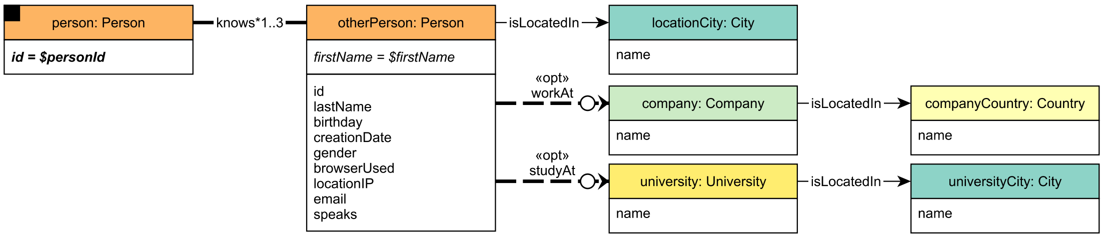
Interactive / complex / 1

IC 1
IC 2
IC 3
IC 4
IC 5
IC 6
IC 7
IC 8
IC 9
IC 10
IC 11
IC 12
IC 13
IC 14v1
IC 14v2

<table>
<tr><td>query</td><td>Interactive / complex / 1</td></tr>
<tr><td>title</td><td>Transitive friends with certain name</td></tr>
<tr><td>pattern</td><td>
</td></tr>
<tr><td>description</td><td>Given a start Person with ID $personId, find Persons with a given first name ($firstName) that the start Person is connected to (excluding start Person) by at most 3 steps via the knows relationships. Return Persons, including the distance (1..3), summaries of the Persons workplaces and places of study.</td></tr>
<tr><td>params</td><td><table>
<tr><td>1</td><td>$personId</td><td>ID</td><td></td></tr>
<tr><td>2</td><td>$firstName</td><td>String</td><td></td></tr>
</table></td></tr>
<tr><td>result</td><td><table>
<tr><td>1</td><td>otherPerson.id</td><td>ID</td><td>R</td><td></td></tr>
<tr><td>2</td><td>otherPerson.lastName</td><td>String</td><td>R</td><td></td></tr>
<tr><td>3</td><td>distanceFromPerson</td><td>32-bit Integer</td><td>C</td><td></td></tr>
<tr><td>4</td><td>otherPerson.birthday</td><td>Date</td><td>R</td><td></td></tr>
<tr><td>5</td><td>otherPerson.creationDate</td><td>DateTime</td><td>R</td><td></td></tr>
<tr><td>6</td><td>otherPerson.gender</td><td>String</td><td>R</td><td></td></tr>
<tr><td>7</td><td>otherPerson.browserUsed</td><td>String</td><td>R</td><td></td></tr>
<tr><td>8</td><td>otherPerson.locationIP</td><td>String</td><td>R</td><td></td></tr>
<tr><td>9</td><td>otherPerson.email</td><td>{Long String}</td><td>R</td><td></td></tr>
<tr><td>10</td><td>otherPerson.speaks</td><td>{String}</td><td>R</td><td></td></tr>
<tr><td>11</td><td>locationCity.name</td><td>String</td><td>R</td><td></td></tr>
<tr><td>12</td><td>universities</td><td>{&lt;String, 32-bit Integer, String&gt;}</td><td>A</td><td>{‹university.name, studyAt.classYear, universityCity.name›}</td></tr>
<tr><td>13</td><td>companies</td><td>{&lt;String, 32-bit Integer, String&gt;}</td><td>A</td><td>{‹company.name, workAt.workFrom, companyCountry.name›}</td></tr>
</table></td></tr>
<tr><td>sort</td><td><table>
<tr><td>1</td><td>distanceFromPerson</td><td>↑</td><td></td></tr>
<tr><td>2</td><td>otherPerson.lastName</td><td>↑</td><td></td></tr>
<tr><td>3</td><td>otherPerson.id</td><td>↑</td><td></td></tr>
</table></td></tr>
<tr><td>limit</td><td>20</td></tr>
<tr><td>CPs</td><td>2.1, 5.3, 8.2</td></tr>
<tr><td>relevance</td><td>This query is a representative of a simple navigational query. It is interesting for several aspects. (1) It requires for a complex aggregation for returning the concatenation of universities, companies, languages and email information of the Person. (2) It tests the ability of the optimizer to move the evaluation of sub-queries functionally dependant on the Person, after the evaluation of the top-k. (3) Its performance is highly sensitive to properly estimating the cardinalities in each transitive path, and paying attention not to explore already visited Persons.</td></tr>
</table>