Proof Systems for Data Queries

http://projects.lsv.ens-cachan.fr/prodaq

Mid-project Review

October 5th, 2016

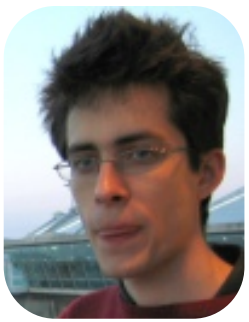
Consortium

Partner

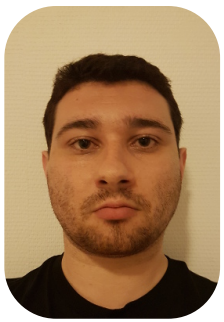
Laboratoire Spécification et Vérification
École Normale Supérieure Cachan

Current Participants

David Baelde



Anthony Lick



Sylvain Schmitz

Life of the Project

May–Jul. 2015 MSc. internship of Simon Lunel

Apr.–Jul. 2016 MSc. internship of Anthony Lick

Jun.–Jul. 2016 BSc. internship of Manoj Kilaru

Sep. 2016–Aug. 2019 PhD of Anthony Lick

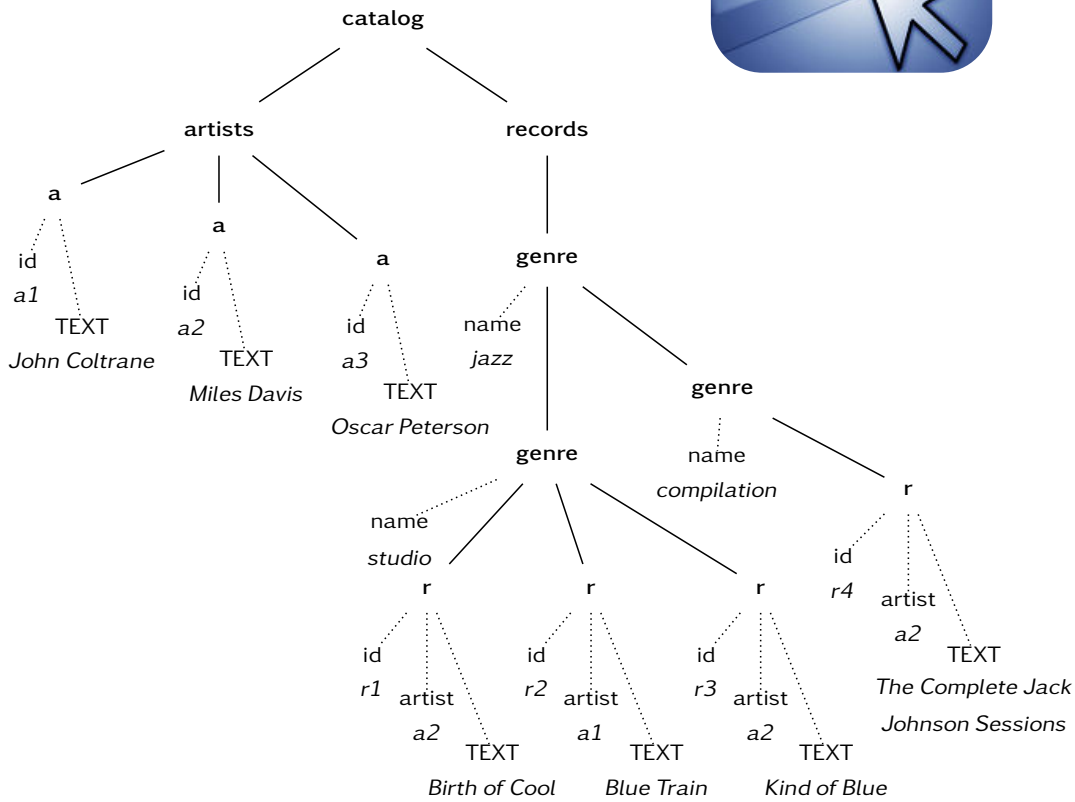
General Context

- XML format
- data trees
- XPath queries
- Satisfiability


```
<catalog>
  <artists>
    <a id="a1">John Coltrane</a>
    <a id="a2">Miles Davis</a>
    <a id="a3">Oscar Peterson</a>
  </artists>
  <records>
    <genre name="jazz">
      <genre name="studio">
        <r id="r1" artist="a2">Birth of Cool</r>
        <r id="r2" artist="a1">Blue Train</r>
        <r id="r3" artist="a2">Kind of Blue</r>
      </genre>
      <genre name="compilation">
        <r id="r4" artist="a2">The Complete Jack
         Johnson Sessions</r>
      </genre>
    </genre>
  </records>
</catalog>
```

General Context

- XML format
- data trees
- XPath queries
- Satisfiability



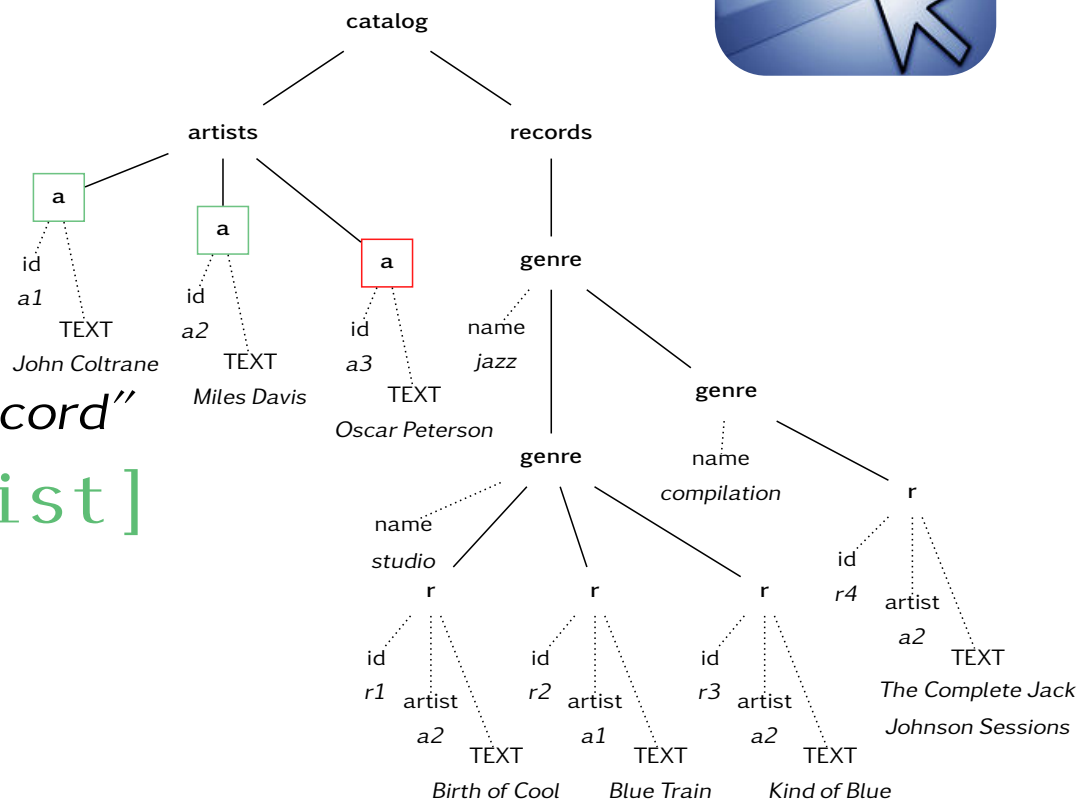
General Context

- XML format
- data trees
- XPath queries

"artists with at least one record"

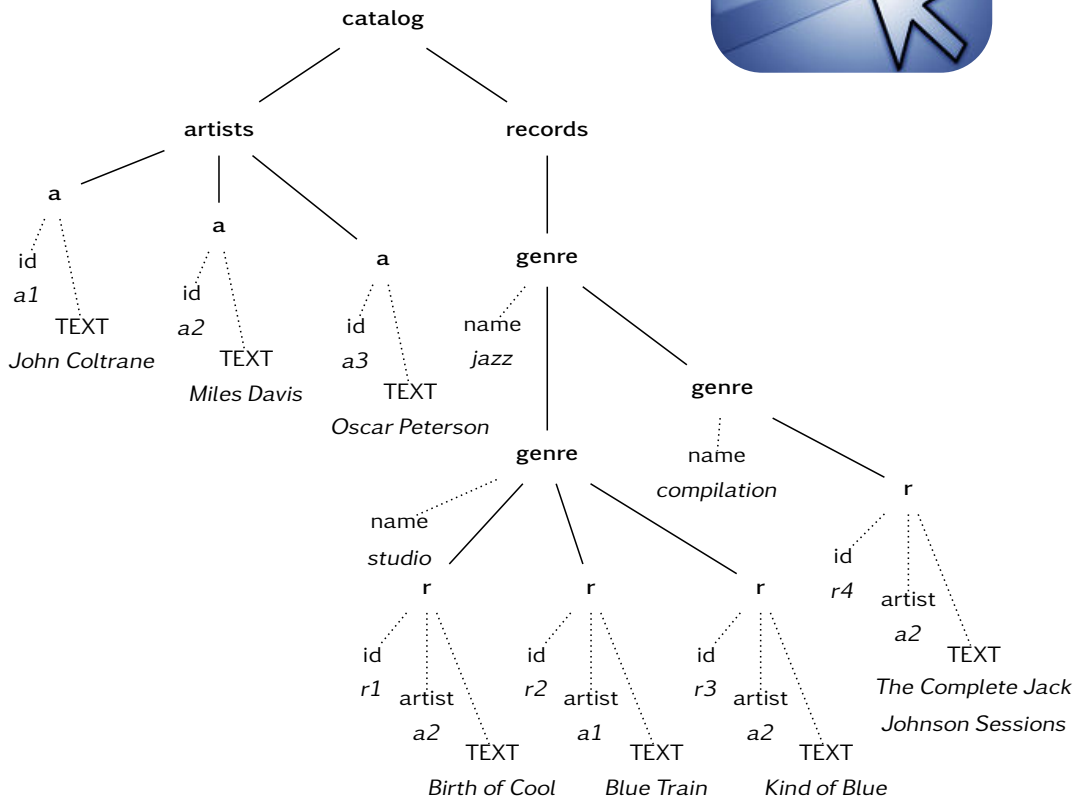
```
//a[@id=//r/@artist]
```

- Satisfiability

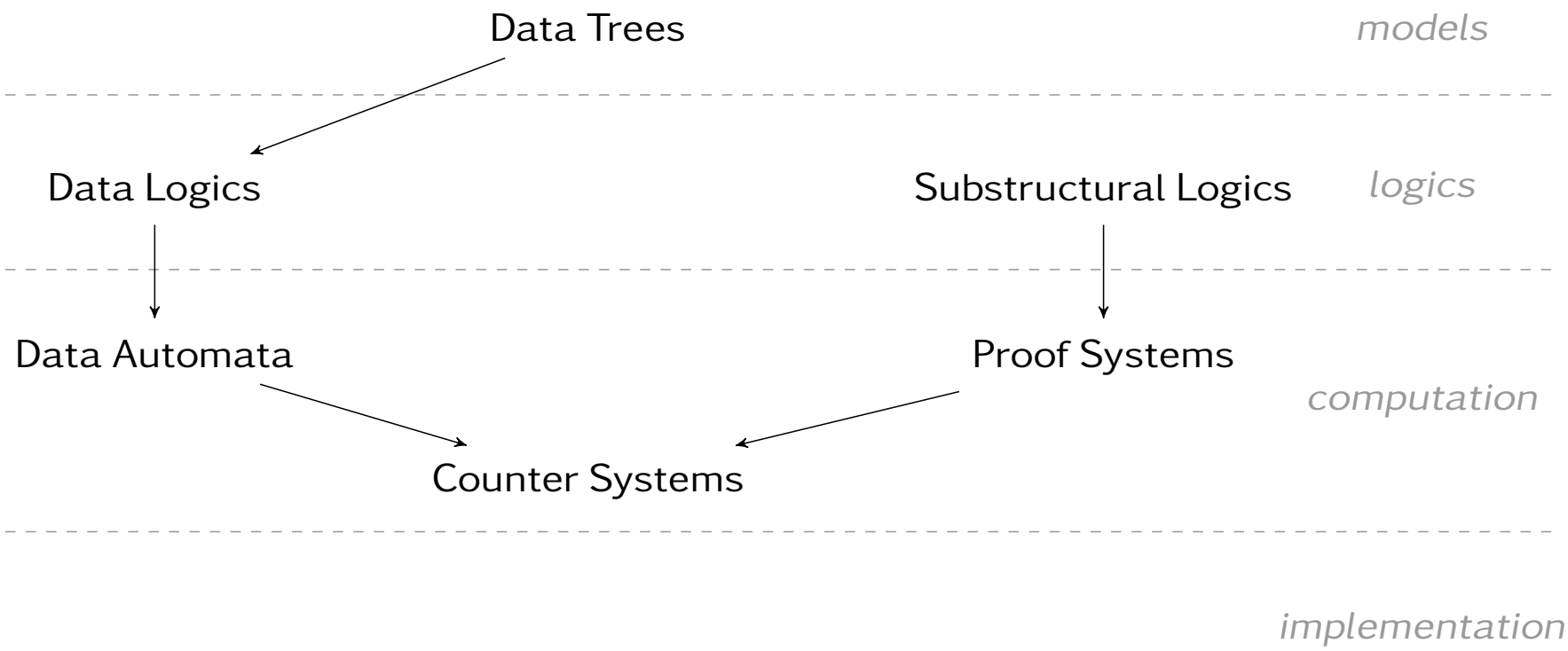


General Context

- XML format
- data trees
- XPath queries
- Satisfiability
  - optimization
  - verification (access control)

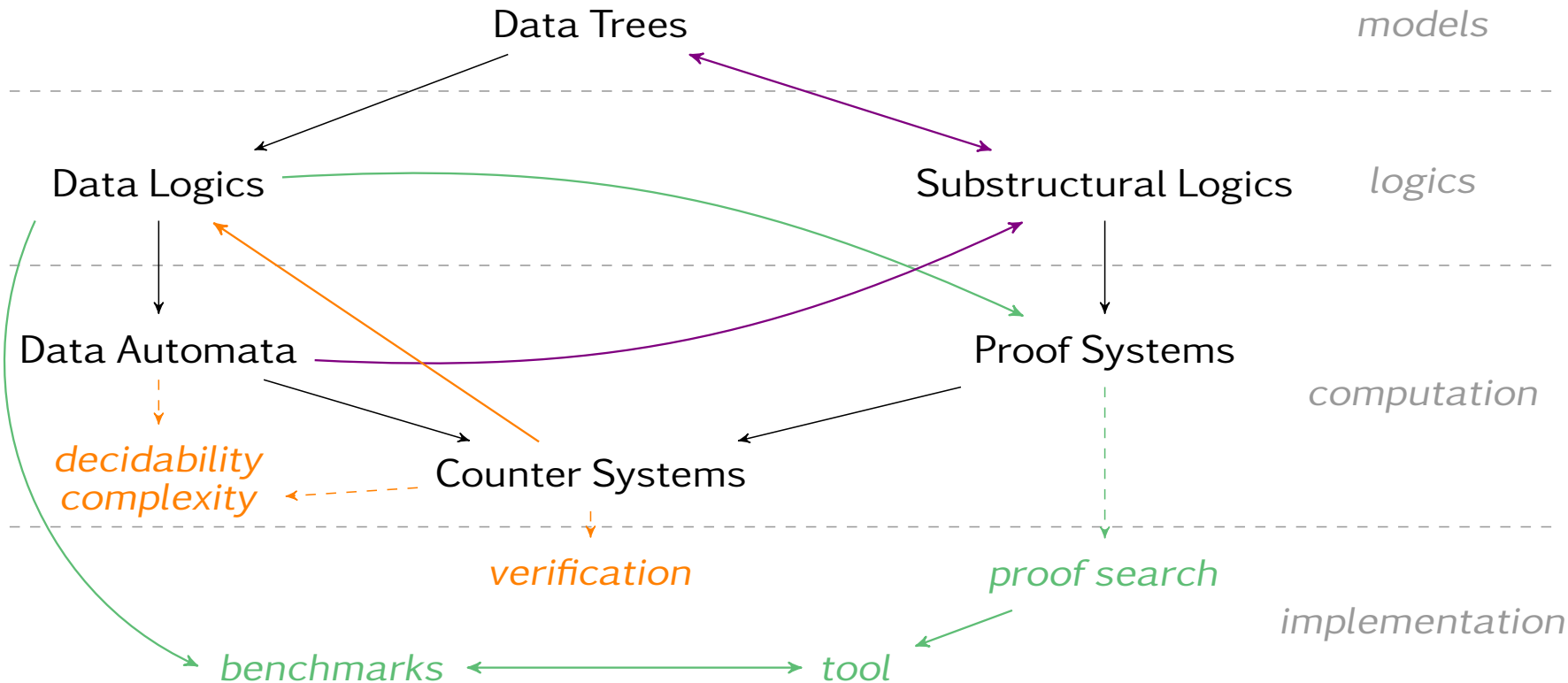


Objectives: Bridges and Tools



Task A Task B Task C

Objectives: Bridges and Tools



Task A Task B Task C

A: Proof Systems for Data Logics

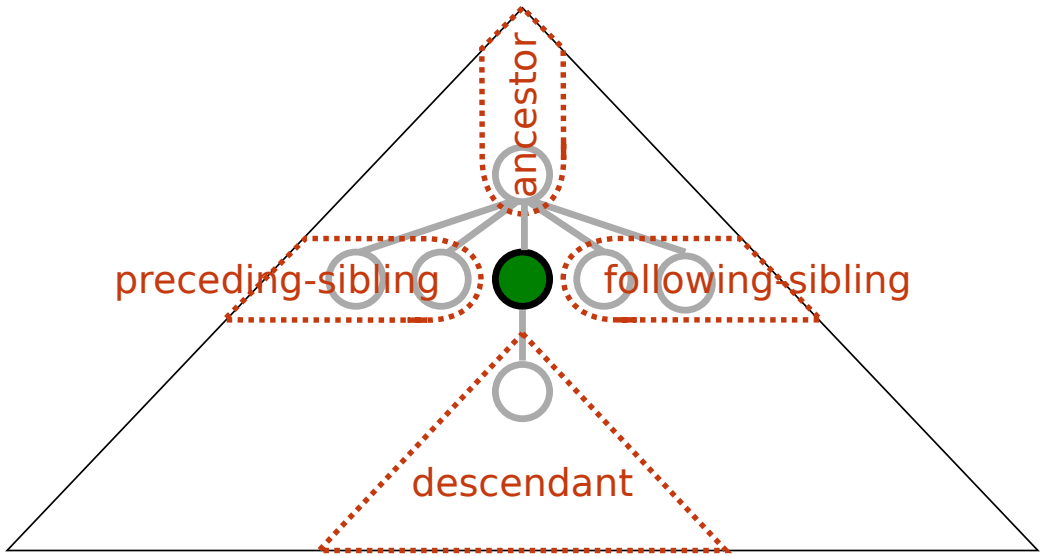
XPath in the literature: **CoreDataXPath**

[Benedikt & Koch, Bojánczyk et al.]

- single attribute: $@d$
- full joins after paths π, π':

$$\pi/@d \; = \; \pi'/@d \qquad \pi/@d \; != \; \pi'/@d$$

- navigation in paths π along several axes, e.g.:



Reasoning about CoreDataXPath

- undecidable satisfiability
- decidable fragments (subsets of axes)
  [Figueira]
  - different proofs for different fragments
  - model-theoretic approaches:
    - bound size of models if they exist
    - combinatorial algorithms: enumerate potential models
- in PRODAQ: proof systems
  - concrete proof search algorithms
  - witnesses of unsatisfiability (proofs)

Reasoning about CoreDataXPath

- undecidable satisfiability
- decidable fragments (subsets of axes)
  [Figueira]
  - different proofs for different fragments
  - model-theoretic approaches:
    - bound size of models if they exist
    - combinatorial algorithms: enumerate potential models
- in PRODAQ: proof systems
  - concrete proof search algorithms
  - witnesses of unsatisfiability (proofs)

Reasoning about CoreDataXPath

- undecidable satisfiability
- decidable fragments (subsets of axes)
  [Figueira]
  - different proofs for different fragments
  - model-theoretic approaches:
    - bound size of models if they exist
    - combinatorial algorithms: enumerate potential models
- in PRODAQ: proof systems
  - concrete proof search algorithms
  - witnesses of unsatisfiability (proofs)

Modal Logic on Data Trees

Publications: [MSc'15, CSL'16]; People: David Baelde, Simon Lunel, Sylvain Schmitz

- "descendant" axis of **CoreDataXPath: DataKL**
  - $\Diamond_{=}\varphi$ "φ holds in a descendant with the same data"

    XPath syntax: `@d = descendant::*[φ]/@d`
  - $\Diamond_{\neq}\varphi$ "φ holds in a descendant with a different data"

    XPath syntax: `@d != descendant::*[φ]/@d`
- sound & complete hypersequent calculus
- optimal PSpace proof search algorithm

  no penalty from the use of proof systems!

Modal Logic on Data Trees

Publications: [MSc'15, CSL'16]; People: David Baelde, Simon Lunel, Sylvain Schmitz

- "descendant" axis of **CoreDataXPath: DataKL**
  - $\Diamond_{=}\varphi$ "φ holds in a descendant with the same data"

    XPath syntax: $@d \;=\; descendant::*[\varphi]/@d$
  - $\Diamond_{\neq}\varphi$ "φ holds in a descendant with a different data"

    XPath syntax: $@d \;!=\; descendant::*[\varphi]/@d$
- sound & complete hypersequent calculus
- optimal PSpace proof search algorithm

  no penalty from the use of proof systems!

Modal Logic on Data Trees

Publications: [MSc'15, CSL'16]; People: David Baelde, Simon Lunel, Sylvain Schmitz

- "descendant" axis of **CoreDataXPath: DataKL**
  - $\Diamond_{=}\varphi$ "φ holds in a descendant with the same data"

    XPath syntax: $@d \;=\; descendant::*[\varphi]/@d$
  - $\Diamond_{\neq}\varphi$ "φ holds in a descendant with a different data"

    XPath syntax: $@d \;!=\; descendant::*[\varphi]/@d$
- sound & complete hypersequent calculus
- optimal PSpace proof search algorithm

  no penalty from the use of proof systems!

Modal Logic on Data Trees

Publications: [MSc'15, CSL'16]; People: David Baelde, Simon Lunel, Sylvain Schmitz

- "descendant" axis of **CoreDataXPath: DataKL**
  - $\Diamond_{=}\varphi$ "φ holds in a descendant with the same data"

    XPath syntax: $@d \;=\; descendant::*[\varphi]/@d$
  - $\Diamond_{\neq}\varphi$ "φ holds in a descendant with a different data"

    XPath syntax: $@d \;!=\; descendant::*[\varphi]/@d$
- sound & complete hypersequent calculus
- optimal PSpace proof search algorithm
  no penalty from the use of proof systems!

Modal Logic on Words

Publications: [MSc'16]; People: David Baelde, Anthony Lick, Sylvain Schmitz

- trichotomous (non-branching) XPath axes: "ancestor", "following-sibling", "preceding-sibling"
- "following-sibling" fragment of **CoreXPath**: **KL.3**
- sound & complete sequent calculus
- optimal coNP proof search

Modal Logic on Words

Publications: [MSc'16]; People: David Baelde, Anthony Lick, Sylvain Schmitz

- trichotomous (non-branching) XPath axes: "ancestor", "following-sibling", "preceding-sibling"
- "following-sibling" fragment of **CoreXPath**: **KL.3**
- sound & complete sequent calculus
- optimal coNP proof search

Modal Logic on Words

Publications: [MSc'16]; People: David Baelde, Anthony Lick, Sylvain Schmitz

- inverse XPath axes: "descendant" & "ancestor", "following-sibling" & "preceding-sibling"
- "following-sibling" & "preceding-sibling" fragment of **CoreXPath: $\mathbf{K_t4.3}$**
- sound & complete hypersequent calculus
- optimal coNP proof search
- need to force finite words: $\mathbf{K_tL.3}$
- need to extend to $\mathbf{DataK_tL.3}$

Modal Logic on Words

Publications: [MSc'16]; People: David Baelde, Anthony Lick, Sylvain Schmitz

- inverse XPath axes: "descendant" & "ancestor", "following-sibling" & "preceding-sibling"
- "following-sibling" & "preceding-sibling" fragment of **CoreXPath: $\mathbf{K_t4.3}$**
- sound & complete hypersequent calculus
- optimal coNP proof search
- need to force finite words: $\mathsf{K_tL.3}$
- need to extend to $\mathsf{DataK_tL.3}$

Modal Logic on Data Words

Publications: [MSc'16]; People: David Baelde, Anthony Lick, Sylvain Schmitz

- inverse XPath axes: "descendant" & "ancestor", "following-sibling" & "preceding-sibling"
- "following-sibling" & "preceding-sibling" fragment of **CoreXPath**: $\mathbf{K_t4.3}$
- sound & complete hypersequent calculus
- optimal coNP proof search
- need to force finite words: $\mathbf{K_tL.3}$
- need to extend to $\mathbf{DataK_tL.3}$

XPath Benchmark

Work in progress; People: David Baelde, Anthony Lick, Sylvain Schmitz

- actual XPath practice ≠ **CoreDataXPath**
- joins are local:

```
<query file="/usr/share/xml/docbook/stylesheet/.../blocks2dbk.xsl"
  line="766" type="test"
  content="@rnd:style = preceding-sibling::dbk:emphasis/@rnd:style"/>
```

- nominal/hybrid constructs:

```
<query file="/usr/share/xml/docbook/stylesheet/.../blocks2dbk.xsl"
  line="236" type="select"
  content="following-sibling::dbk:para[@rnd:style = current()/@rnd:style]"/>

<query file="/usr/lib/python2.7/.../iso-schematron-xslt1/iso_dsdl_include.xsl"
  line="975" type="select"
  content="//*[@xml:id = current()/@xpointer] | id(@xpointer)"/>
```

A: Outlook

- Enrich our hypersequent calculus
  - more axes
  - nested paths
  - hybrid constructs like `id()` and `current()`
- Compile an XPath benchmark
- Implement a proof search engine

B: Data Models for Substructural Logics

Separation logic:
[O'Hearn, Reynolds, et al.]

- $M \models \varphi_1 * \varphi_2$ if $M = M_1 * M_2$ with $\forall i. M_i \models \varphi_i$
- versatile: heaps, concurrent data structures, etc.
- in PRODAQ: data models
  - separation should be compatible with data values
  - a new viewpoint on data logics

B: Data Models for Substructural Logics

Separation logic:
[O'Hearn, Reynolds, et al.]

- $M \models \varphi_1 * \varphi_2$ if $M = M_1 * M_2$ with $\forall i. M_i \models \varphi_i$
- versatile: heaps, concurrent data structures, etc.
- in PRODAQ: data models
  - separation should be compatible with data values
  - a new viewpoint on data logics

Separation Logic on Data Words

Work in progress; People: Manoj Kilaru, Étienne Lozes, Sylvain Schmitz

- separating shuffle $\circledast$ over data words:
  - $\begin{pmatrix}a\\1\end{pmatrix}\begin{pmatrix}b\\2\end{pmatrix}\begin{pmatrix}c\\1\end{pmatrix}\begin{pmatrix}d\\3\end{pmatrix} \in \begin{pmatrix}a\\1\end{pmatrix}\begin{pmatrix}c\\1\end{pmatrix} \circledast \begin{pmatrix}b\\2\end{pmatrix}\begin{pmatrix}d\\3\end{pmatrix}$
  - $\begin{pmatrix}a\\1\end{pmatrix}\begin{pmatrix}b\\2\end{pmatrix}\begin{pmatrix}c\\1\end{pmatrix}\begin{pmatrix}d\\3\end{pmatrix} \notin \begin{pmatrix}a\\1\end{pmatrix}\begin{pmatrix}d\\3\end{pmatrix} \circledast \begin{pmatrix}b\\2\end{pmatrix}\begin{pmatrix}c\\1\end{pmatrix} = \emptyset$
- separating shuffle regular expressions (SSRE) on top of regular expressions E:

$$\varphi ::= E \mid \varphi \vee \varphi \mid \neg\varphi \mid \varphi \cdot \varphi \mid \varphi \circledast \varphi$$

Separation Logic on Data Words

Work in progress; People: Manoj Kilaru, Étienne Lozes, Sylvain Schmitz

- separating shuffle $\circledast$ over data words:
  - $\begin{pmatrix} a \\ 1 \end{pmatrix}\begin{pmatrix} b \\ 2 \end{pmatrix}\begin{pmatrix} c \\ 1 \end{pmatrix}\begin{pmatrix} d \\ 3 \end{pmatrix} \in \begin{pmatrix} a \\ 1 \end{pmatrix}\begin{pmatrix} c \\ 1 \end{pmatrix} \circledast \begin{pmatrix} b \\ 2 \end{pmatrix}\begin{pmatrix} d \\ 3 \end{pmatrix}$
  - $\begin{pmatrix} a \\ 1 \end{pmatrix}\begin{pmatrix} b \\ 2 \end{pmatrix}\begin{pmatrix} c \\ 1 \end{pmatrix}\begin{pmatrix} d \\ 3 \end{pmatrix} \notin \begin{pmatrix} a \\ 1 \end{pmatrix}\begin{pmatrix} d \\ 3 \end{pmatrix} \circledast \begin{pmatrix} b \\ 2 \end{pmatrix}\begin{pmatrix} c \\ 1 \end{pmatrix} = \emptyset$
- separating shuffle regular expressions (SSRE) on top of regular expressions E:

$$\varphi ::= E \mid \varphi \vee \varphi \mid \neg\varphi \mid \varphi \cdot \varphi \mid \varphi \circledast \varphi$$

B: Outlook

- undecidable emptiness of SSRE
- extension with homomorphisms strictly contains EMSO^2
  [Bojánczyk et al.]
- negation-free SSRE strict fragment of EMSO^2, thus decidable emptiness
- is there a fragment equivalent to FO^2?
  to EMSO^2?

B: Outlook

- undecidable emptiness of SSRE
- extension with homomorphisms strictly contains EMSO^2
  [Bojánczyk et al.]
- negation-free SSRE strict fragment of EMSO^2, thus decidable emptiness
- is there a fragment equivalent to FO^2? to EMSO^2?

B: Outlook

- undecidable emptiness of SSRE
- extension with homomorphisms strictly contains EMSO^2
  [Bojánczyk et al.]
- negation-free SSRE strict fragment of EMSO^2, thus decidable emptiness
- is there a fragment equivalent to FO^2?
  to EMSO^2?

C: Counter Systems and Data

- data logics restricted to (dis)equality: counting
- counter systems: operational model
- complexity & algorithms

Complexity in Petri Nets

Publications: [LICS'15]; People: Jérôme Leroux, Sylvain Schmitz

- reachability decidable
  [Mayr, Kosaraju, Lambert]
- related to numerous formalisms on data words: EMSO^2, FO^2, modal-μ-calculi, data automata, class memory automata, etc.
- also related to many problems in other fields (process algebra, verification of concurrent systems, ...)
- first known upper bound: $\mathbf{F}_{\omega^3}$
- 35 years old major open problem

Complexity in Petri Nets

Publications: [LICS'15]; People: Jérôme Leroux, Sylvain Schmitz

- reachability decidable
  [Mayr, Kosaraju, Lambert]
- related to numerous formalisms on data words: EMSO^2, FO^2, modal-μ-calculi, data automata, class memory automata, etc.
- also related to many problems in other fields (process algebra, verification of concurrent systems, ...)
- first known upper bound: $\mathbf{F}_{\omega^3}$
- 35 years old major open problem

Complexity in Petri Nets

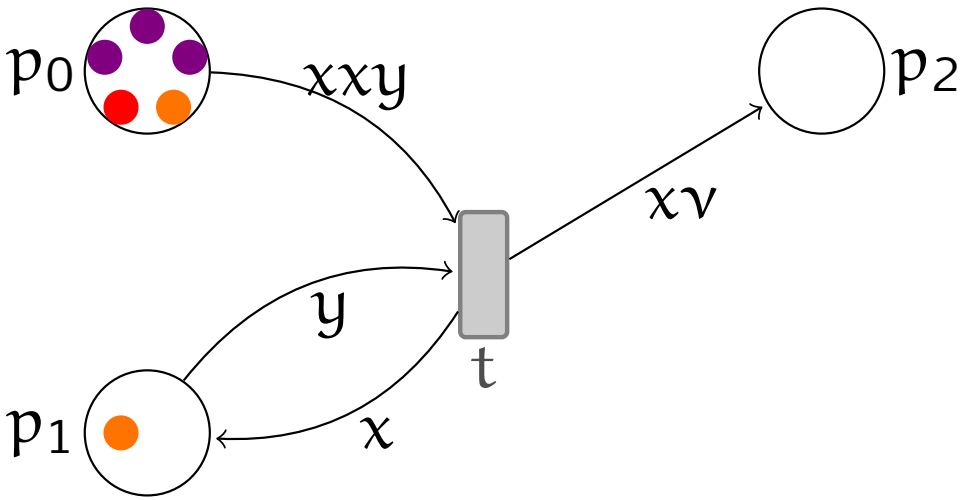
Publications: [LICS'15]; People: Jérôme Leroux, Sylvain Schmitz

- reachability decidable
  [Mayr, Kosaraju, Lambert]
- related to numerous formalisms on data words: EMSO^2, FO^2, modal-μ-calculi, data automata, class memory automata, etc.
- also related to many problems in other fields (process algebra, verification of concurrent systems, ...)
- first known upper bound: $\mathbf{F}_{\omega^3}$
- 35 years old major open problem

Petri Nets with Data

Publications: [FoSSaCS'16, LICS'16]; People: Piotr Hofman, Sławomir Lasota, Ranko Lazić, Jérôme Leroux, Sylvain Schmitz, Patrick Totzke

- tokens carry data

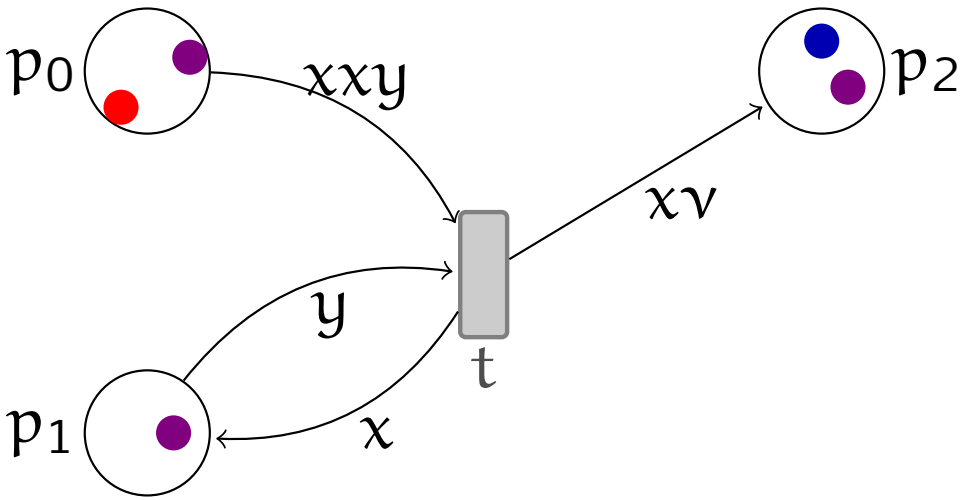


- ν-Petri nets: enforce fresh data
  - $\mathbf{F}_{\omega.2}$-complete safety verification
- data Petri nets: data might be non-fresh
  - decidable boundedness problems (in $\mathbf{F}_{\omega^\omega}$)

Petri Nets with Data

Publications: [FoSSaCS'16, LICS'16]; People: Piotr Hofman, Sławomir Lasota, Ranko Lazić, Jérôme Leroux, Sylvain Schmitz, Patrick Totzke

- tokens carry data

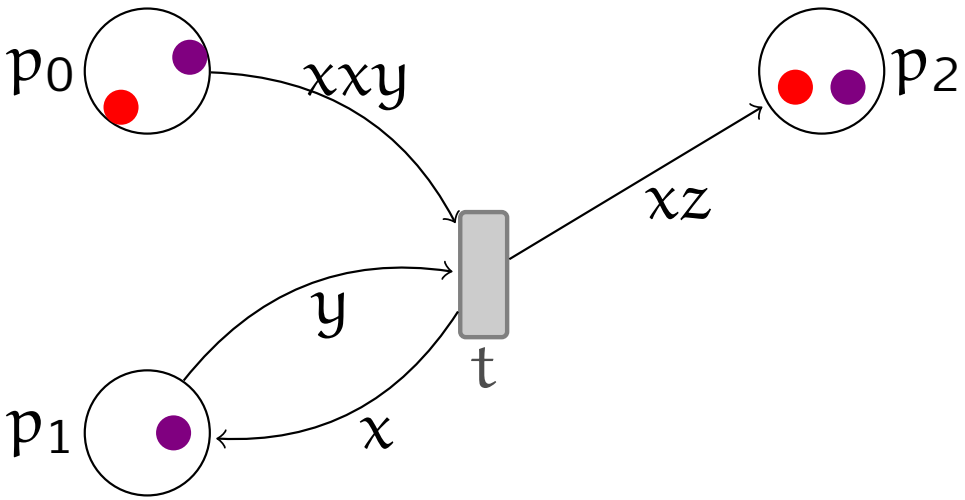


- ν-Petri nets: enforce fresh data
  - $\mathbf{F}_{\omega \cdot 2}$-complete safety verification
- data Petri nets: data might be non-fresh
  - decidable boundedness problems (in $\mathbf{F}_{\omega^\omega}$)

Petri Nets with Data

Publications: [FoSSaCS'16, LICS'16]; People: Piotr Hofman, Sławomir Lasota, Ranko Lazić, Jérôme Leroux, Sylvain Schmitz, Patrick Totzke

- tokens carry data



- ν-Petri nets: enforce fresh data
  - $\mathbf{F}_{\omega \cdot 2}$-complete safety verification
- data Petri nets: data might be non-fresh
  - decidable boundedness problems (in $\mathbf{F}_{\omega^\omega}$)

C: Outlook

- deviation from program centered on branching extension of Petri nets
- towards verification of data-centric dynamic systems
  [Vianu, Deutsch, et al.]

Updated Schedule

Task A delayed by roughly a year

Task B currently on track

Task C on track?

Publications

MSc'16 *Systèmes de preuves pour logiques modales*. Anthony Lick. **MSc. Thesis, MPRI**, 2016.

CSL'16 *A Sequent Calculus for a Modal Logic on Finite Data Trees*. David Baelde, Simon Lunel, Sylvain Schmitz. **Proc. CSL 2016**.

LICS'16 *The Complexity of Coverability in ν-Petri Nets*. Ranko Lazić, Sylvain Schmitz. **Proc. LICS 2016**.

FoSSaCS'16 *Coverability Trees for Petri Nets with Unordered Data*. Piotr Hofman, Sławomir Lasota, Ranko Lazić, Jérôme Leroux, Sylvain Schmitz, Patrick Totzke. **Proc. FoSSaCS 2016**.

MSc'15 *Systèmes de preuves pour logiques modales à données*. Simon Lunel. **MSc. Thesis, LMFI**, 2015.

LICS'15 *Demystifying Reachability in Vector Addition Systems*. Jérôme Leroux, Sylvain Schmitz. **Proc. LICS 2015**.

Image Credits

- Scientific Review Public Domain
- Internet1 CC BY-SA 3.0, by Rock1997