

Graph-Based Modelling of Students' Interaction Data from Exploratory Learning Environments

Alex Poulouvassilis, Sergio Gutierrez-Santos, Manolis Mavrikis

The Problem

- Much research focuses on interactive educational applications that encourage students' open-ended experimentation within a knowledge domain
- For students to benefit from interaction with such *Exploratory Learning Environments* (ELEs) there is a need for explicit pedagogical support to be provided
- Has led to research in intelligent techniques for providing *personalised, adaptive support* to students, to foster their productive interaction with ELEs
- Data gathered from students' interactions in ELEs has the potential to help pedagogical experts understand how students are interacting with the system, and technical experts to develop enhanced intelligent support features

The Problem

- However, log files from ELEs can contain large quantities of data, making their interpretation difficult and costly
- Also, it is not possible to always know in advance what data are relevant for analytical purposes, therefore an exploratory analysis of the data gathered may be needed
- In this paper we consider how modelling student-system interaction data as a graph makes possible graph-based queries and analyses that can provide insights into the ways that students are using the affordances of the ELE, and the effects of system interventions on students' behaviour
- Our ultimate aim is to feed back this understanding into improving the design of the ELE and its intelligent support
- Two case studies: eXpresser and FractionsLab

eXpresser microworld

File Activities Edit

Page 1

My World

World Colouring Rule

$$5 \times \text{reds } 4 + 3$$

Properties

reds

4 × E

2 →

0 ↓

How many tiles?

5 × (reds 4)

5 × (reds 4)

student feedback: a nudge...

The screenshot shows a software interface for a math model. At the top, there is a header "My Model" with a color palette (red, green, blue, yellow) and a numerical input field containing "35". A green play button and zoom controls are also present. The main workspace is a grid with a pattern of six green vertical bars. A purple dashed box highlights this pattern. Two "Properties" windows are open. The foreground window has a "Make" tab and a "Place" tab. A blue circle highlights the number "6" in a text field, with an orange arrow pointing to it from a yellow feedback box. Below the text field is a slider and a "How many tiles?" section with a "30" input field and a green tile icon. At the bottom, there is a "Model Rule" section with a folder icon and a question mark, a "See previous" button with a lightbulb icon, a progress bar, and a "Help" button.

My Model

35

Properties

Make Place

6

How many tiles?

30

Add these tiles (+)
 Remove these tiles (-)

Properties

5

Would the pattern be coloured if you changed the number of building blocks?

OK

Model Rule

See previous

Help

an unsolicited prompt...

The screenshot shows a software interface for a math problem. At the top, it says "My Model" and has a display showing "20" with a play button and zoom controls. Below this, there are four green vertical bars. A yellow callout box with the text "This is correct. But use a general rule to show the link between these numbers." and an "OK" button is overlaid on the bars. In the center, a "Properties" window shows the equation $5 \times \square = 20$. Below that, a "Model Rule" window asks "How many tiles?" and shows a selection of "20" tiles, which is circled in blue. At the bottom, there are icons for "Model Rule", a "See previous" button, a progress bar, and a "Help" button.

My Model

20

This is correct. But use a general rule to show the link between these numbers.

OK

Properties

$5 \times \square = 20$

Model Rule

How many tiles?

20

• Add these tiles (+)

• Remove these tiles (-)

See previous

Help

feedback after a student's explicit request for help...

The screenshot shows a software interface for a learning environment. At the top, there is a toolbar with a play button and a search icon. Below the toolbar, a yellow dialog box contains the text: "Check the number of tiles in your **building block**. How many building blocks do you have? What's the rule needed here?" with "OK" and "More help..." buttons. Two red arrows point from the "More help..." button to two "Properties" windows. The first "Properties" window has a "Make" tab and shows a "3" tile multiplied by a red square tile, with a "9" tile below it and the text "How many tiles?". The second "Properties" window has a "Place" tab and shows a "2" tile multiplied by a red square tile. At the bottom, there is a "Model Rule" section with a search bar containing "I am trying to", "colour", and "the patterns in My Model.", and a "Help" button. A "See previous" button is also visible in the bottom left corner.

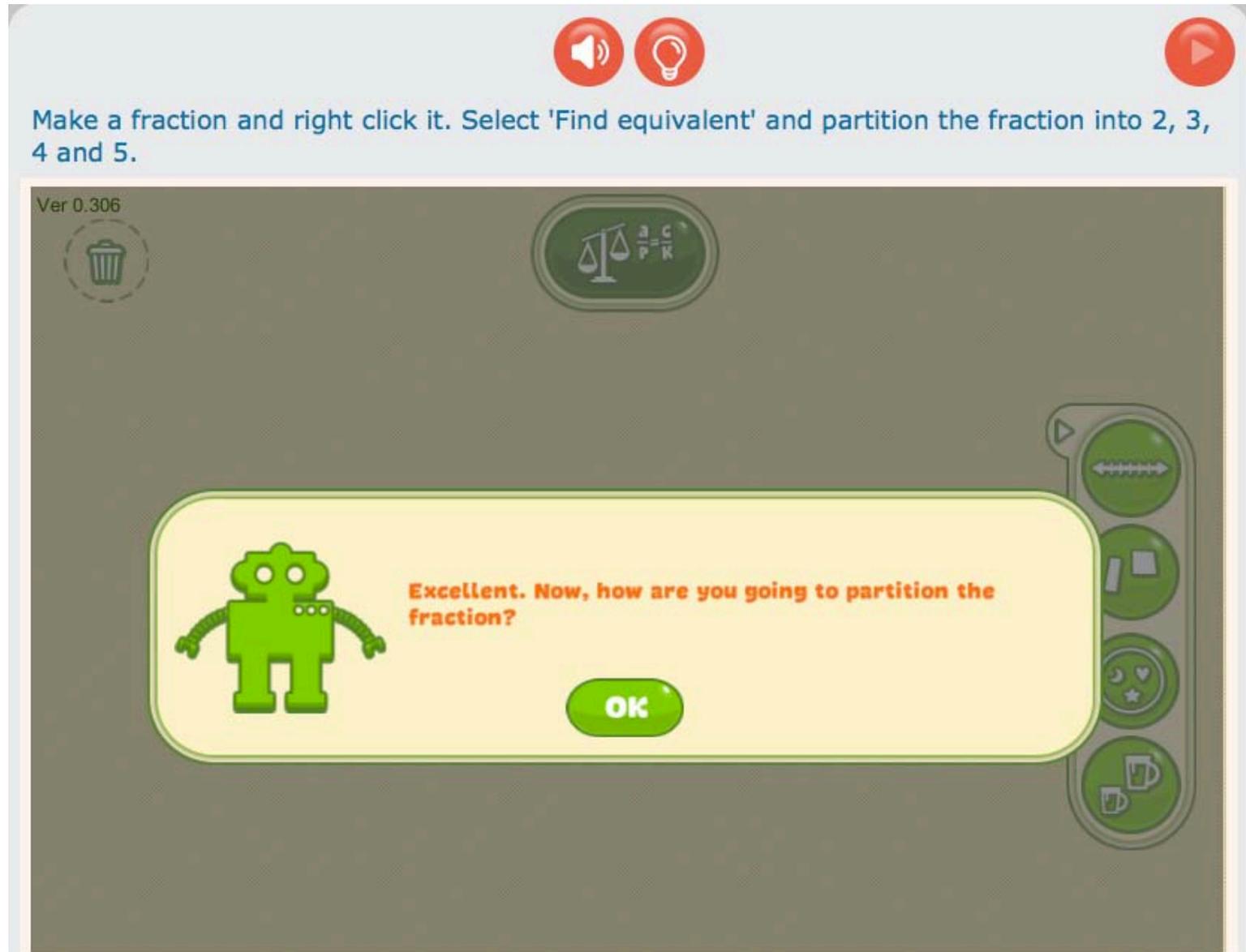
FractionsLab microworld

Make a fraction and right click it. Select 'Find equivalent' and partition the fraction into 2, 3, 4 and 5.

Ver 0.306

The interface features a top navigation bar with three red circular icons: a speaker, a lightbulb, and a play button. Below the instructions, the main workspace is a light green textured area. In the top left, there is a trash can icon. In the top center, there is a green oval icon containing a scale of justice and the mathematical equation $\frac{a}{p} = \frac{c}{k}$. In the center, a fraction $\frac{1}{3}$ is displayed, with the numerator '1' above a horizontal line and the denominator '3' below it. To the left of the fraction is a rectangular bar divided into three equal vertical sections, with the first section on the left filled with red. To the right of the fraction are two small green circular buttons with upward and downward arrows. On the right side of the workspace, there is a vertical toolbar with five green circular buttons: a double-headed arrow, a square with a diagonal line, a circle with a star and two hearts, and a square with a diagonal line and a circle.

Low-interruption feedback



Make a fraction and right click it. Select 'Find equivalent' and partition the fraction into 2, 3, 4 and 5.

Ver 0.306

Excellent. Now, how are you going to partition the fraction?

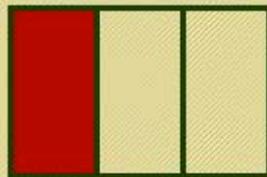
OK

The interface includes a top navigation bar with a speaker icon, a lightbulb icon, and a play button icon. The main workspace contains a trash can icon, a scale icon with the equation $\frac{a}{p} = \frac{c}{k}$, and a vertical toolbar on the right with icons for zooming, erasing, and other tools.

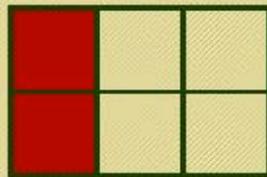


Make a fraction and right click it. Select 'Find equivalent' and partition the fraction into 2, 3, 4 and 5.

Ver 0.306



$$\frac{1}{3}$$

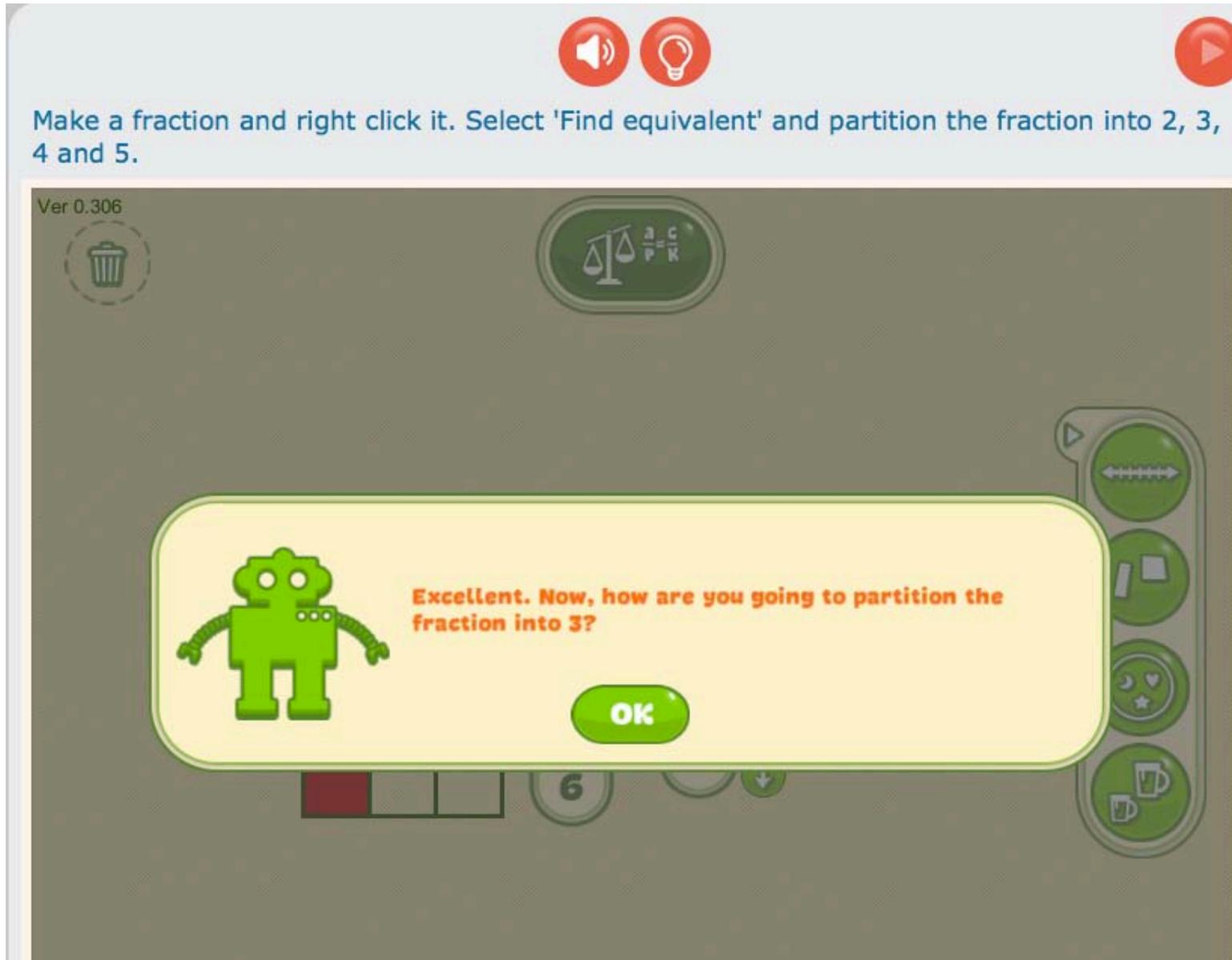


$$\frac{2}{6}$$

$$\frac{2}{2}$$



High-interruption feedback



Make a fraction and right click it. Select 'Find equivalent' and partition the fraction into 2, 3, 4 and 5.

Ver 0.306

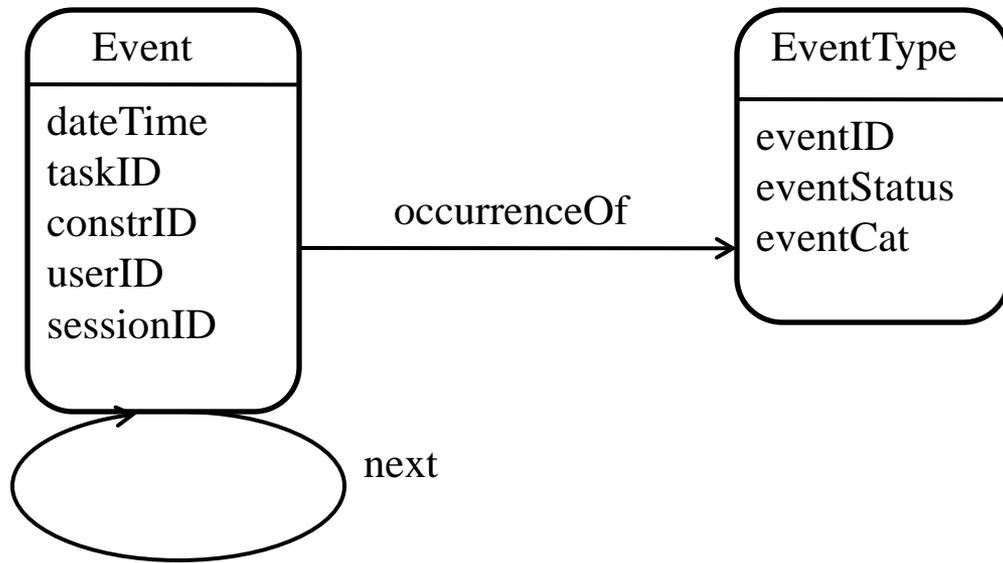
Excellent. Now, how are you going to partition the fraction into 3?

OK

The screenshot shows a software interface with a light blue header bar containing three red circular icons: a speaker, a lightbulb, and a play button. Below the header, a blue instruction text reads: "Make a fraction and right click it. Select 'Find equivalent' and partition the fraction into 2, 3, 4 and 5." The main workspace is a dark grey area with a trash can icon in the top left, a scale icon with the formula $\frac{a}{b} = \frac{c}{d}$ in the top center, and a vertical toolbar on the right with icons for a ruler, a square, a circle, a star, and a cup. A yellow feedback dialog box with rounded corners is centered on the screen, featuring a green robot icon on the left and an "OK" button at the bottom. The dialog text says: "Excellent. Now, how are you going to partition the fraction into 3?". At the bottom of the workspace, a red bar and a number "6" are visible.

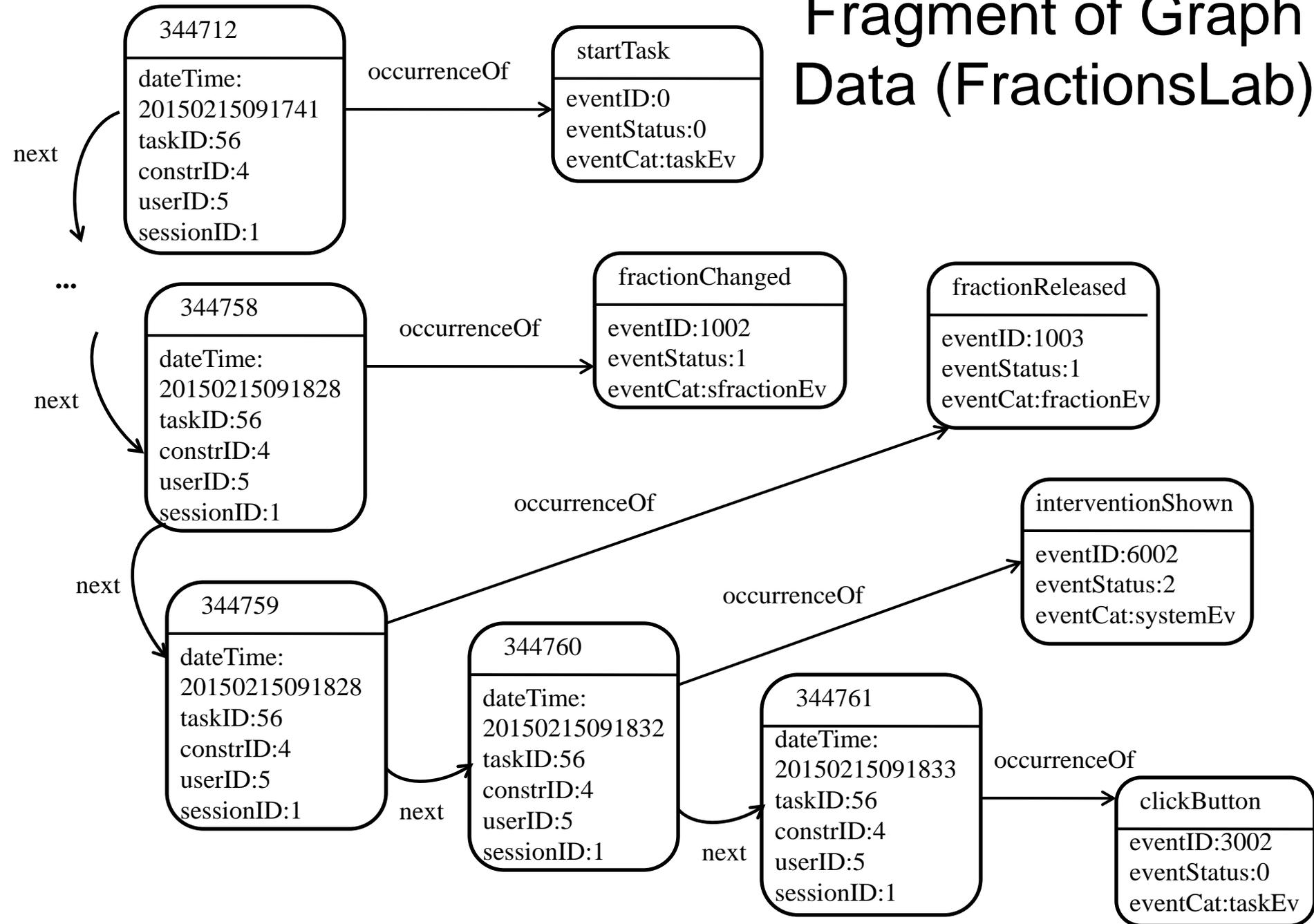
The data gathered

- Event-based data:
 - Students' interactions with the ELE
 - Occurrence of key *indicators* as students interact with the ELE
 - Provision of feedback by the ELE to students
- Students' constructions:
 - including history of development of these
- Task information: learning goals, solution approaches
- Students' learner models



Core Graph Data Model

Fragment of Graph Data (FractionsLab)



Some Exploratory Queries

```
(?X,?Y,?Z) <- (?X, occurrenceOf, interventionShown),  
                (?X, next, ?Y),  
                (?Y, occurrenceOf, ?Z)
```

allows researchers to see types of events directly follow the display of an intervention message; allows confirmation/contradiction of their expectations regarding the immediate effect of intervention messages on students' behaviours

```
(?X,?Y,?Z) <- (?X, occurrenceOf, interventionShown),  
                (?X, next+, ?Y),  
                (?Y, occurrenceOf, ?Z)
```

allows researchers to see types of events directly *or indirectly* follow the display of an intervention message; allows confirmation/contradiction of expectations regarding longer-term effects of intervention messages on students' behaviours

Some Exploratory Queries

modify the query to retain only pairs X, Y that relate to the same construction:

```
(?X,?Y,?Z) <- (?X, occurrenceOf, interventionShown),  
              (?X, constrID, ?C), (?X, next+, ?Y),  
              (?Y, constrID, ?C), (?Y, occurrenceOf, ?Z)
```

use a *path variable* to return the matched paths through the data as well:

```
(?X,?P, ?Y,?Z) <- (?X, occurrenceOf, interventionShown),  
                 (?X, constrID, ?C), (?X, next+ : ?P, ?Y),  
                 (?Y, constrID, ?C), (?Y, occurrenceOf, ?Z)
```

use *approximate matching* to return increasingly longer paths to the user:

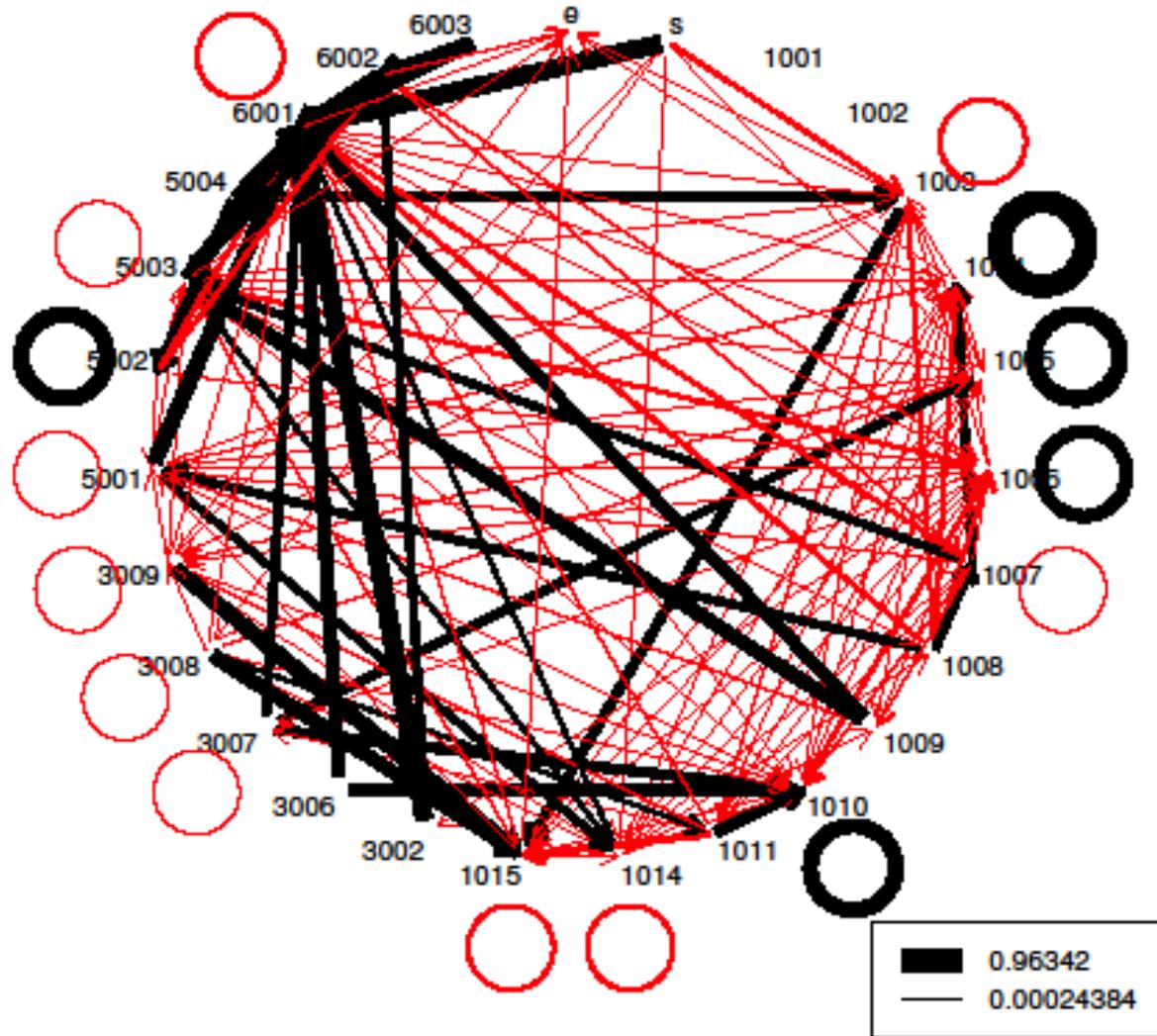
```
(?X,?P, ?Y,?Z) <- (?X, occurrenceOf, interventionShown),  
                 (?X, constrID, ?C), APPROX (?X, next : ?P, ?Y),  
                 (?Y, constrID, ?C), (?Y, occurrenceOf, ?Z)
```

Possible Graph Structure Analyses

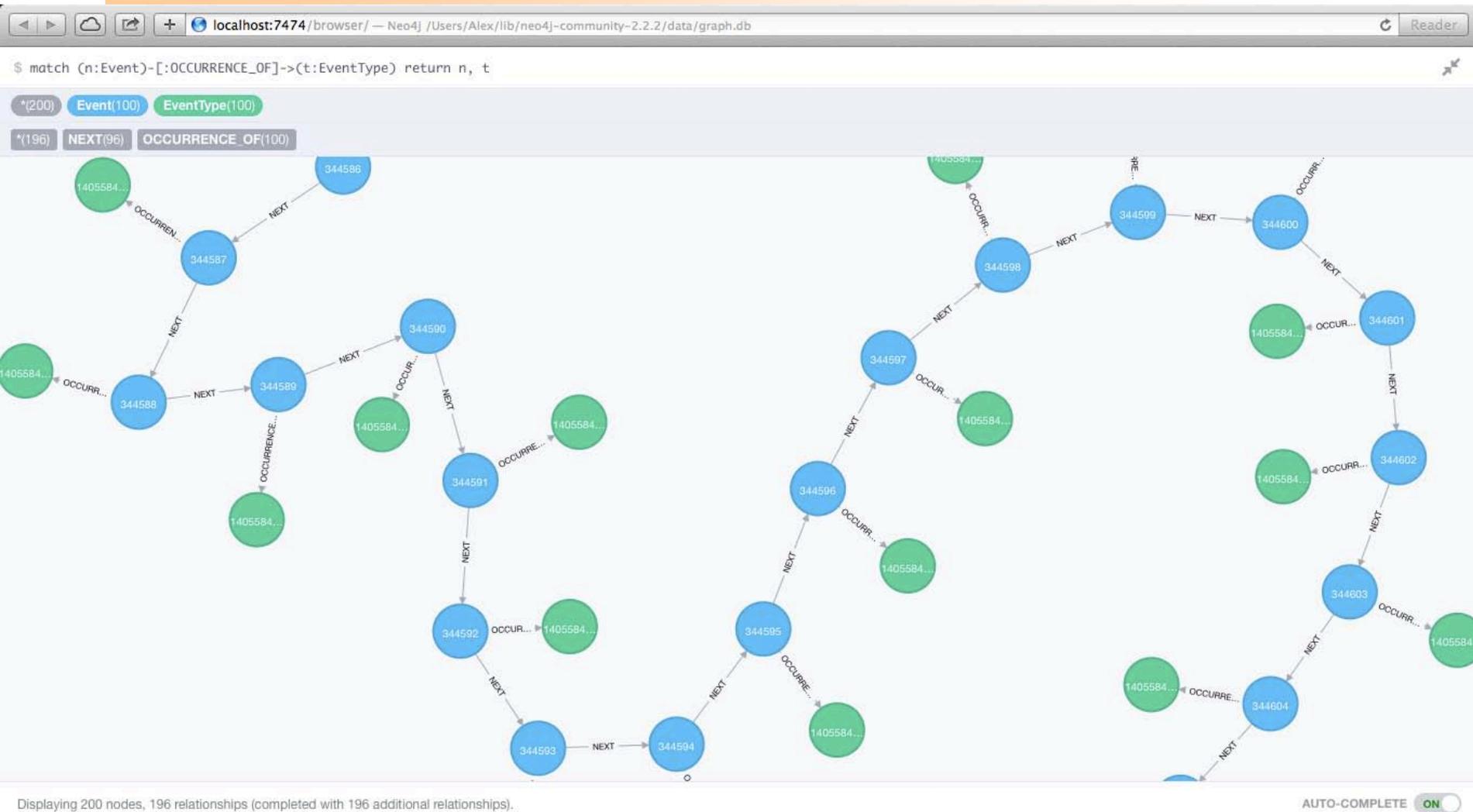
In addition to evaluating queries over the interaction data, by representing the data in the form of a graph it is possible to apply graph structure analyses such as the following:

- *Path finding and clustering*: for determining patterns of interest across a whole dataset, or focussing on particular students, tasks or sessions
- *Average path length*: determining the amount of student activity
- *Graph diameter*: determining the most long-running or most intensive task(s)
- *Degree centrality*: identifying key event types occurring during students' interactions
- *Betweenness centrality*: identifying event types that play key mediating roles between other event types

Ad hoc analyses: Transitions between indicator types



Implementation in a Graph DB (Neo4J)



Displaying 200 nodes, 196 relationships (completed with 196 additional relationships).

AUTO-COMPLETE

Conclusions

- We have presented a general graph model for representing event-based student-system interaction data arising from ELEs
- We have explored the possibilities that evaluating regular path queries over this graph-based representation provides for exploring the behaviours of students as they are working in the ELE and the effectiveness of the ELE's intelligent support
- We have also identified additional graph algorithms that may yield further insights about learners, tasks and significant interaction indicators
- In recent work, we have transformed and uploaded an interaction data set gathered during a trial of FractionsLab into the Neo4J graph database

Future Work

- Design, implementation and evaluation of meaningful queries, analyses and visualisations over the graph data, in collaboration with pedagogical experts and teachers
- Aiming to lead to improved understanding of student-system interaction, and to feed back into enhanced design of the ELEs
- Investigation of how existing flexible querying processing techniques for graph data might be applied/adapted/extended to fine-granularity student-system interaction data, and to the pedagogical setting of providing effective intelligent support to learners undertaking exploratory tasks in ELEs

Acknowledgements

- **MiGen** team members, particularly Eirini Geraniou, Ken Kahn, Darren Pearce-Lazzard, Celia Hoyles, George Magoulas, Richard Noss
- **iTalk2Learn** team members, particularly Beate Grawemeyer, José Luis Fernandez-Gomez, Wayne Holmes
- Alex Wollenschlaeger: interactions data transformation and uploading to Neo4J, generation of new visualisations