MI-PDB, MIE-PDB: **Advanced Database Systems**

Practical class 3:

# XML Path Language (XPath)

22. 3. 2016

**Martin Svoboda**
svoboda@ksi.mff.cuni.cz

http://www.ksi.mff.cuni.cz/~svoboda/courses/2015-2-MIE-PDB/

# Path Expressions

- **Paths**
  - Absolute
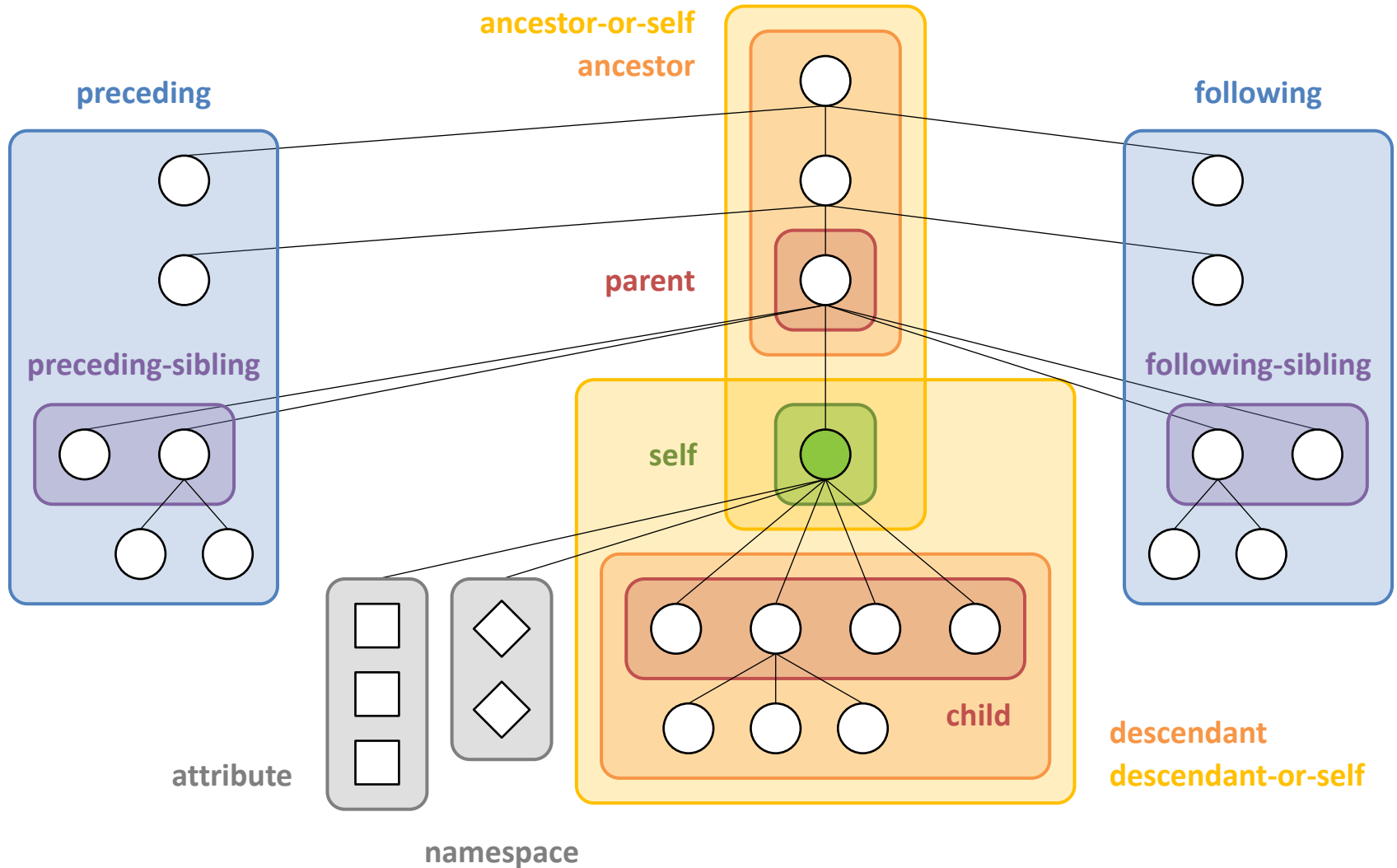    - $/Step_1/Step_2/…/Step_N$
  - Relative
    - $Step_1/Step_2/…/Step_N$
- **Steps**
  - `axis::test predicate`$_1$ `predicate`$_2$ …

# Axes

# Axes

- **Forward axes**
  - self, child, descendant(-or-self), following(-sibling)

- **Reverse axes**
  - parent, ancestor(-or-self), preceding(-sibling)

- **Attributes**
  - attribute

- **Namespace declarations**
  - namespace

# Node Tests

- Tests
  - `node()` – all nodes selected by the axis
  - `text()` – all texts nodes
  - `*` – all elements / attributes selected by the axis
  - *name* – elements / attributes of the given *name*
  - `comment()`
  - `processing-instruction()`

# Abbreviations

- **Abbreviations**
  - …/… <=> …/child::…
  - …/@… <=> …/attribute::…
  - …/.… <=> …/self::node()…
  - …/..… <=> …/parent::node()…
  - …//… <=> …/descendant-or-self::node()/…

# Predicates

- **Predicates**
  - Path expressions: both relative and absolute
  - Comparisons: $= \neq < \leq \geq >$
  - Positions

# Functions

- **A few useful functions**
  - `position(), last()`
  - `count(), sum()`
  - `avg(), min(), max()`
  - `data()`
  - `distinct-values()`
  - …

# Assignment 1

- Express the following XPath queries

  - Use *employees.xml*

  - Return all employees (with their entire subtrees)

  - Return surnames of all employees (just text content)

  - Return these surnames without duplicate values

  - Return salaries of all employees with surname *Smith*

# Assignment 2

- Express the following XPath query

  – Use *employees.xml*

  ▪ Return e-mail addresses of all employees with salaries above the average

# Assignment 3

- Express the following XPath query
  - Use *departments.xml*
  - Return identifiers of all departments with no directly subordinated employees

# Assignment 4

- Express the following XPath query
  - Use *departments.xml*
  - Return the name of the very last department in the whole input document

# Assignment 5

- Express the following XPath query

  – Use *departments.xml*

  ■ Return identifiers of all departments that have at least two subdepartments (even recursively)

# Assignment 6

- Express the following XPath query
  - Use *departments.xml*
    - Return identifier of the top level department that involves a given particular department (e.g. *D1.2.1*)