Zastosowania teorii automatów do przetwarzania dokumentów XML

Paweł Parys

Rozważany problem

Dane:

Dokument XML

Zapytanie XPath

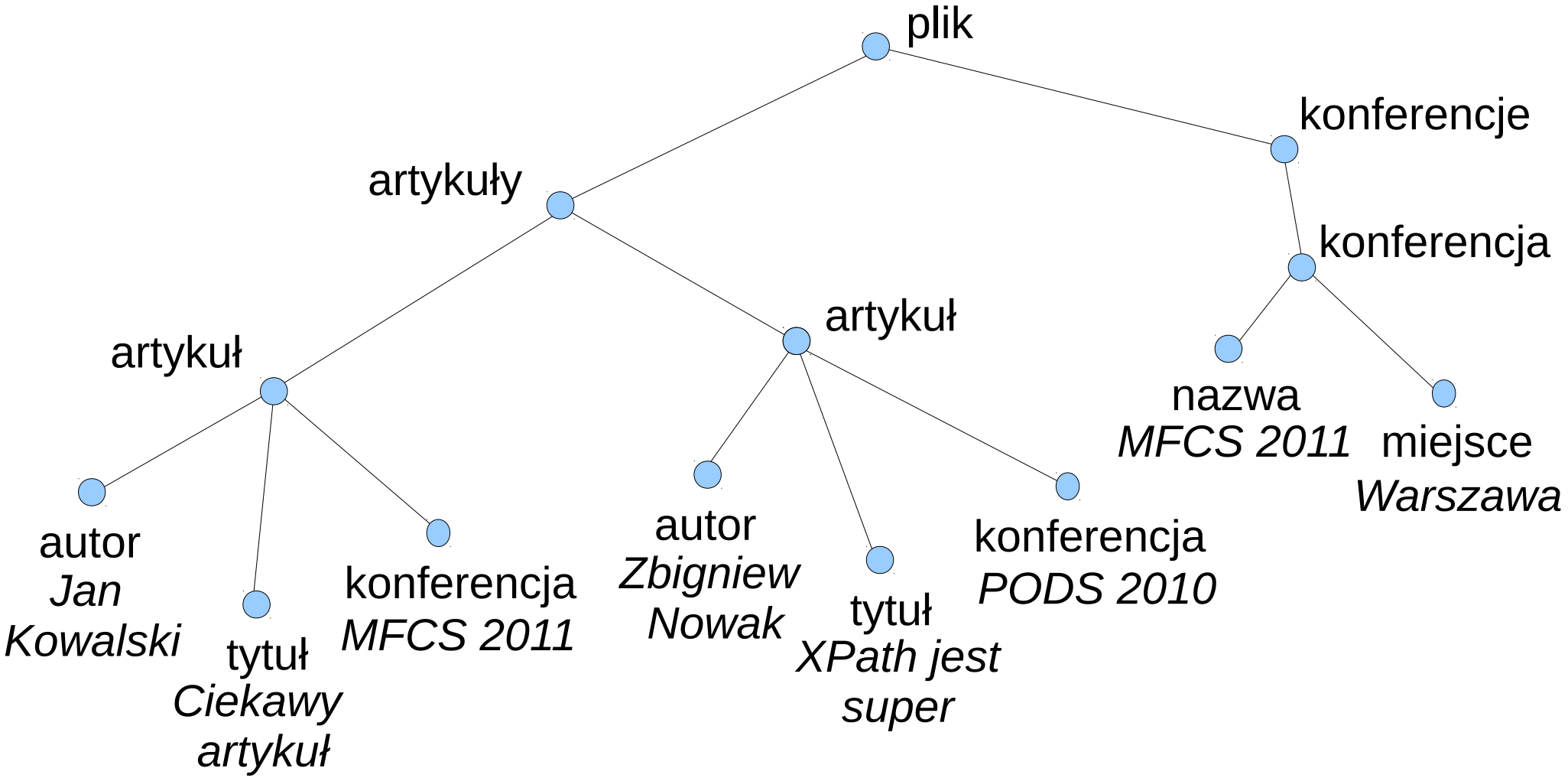
Wynik:

Wierzchołki dokumentu spełniające zapytanie

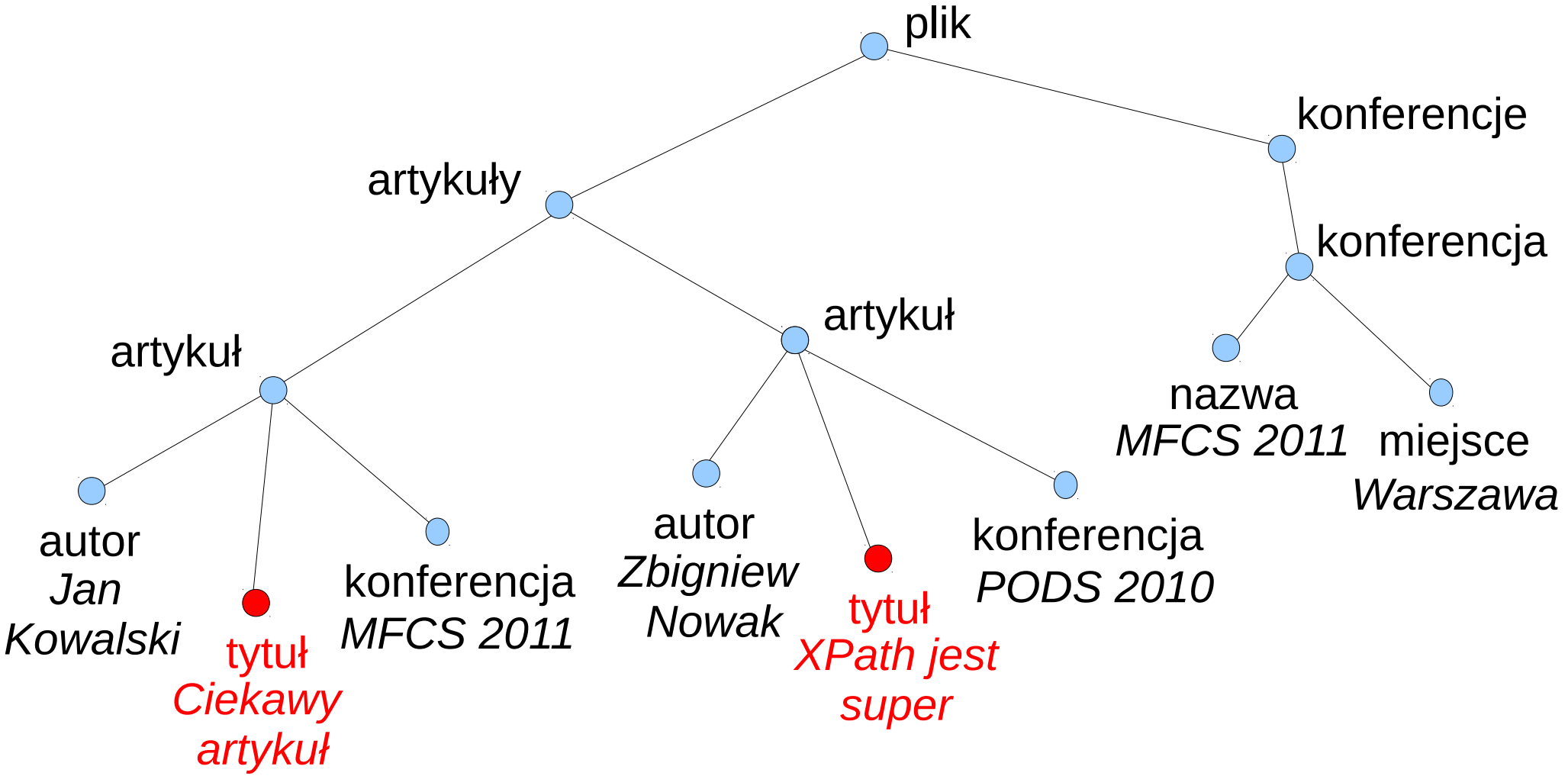
Przykład

```
<plik>
  <artykuly>
    <artykul>
      <autor>Jan Kowalski</autor>
      <tytul>Ciekawy artykuł</tytul>
      <konferencja>MFCS 2011</konferencja>
    </artykul>
    <artykul>
      <autor>Zbigniew Nowak</autor>
      <tytul>XPath jest super</tytul>
      <konferencja>PODS 2010</konferencja>
    </artykul>
  </artykuly>
  <konferencje>
    <konferencja>
      <nazwa>MFCS 2011</nazwa>
      <miejsce>Warszawa</miejsce>
    </konferencja>
  </konferencje>
</plik>
```

Przykład


plik
artykuły
konferencje
artykuł
konferencja
artykuł
nazwa
MFCS 2011
miejsce
Warszawa
autor
Jan
Kowalski
tytuł
Ciekawy
artykuł
konferencja
MFCS 2011
autor
Zbigniew
Nowak
tytuł
XPath jest
super
konferencja
PODS 2010

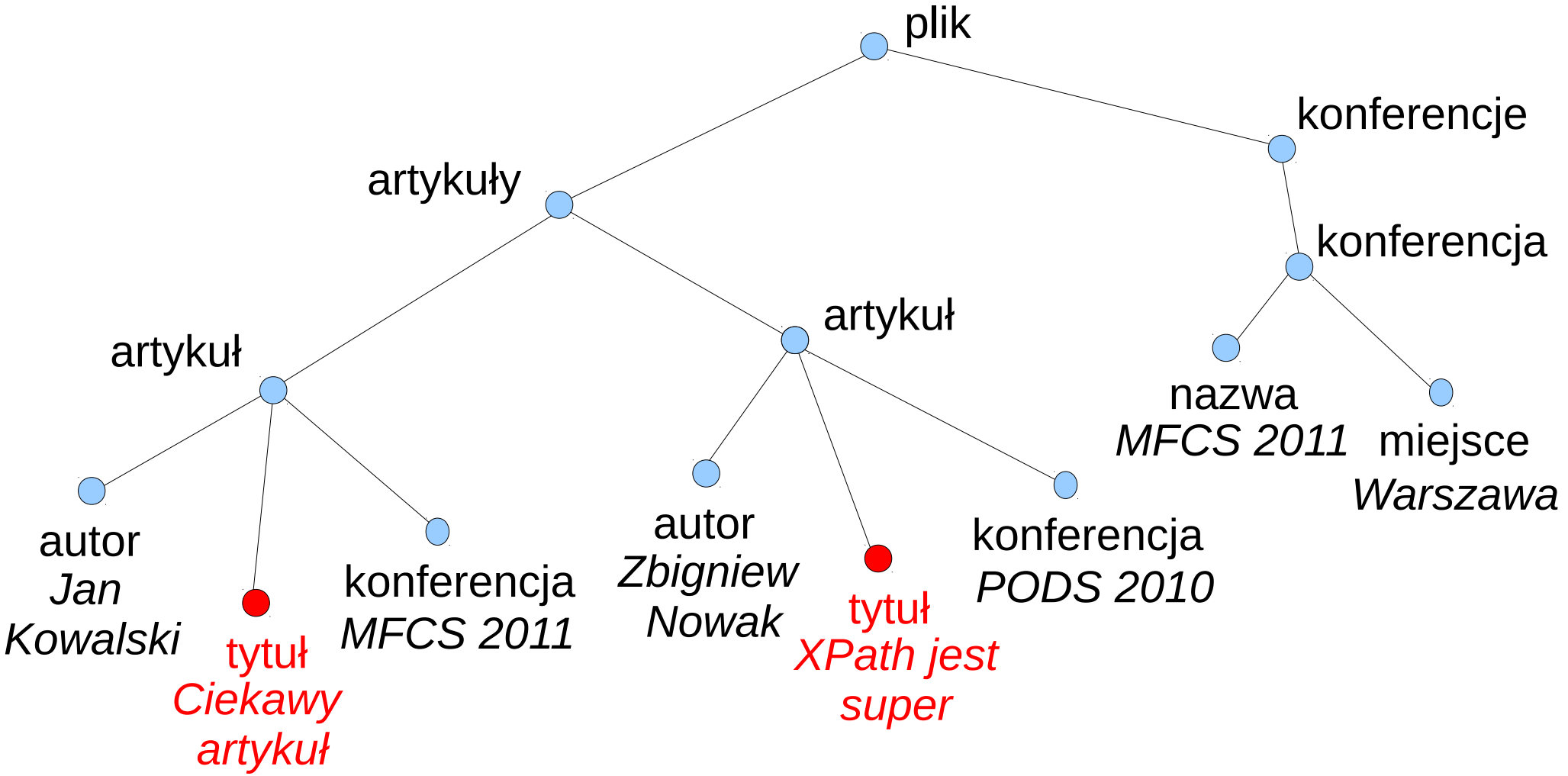
Przykład



Zapytanie:

plik/artykuły/artykuł/tytuł

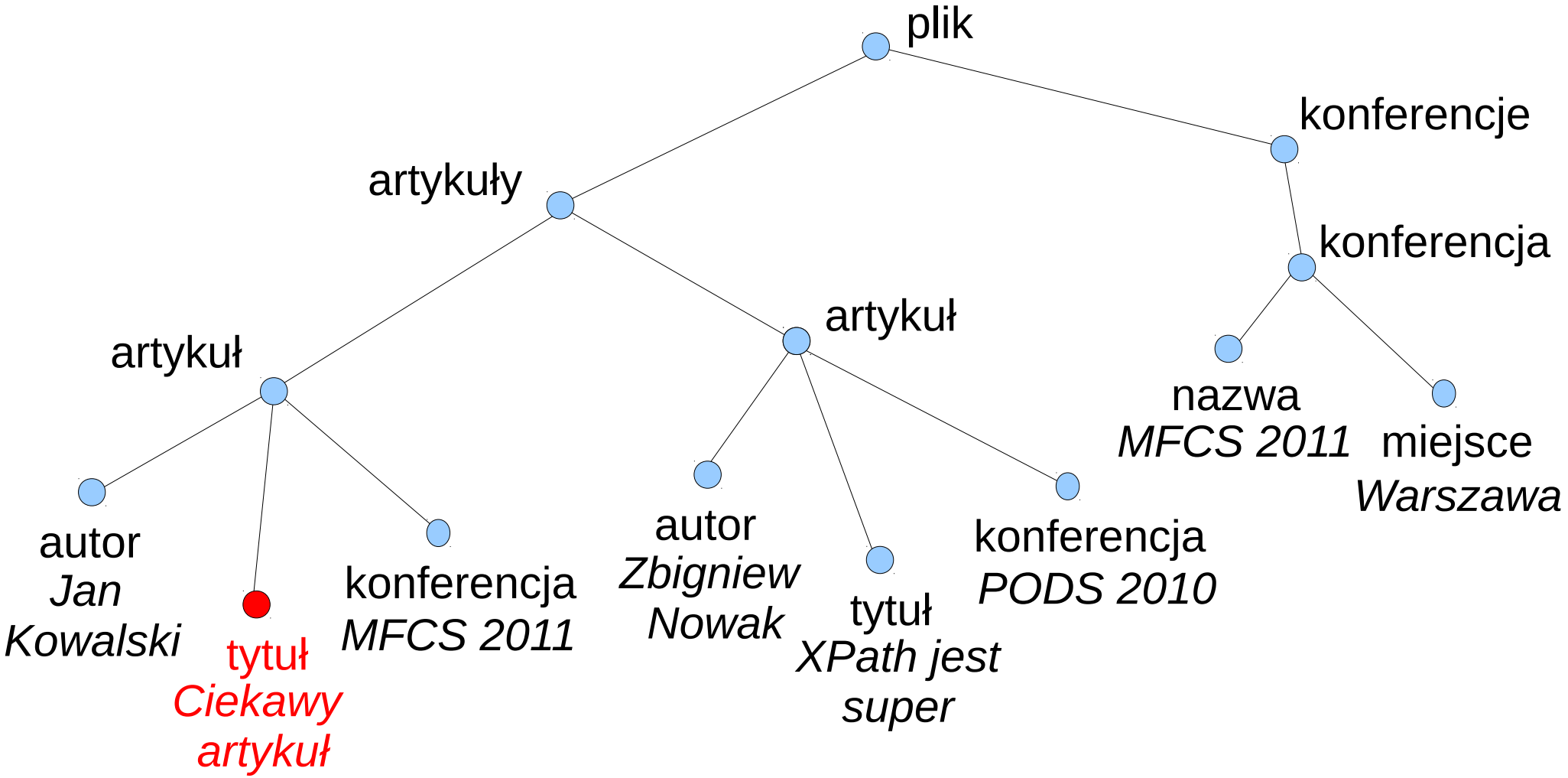
Przykład



Zapytanie:

plik/artykuły/artykuł/../artykuł/../artykuł/../artykuł/../artykuł/../artykuł/../artykuł/../artykuł/../artykuł/tytuł

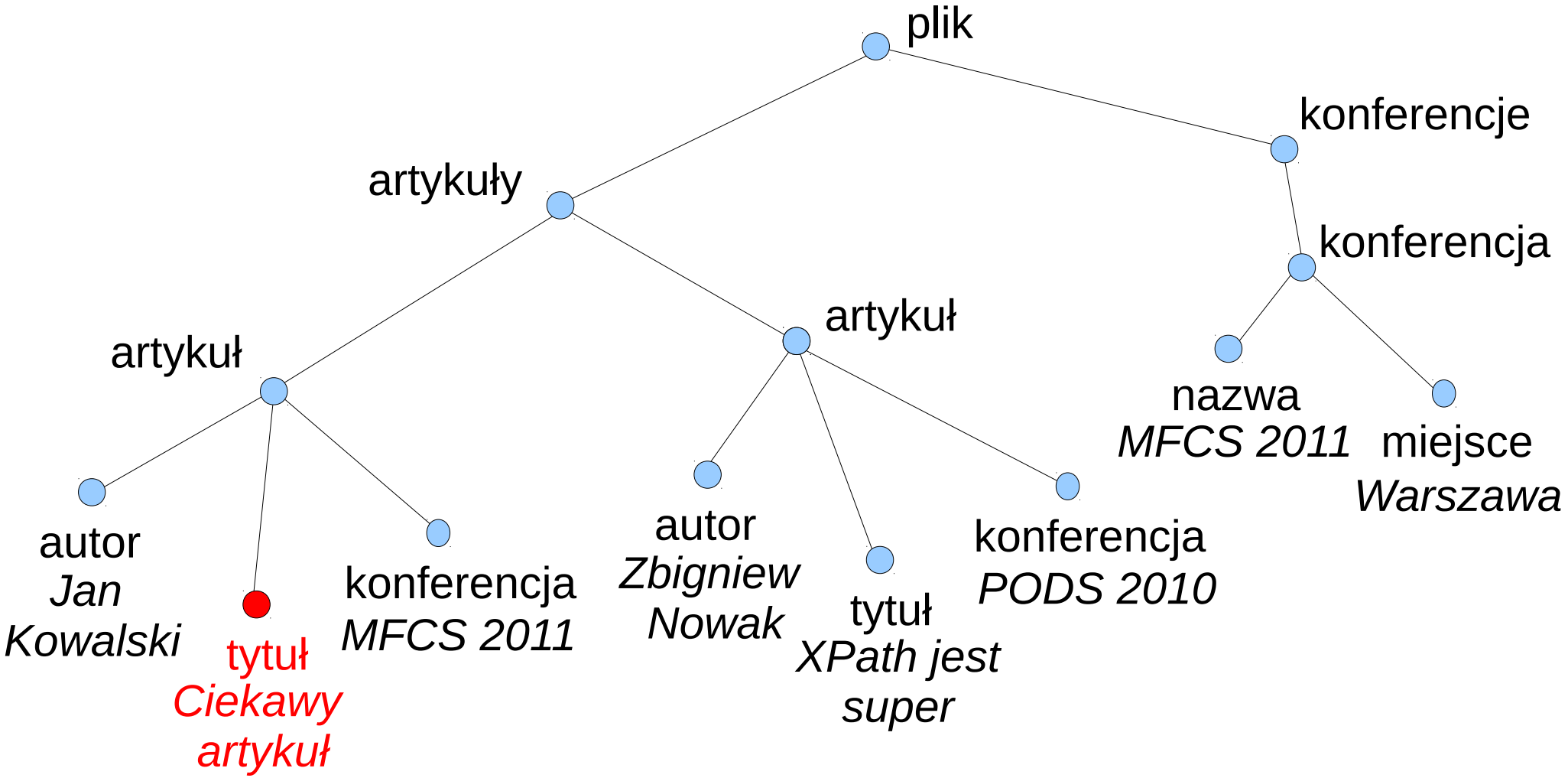
Przykład



Zapytanie:

plik/artykuły/artykuł[autor='Jan Kowalski']/tytuł

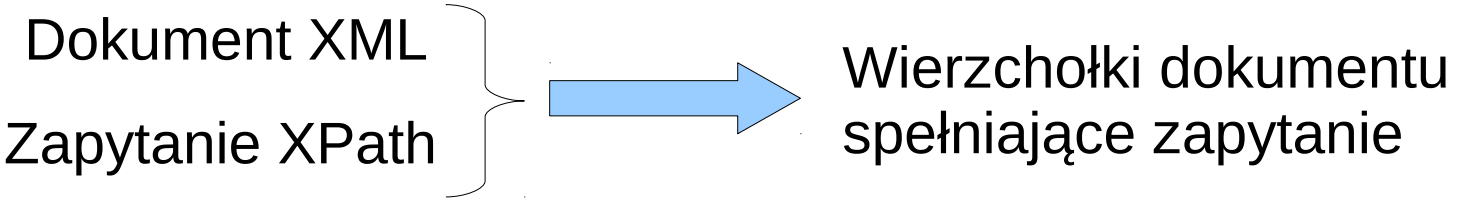
Przykład



Zapytanie:

plik/artykuły/artykuł[konferencja=
../../konferencje/konferencja[miejsce='Warszawa']/nazwa]/tytuł

Podsumowanie wyników



XPath bez danych

$O(D \cdot Q)$ - Gottlob, Koch, Pichler 2002

XPath z danymi (ale bez zliczania)

$O(D^2 \cdot Q)$ - Gottlob, Koch, Pichler 2002

$O(D \cdot Q^3)$ - ta rozprawa

gdzie: D - rozmiar dokumentu
Q - rozmiar zapytania

Podproblem:

Ustalmy język regularny L. Dane jest słowo $u = a_1 \ldots a_n$

Najpierw w czasie liniowym względem n możemy się przygotować

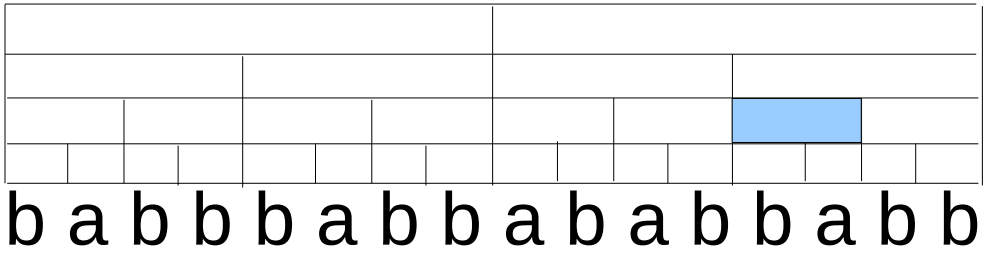
Następnie, chcemy w czasie stałym odpowiadać na zapytania:

czy $a_i \ldots a_j \in L$?

Podproblem:

Ustalmy język regularny L. Dane jest słowo $u = a_1 \dots a_n$

Przygotowanie: dziel i zwyciężaj



Dla każdego z podsłów zapamiętaj wszystkie możliwe przejścia automatu: pary stanów p, q takie, że

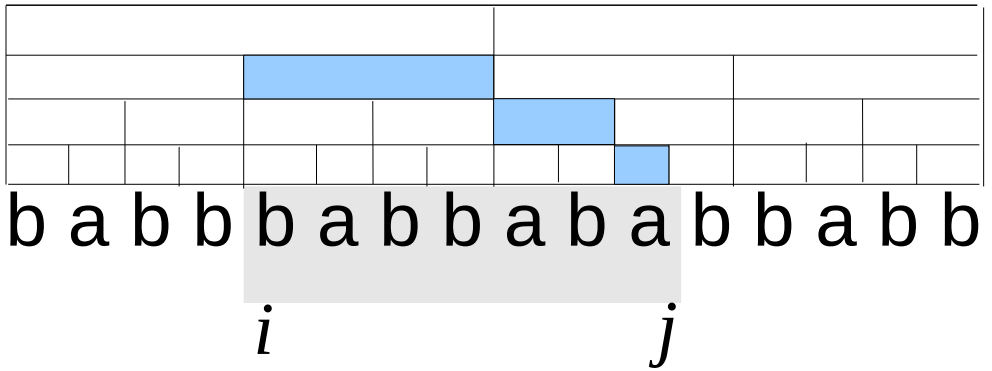
$$p \xrightarrow{a_i \dots a_j} q$$

czas: $O(n)$

Podproblem:

Ustalmy język regularny L. Dane jest słowo $u = a_1 \ldots a_n$

Dane: i, j



Czy $a_i \ldots a_j \in L$?

Wystarczy złożyć zapamiętane przejścia!

czas: $O(\log n)$

Podproblem:

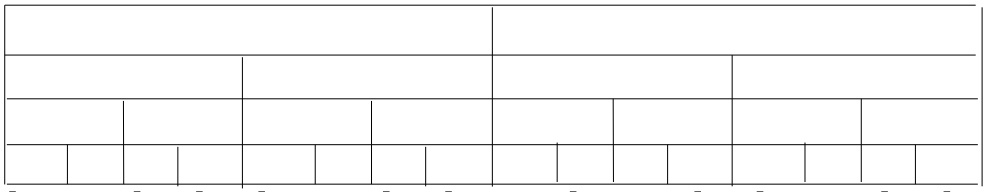
Użyte narzędzie: twierdzenie Simon'a

(I. Simon, Factorization forests of finite height, 1990)

Podproblem:

Ustalmy język regularny L. Dane jest słowo $u = a_1 \dots a_n$

W dekompozycji „logarytmicznej” dzielimy zawsze na 2 cześci



Aby uzyskać stałą wysokość drzewa podziału,
trzeba dozwolić podziały na dowolnie wiele części
- ale wtedy wszystkie części muszą być bardzo podobne.

Dekompozycja Simona:

Każde słowo u w drzewie podziału dzielimy
- na 2 (dowolne) części $u = u_1 u_2$, lub
- na dowolną liczbę części $u = u_1 \ldots u_k$, gdzie wszystkie $u_i \ldots u_j$ są równoważne.

Tw. Simona
Dla każdego słowa istnieje takie drzewo podziału tej samej głębokości.

u oraz v są równoważne, jeśli dla dowolnych słów w_1, w_2 zachodzi

$$w_1 u w_2 \in L \Leftrightarrow w_1 v w_2 \in L$$

Dekompozycja Simona:

Każde słowo u w drzewie podziału dzielimy
- na 2 (dowolne) części $u = u_1 u_2$, lub
- na dowolną liczbę części $u = u_1 \dots u_k$, gdzie wszystkie $u_i \dots u_j$ są równoważne.

Przykład

L=(a+b)*b

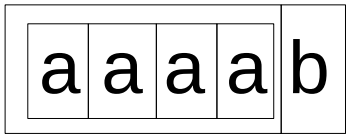
a a a a b

Dekompozycja Simona:

Każde słowo u w drzewie podziału dzielimy
- na 2 (dowolne) części $u = u_1 u_2$, lub
- na dowolną liczbę części $u = u_1 \dots u_k$, gdzie wszystkie $u_i \dots u_j$ są równoważne.

Przykład

L=(a+b)*b

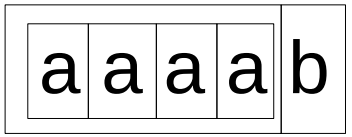


Dekompozycja Simona:

Każde słowo u w drzewie podziału dzielimy
- na 2 (dowolne) części $u = u_1 u_2$, lub
- na dowolną liczbę części $u = u_1 \dots u_k$, gdzie wszystkie $u_i \dots u_j$ są równoważne.

Przykład

L=(a+b)*b



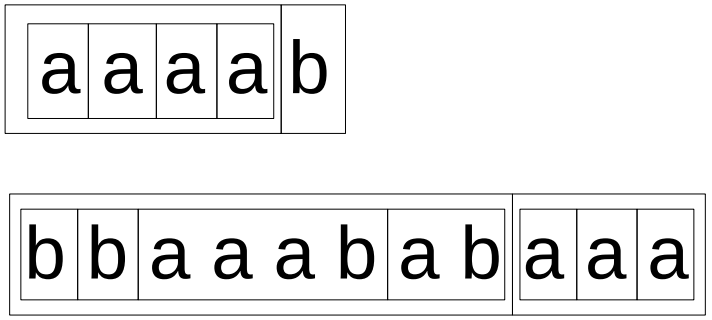
b b a a a b a b a a a

Dekompozycja Simona:

Każde słowo u w drzewie podziału dzielimy
- na 2 (dowolne) części $u = u_1 u_2$, lub
- na dowolną liczbę części $u = u_1 \dots u_k$, gdzie wszystkie $u_i \dots u_j$ są równoważne.

Przykład

L=(a+b)*b



Podproblem:

Ustalmy język regularny L. Dane jest słowo $u = a_1 \ldots a_n$

Przygotowanie:
- liczymy dekompozycję Simona
- dla każdego podsłowa w demopozycji liczymy przejścia automatu

czas: $O(n)$

Czy $a_i \ldots a_j \in L$?

- Wystarczy złożyć zapamiętane przejścia

czas: $O(1)$

Podproblem:

Ustalmy język regularny L. Dane jest słowo $u = a_1 \dots a_n$

Przygo
- liczyr
- dla k
  liczyr

czas:

Czy c

- Wys

czas:

Zależność od języka L

Wysokość drzewa dekompozycji jest proporcjonalna do liczby klas abstrakcji, która jest wykładnicza względem rozmiaru automatu.

Ale drzewo ma $O(n)$ wierzchołków.

<u>Nowy wynik:</u>
jak radzić sobie z tą dekompozycją w czasie wielomianowym względem rozmiaru automatu

Dziękuję za uwagę