\text{chapter}) \to \text{chapter}\ q'$

$(p,\text{title}) \to \text{title}$

$(q',\text{title}) \to \text{title}$

$(p,\text{sec}) \to p$

$(q',\text{intro}) \to \text{intro}$



XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \longrightarrow$ books
q_0

$(p,\text{chapter}) \longrightarrow \text{chapter}\ p$

$(p,\text{title}) \longrightarrow \text{title}$

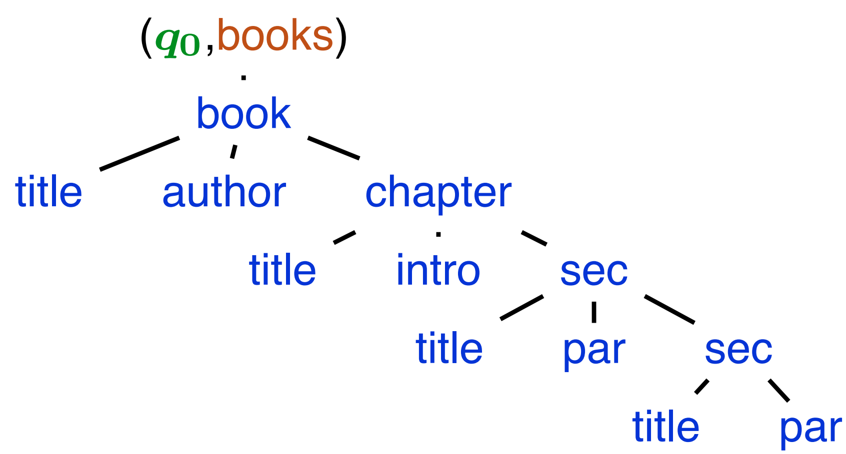
$(p,\text{sec}) \longrightarrow p$

$(q_0,\text{book}) \longrightarrow$ book
$p \quad q$

$(q,\text{chapter}) \longrightarrow \text{chapter}\ q'$

$(q',\text{title}) \longrightarrow \text{title}$

$(q',\text{intro}) \longrightarrow \text{intro}$



XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \longrightarrow \text{books}(q_0)$

$(p,\text{chapter}) \longrightarrow \text{chapter}\ p$

$(p,\text{title}) \longrightarrow \text{title}$

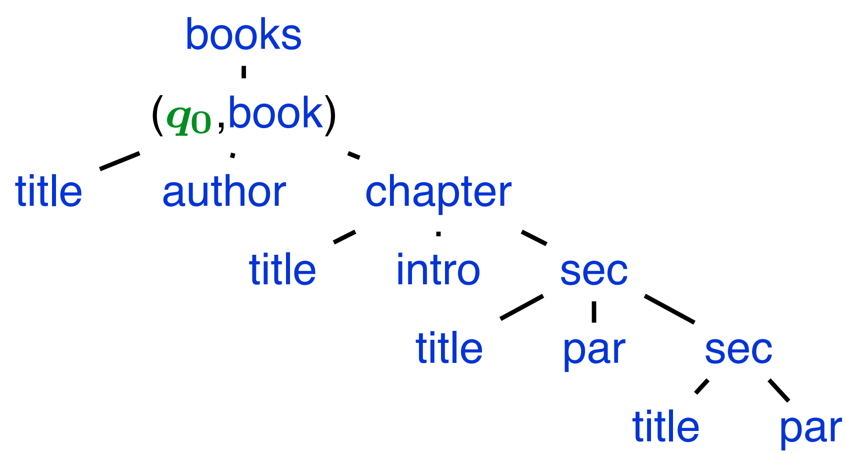
$(p,\text{sec}) \longrightarrow p$

$(q_0,\text{book}) \longrightarrow \text{book}(p\ q)$

$(q,\text{chapter}) \longrightarrow \text{chapter}\ q'$

$(q',\text{title}) \longrightarrow \text{title}$

$(q',\text{intro}) \longrightarrow \text{intro}$



XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \longrightarrow \text{books}(q_0)$

$(q_0,\text{book}) \longrightarrow \text{book}(p\ q)$

$(p,\text{chapter}) \longrightarrow \text{chapter}\ p$

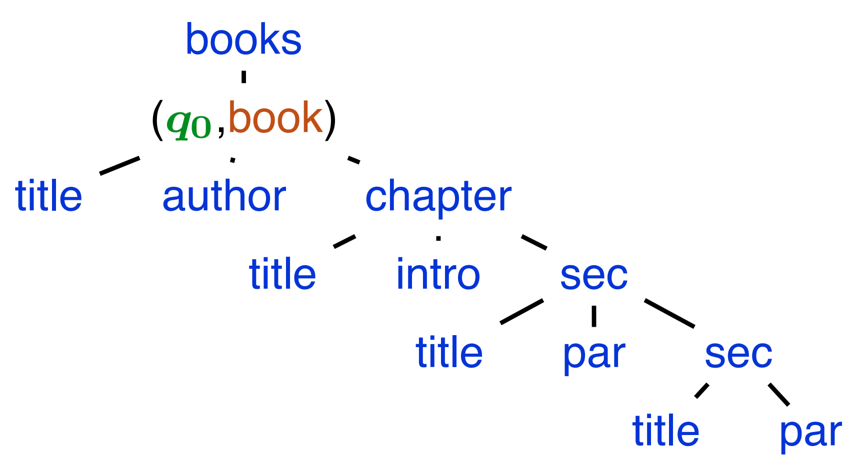
$(q,\text{chapter}) \longrightarrow \text{chapter}\ q'$

$(p,\text{title}) \longrightarrow \text{title}$

$(q',\text{title}) \longrightarrow \text{title}$

$(p,\text{sec}) \longrightarrow p$

$(q',\text{intro}) \longrightarrow \text{intro}$



XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \rightarrow \begin{array}{c}\text{books}\\ | \\ q_0\end{array}$

$(p,\text{chapter}) \rightarrow \text{chapter}\ p$

$(p,\text{title}) \rightarrow \text{title}$

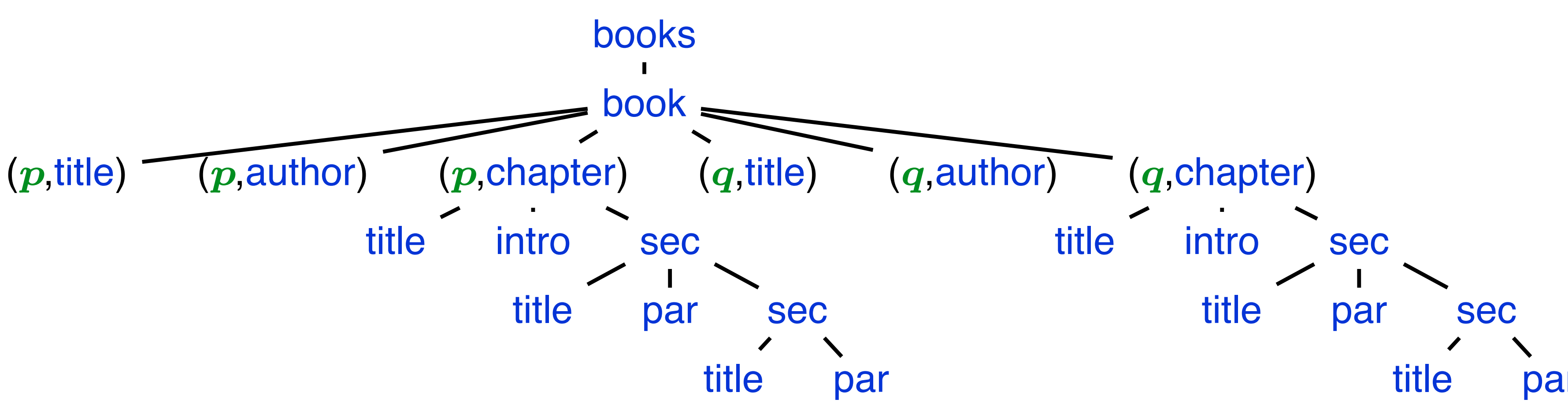
$(p,\text{sec}) \rightarrow p$

$(q_0,\text{book}) \rightarrow \begin{array}{c}\text{book}\\ / \quad \backslash \\ p \qquad q\end{array}$

$(q,\text{chapter}) \rightarrow \text{chapter}\ q'$

$(q',\text{title}) \rightarrow \text{title}$

$(q',\text{intro}) \rightarrow \text{intro}$



XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0, \text{books}) \rightarrow \text{books}(q_0)$

$(p, \text{chapter}) \rightarrow \text{chapter } p$

$(p, \text{title}) \rightarrow \text{title}$

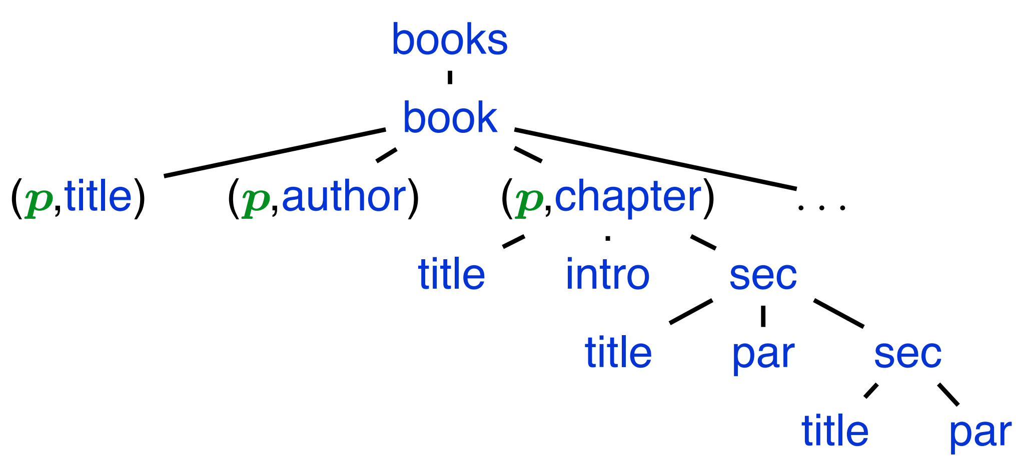
$(p, \text{sec}) \rightarrow p$

$(q_0, \text{book}) \rightarrow \text{book}(p, q)$

$(q, \text{chapter}) \rightarrow \text{chapter } q'$

$(q', \text{title}) \rightarrow \text{title}$

$(q', \text{intro}) \rightarrow \text{intro}$



XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \longrightarrow \text{books}(q_0)$

$(p,\text{chapter}) \longrightarrow \text{chapter}\ p$

$(p,\text{title}) \longrightarrow \text{title}$

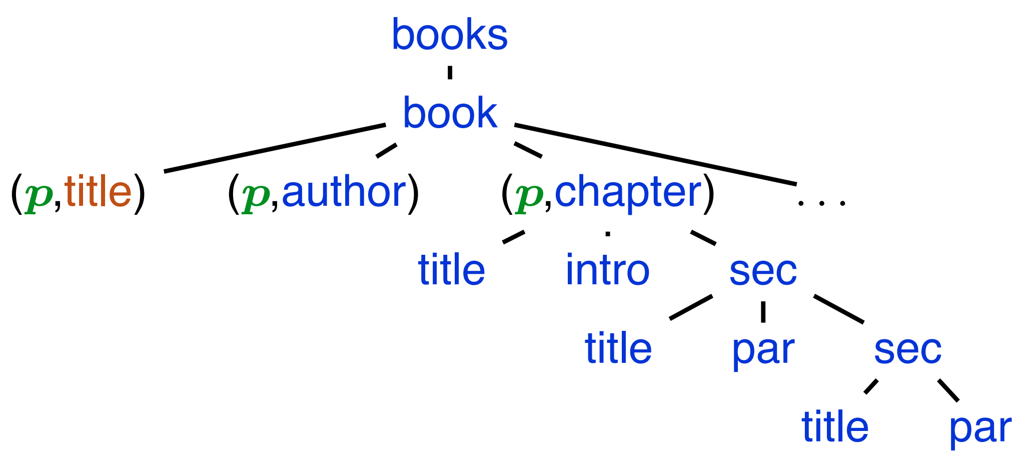
$(p,\text{sec}) \longrightarrow p$

$(q_0,\text{book}) \longrightarrow \text{book}(p\ q)$

$(q,\text{chapter}) \longrightarrow \text{chapter}\ q'$

$(q',\text{title}) \longrightarrow \text{title}$

$(q',\text{intro}) \longrightarrow \text{intro}$



XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \rightarrow \text{books}$
$\qquad q_0$

$(q_0,\text{book}) \rightarrow \text{book}$
$\qquad p \quad q$

$(p,\text{chapter}) \rightarrow \text{chapter}\ p$

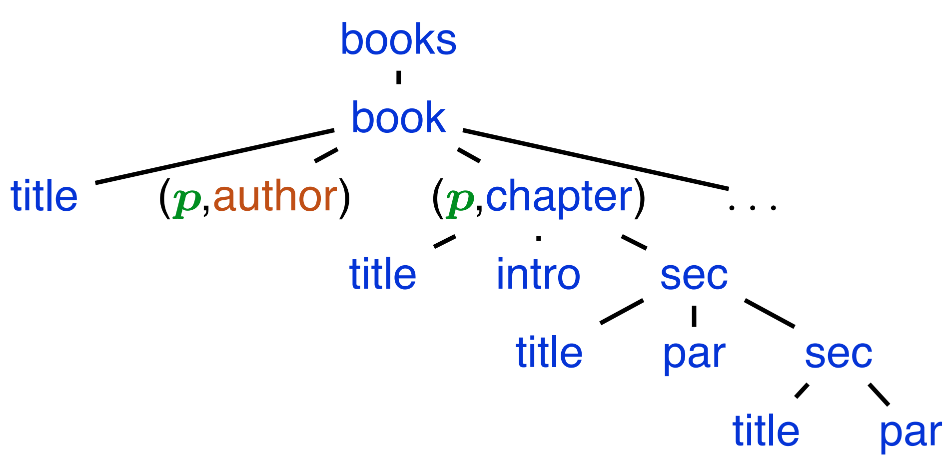
$(q,\text{chapter}) \rightarrow \text{chapter}\ q'$

$(p,\text{title}) \rightarrow \text{title}$

$(q',\text{title}) \rightarrow \text{title}$

$(p,\text{sec}) \rightarrow p$

$(q',\text{intro}) \rightarrow \text{intro}$



XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \rightarrow \text{books}(q_0)$

$(q_0,\text{book}) \rightarrow \text{book}(p\ q)$

$(p,\text{chapter}) \rightarrow \text{chapter}\ p$

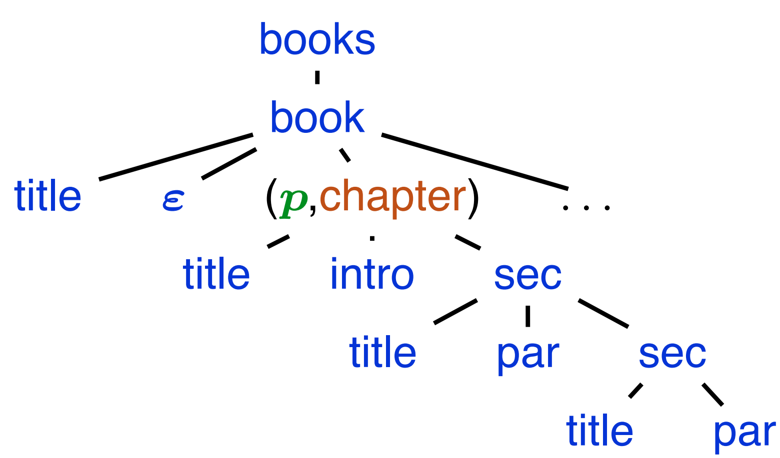
$(q,\text{chapter}) \rightarrow \text{chapter}\ q'$

$(p,\text{title}) \rightarrow \text{title}$

$(q',\text{title}) \rightarrow \text{title}$

$(p,\text{sec}) \rightarrow p$

$(q',\text{intro}) \rightarrow \text{intro}$



XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \to$ books
$\quad\quad\quad\quad\quad\quad\quad q_0$

$(q_0,\text{book}) \to$ book
$\quad\quad\quad\quad\quad\quad\quad p \quad q$

$(p,\text{chapter}) \to \text{chapter}\ p$

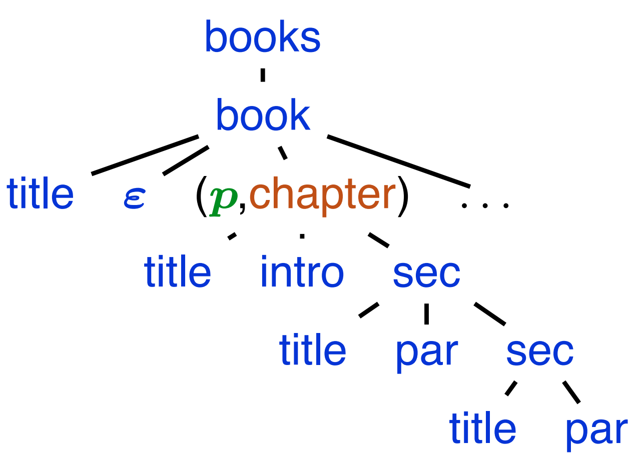
$(q,\text{chapter}) \to \text{chapter}\ q'$

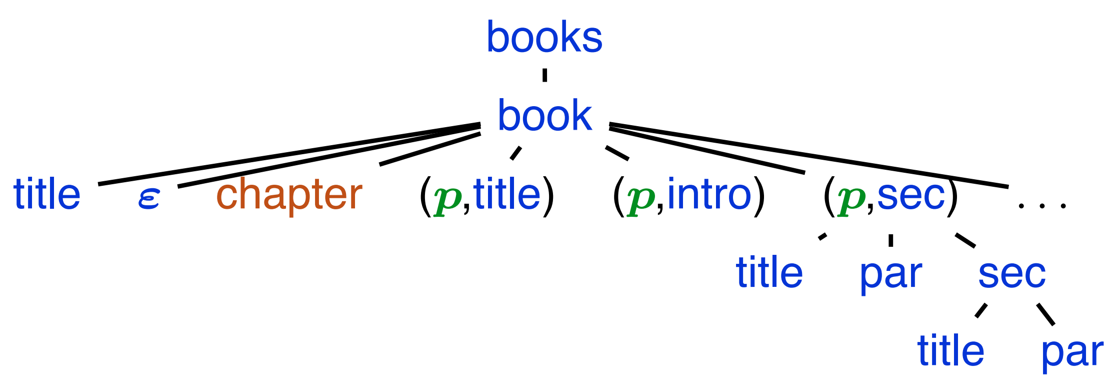
$(p,\text{title}) \to \text{title}$

$(q',\text{title}) \to \text{title}$

$(p,\text{sec}) \to p$

$(q',\text{intro}) \to \text{intro}$





XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \to \text{books}(q_0)$

$(q_0,\text{book}) \to \text{book}(p\ q)$

$(p,\text{chapter}) \to \text{chapter}\ p$

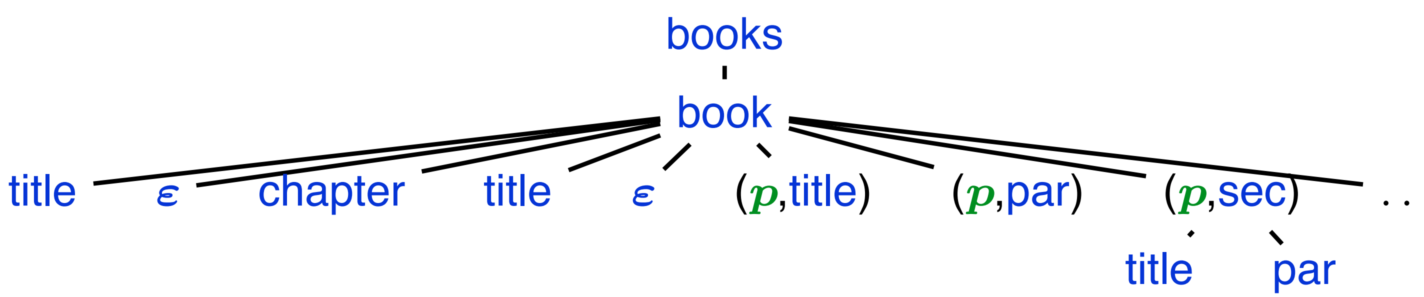
$(q,\text{chapter}) \to \text{chapter}\ q'$

$(p,\text{title}) \to \text{title}$

$(q',\text{title}) \to \text{title}$

$(p,\text{sec}) \to p$

$(q',\text{intro}) \to \text{intro}$



XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0, \text{books}) \to \text{books}(q_0)$

$(p, \text{chapter}) \to \text{chapter}\ p$

$(p, \text{title}) \to \text{title}$

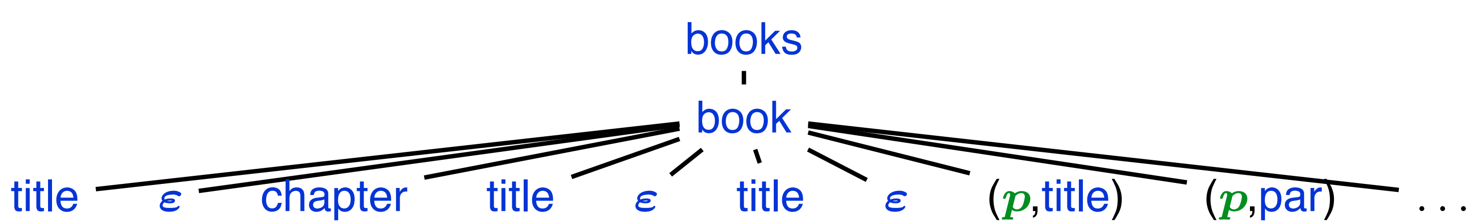
$(p, \text{sec}) \to p$

$(q_0, \text{book}) \to \text{book}(p\ q)$

$(q, \text{chapter}) \to \text{chapter}\ q'$

$(q', \text{title}) \to \text{title}$

$(q', \text{intro}) \to \text{intro}$



XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \longrightarrow \text{books}(q_0)$

$(p,\text{chapter}) \longrightarrow \text{chapter}\ p$

$(p,\text{title}) \longrightarrow \text{title}$

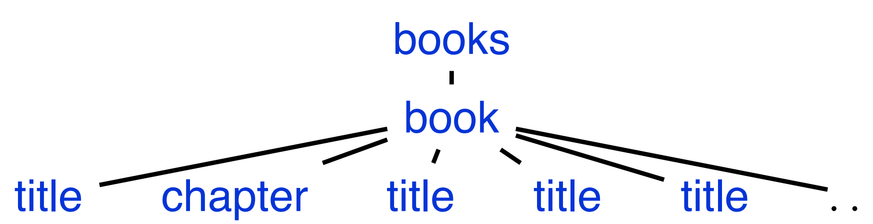
$(p,\text{sec}) \longrightarrow p$

$(q_0,\text{book}) \longrightarrow \text{book}(p\ \ q)$

$(q,\text{chapter}) \longrightarrow \text{chapter}\ q'$

$(q',\text{title}) \longrightarrow \text{title}$

$(q',\text{intro}) \longrightarrow \text{intro}$



XSLT: Example

Generate for each book the list of titles for each chapter, followed by a summary of the book.

$(q_0,\text{books}) \to \text{books}(q_0)$

$(q_0,\text{book}) \to \text{book}(p\ q)$

$(p,\text{chapter}) \to \text{chapter}\ p$

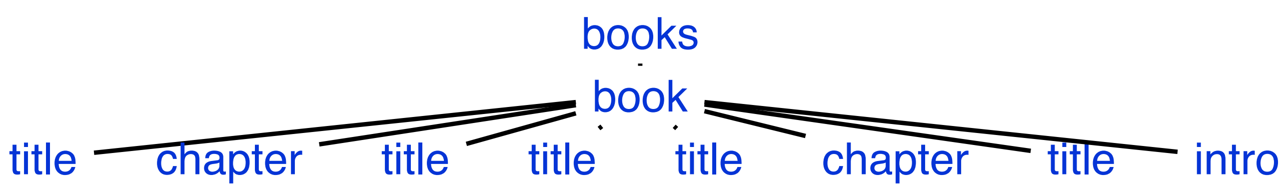
$(q,\text{chapter}) \to \text{chapter}\ q'$

$(p,\text{title}) \to \text{title}$

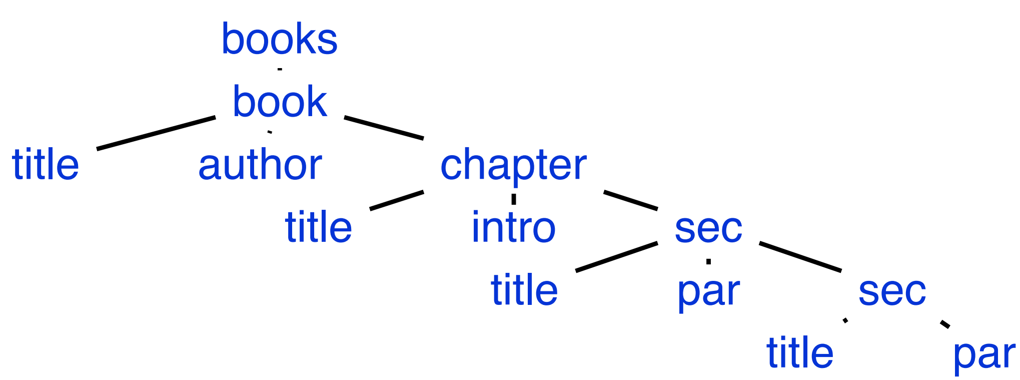
$(q',\text{title}) \to \text{title}$

$(p,\text{sec}) \to p$

$(q',\text{intro}) \to \text{intro}$



Original tree:



Overview

- Introduction
- Tree Languages
- Tree Transformations : XSLT
- The Typechecking Problem
- Tractable Deleting Transformations
- Tractable Copying Transformations
- Conclusion

The Typechecking Problem

Given:

- input tree language L_{in} DTD
- output tree language L_{out} DTD
- XML-transformation T XSLT

Is it true that,

$$\forall t \in L_{in} : T(t) \in L_{out}?$$

Known Results

The complexity of the typechecking problem [M., Neven 03]:

	Tree Automata	DTD(NFA)	DTD(DFA)
deletion, unbounded copy	**EXPTIME**	**EXPTIME**	**EXPTIME**
deletion, bounded copy	**EXPTIME**	**EXPTIME**	**EXPTIME**
no deletion, unbounded copy	**EXPTIME**	**PSPACE**	**PSPACE**
no deletion, bounded copy	**EXPTIME**	**PSPACE**	**PTIME**

The **PTIME** fragment is very restricted

Goal: enlarge **PTIME** fragment

Overview

- Introduction
- Tree Languages
- Tree Transformations : XSLT
- The Typechecking Problem
- Tractable Deleting Transformations
- Tractable Copying Transformations
- Conclusion

What can we Delete?

Parametrize how much a certain transformation deletes:

Formally:

$$
\begin{array}{ll}
(q_1, a) \to q_2\, a\, q_2\, q_2\, a & (q_5, a) \to q_6\, a\, a \\
(q_2, a) \to a\, q_3\, a & (q_6, a) \to q_7 \\
(q_3, a) \to q_4\, q_4 & (q_7, a) \to a\, q_8\, a \\
(q_4, a) \to a & (q_8, a) \to a\, a\, q_7
\end{array}
$$

state	q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8
deletion depth	3	2	1	0	∞	∞	∞	∞

$\infty \equiv$ "unbounded"

What can we Delete?

$$
\begin{array}{ll}
(q_1, a) \rightarrow q_2\, a\, q_2\, q_2\, a & (q_5, a) \rightarrow q_6\, a\, a \\
(q_2, a) \rightarrow a\, q_3\, a & (q_6, a) \rightarrow q_7 \\
(q_3, a) \rightarrow q_4\, q_4 & (q_7, a) \rightarrow a\, q_8\, a \\
(q_4, a) \rightarrow a & (q_8, a) \rightarrow a\, a\, q_7
\end{array}
$$

- del depth D: maximally skip branches of length D
- del width W: maximal no of states on top of rhs
- copy width C: maximal no of states in rhs

$\mathcal{T}_{\text{trac}}$: $max(C)$ and $max(W^D)$ are constant

What can we Delete?

Theorem: $TC[\mathcal{T}_{\text{trac}}, DTD(DFA)]$ is in time $\mathcal{O}(n^{C \cdot W^D})$

Why is this relevant?

$C \cdot W^D$ is usually very small in practice, i.e.

- 1 for filtering transformations
- 2 or 3 for most other restructuring transformations

What can we Delete?

Theorem: $TC[\mathcal{T}_{\text{trac}}, DTD(DFA)]$ is in time $\mathcal{O}(n^{C \cdot W^D})$

What does $\mathcal{T}_{\text{trac}}$ allow?

Generate the list of titles for each chapter, followed by a summary of the book.

$(q_0$,book$) \rightarrow$ book$(p\ q)$

$(q$,chapter$) \rightarrow$ chapter q'

$(q'$,title$) \rightarrow$ title

$(q'$,intro$) \rightarrow$ intro

$(p$,chapter$) \rightarrow$ chapter p

$(p$,title$) \rightarrow$ title

$(p$,sec$) \rightarrow p$

q_0: $C = 2$

p: $C = 1$ $W = 1$ $D = \infty$

q: $C = 1$ $W = 1$ $D = 1$

q': $C = 0$

Recursive deletion only

when no simultaneous copy!

$\cancel{(q, a) \rightarrow qq}$

So, $C = 2$ and $W^D = 1$

What can we Delete?

Theorem: $TC[\mathcal{T}_{\text{trac}}, DTD(DFA)]$ is in **PTIME**

Can we extend this?

1. If C is unbounded, then TC is **PSPACE**-hard.
2. W^D is unbounded if
   - W is unbounded and $D \geq 1$;
   - $W \geq 2$ and D is unbounded

   TC is also **PSPACE**-hard in either of these cases

So, “TC is tractable iff C and W^D are bounded”

Overview

- Introduction
- Tree Languages
- Tree Transformations : XSLT
- The Typechecking Problem
- Tractable Deleting Transformations
- Tractable Copying Transformations
- Conclusion

When can we Copy?

DFAs in DTDs are too strong, consider

RE^+ expressions: concatenations of a and a^+

Example: $a^+bbac^+bb^+$

Theorem: $TC[\mathcal{T}_{d,uc}, DTD(RE^+)]$ is in **PTIME**

... but RE^+ is very limited

When can we Copy?

Consider the following extensions of RE^{+}:

- allow a and a^*;
- allow a and $a?$;
- allow a and $(a_1^+ + \cdots + a_n^+)$;
- allow a and $(a_1 \cdots a_n)^+$;
- allow a and $(a_1 + \cdots + a_n)^+$; and
- allow $(a_1 + \cdots + a_n)$ and a^+.

The inclusion problem is CONP-hard for all these extensions [M., Neven, Schwentick 04]

Overview

- Introduction
- Tree Languages
- Tree Transformations : XSLT
- The Typechecking Problem
- Tractable Deleting Transformations
- Tractable Copying Transformations
- Conclusion

Conclusion

We identified several interesting **PTIME**-fragments for TC:

$\mathcal{T}_{\text{trac}}$,	extended with XPath$\{\ell, /, *\}$,	w.r.t. DTD(DFA)
$\mathcal{T}_{nd,bc}$,	extended with DFA,	w.r.t. DTD(DFA)
$\mathcal{T}_{d,uc}$,		w.r.t. DTD(RE^{+})

Slightly extending these fragments results in **CONP**-hardness

The typeckecking algorithms can be adapted to generate counterexamples in case of negative output

Our results show when it is needed to search for incomplete typechecking algorithms