

2. Nodes (blocks)

Scientific Poster with TikZ version 2.0

Elena Botoeva
KRDB Research Centre, Free University of Bozen-Bolzano, Italy
botoeva@inf.unibz.it

Starting

Start with the following document:

```
\documentclass[aoposter]{usepackage{tikzposter}} % here most of the things are defined% change parameters only after this line\usepackage[margin=cm, paperwidth=84.1cm, paperheight=118.9cm]{geometry}\title{Title}\author{Author}\institution{\texttt{email}}\begin{document}\AddToShipoutPicture{\BackgroundPicture}\noindent\begin{tikzpicture}\initializesizeandshifts\titleblock[50]{1}\blocknode[Block Title]{Block Content}\startsecondcolumn\blocknode[Block Title 2]{Block Content 2}\end{tikzpicture}\end{document}
```

Macro for creating a block node:
`\blocknode[Block Title]{Block Content}`
Macro `\blocknode` has three parameters. The first one is optional and it is the position of the block. The first block will be automatically placed to $(5\text{fstraw})+(\text{shiftr})+(\text{shiftr})$, which is the left corner below the title block. In most of the templates, (fstraw) is set to (fstraw) , where fstraw is the alias for the title block. Each subsequent block is automatically placed to $(\text{fstraw})+(\text{shiftr})$, i.e. below the previous block aligned box. You can also use an explicit parameter: e.g. $(-10,20)$ (note that $(0,0)$ is the center of the poster). The second parameter is the title of the block. Finally, the last parameter is the actual content.

Making Title

To make title, use the standard commands `\title` and `\author` in the preamble, and then the following macro:

```
\titleblock[50]{1}
```

Macro `\titleblock` has three parameters. The first one is optional and it specifies the shift of the title block w.r.t. its default position, which is set to $(50.5^{\circ}(0,\text{paperheight})-(0,\text{margin}))$. The second parameter is the width of the title block, and the third parameter is the scaling ratio (to make the title bigger or smaller).

The syntax for specifying authors is similar to the one in `aaastity`. Author information can be set in various styles. For several authors from the same institution:

```
\author{Author 1 \and \and Author n}\address line \{ ... \}\address line
```

```
\author{Author 1 \{Mf Author 2 \} \{Mf Author n \}}\address line \{ ... \}\address line
```

```
For authors from different institutions:\author{Author 1 \{Address line \{ ... \}\address line \and \and Author n \{Address line \{ ... \}\address line \}
```

```
To start a separate "row" of authors use \AND as in\author{Author 1 \{Address line \{ ... \}\address line \AND Author 2 \{Address line \{ ... \}\address line \and Author 3 \{Address line \{ ... \}\address line \}
```

```
(though, I must say \and ... \and did not work for me with more than 2 authors, so just use commas where you need it if it does not work for you either).
```

Variable Width Block Nodes

You can also create blocks of arbitrary width
`\blocknode[coordinate]{Block width}{Block Title}{Block Content}`
In this case it is better to specify coordinate manually if you want to have blocks aligned vertically.

Note that `(shiftr)` and `(shiftr)` are coordinates created in macro `\initializesizeandshifts`, and they allow to have relative positioning of blocks nodes in an automatic fashion. If you want to define your own shifts, set new values for `(shiftr)` and `(shiftr)` using commands `\setshiftr` and `\setshiftr`.

Also, it might be useful to know the y -coordinate of the south border of the previous block. You can retrieve it by using the command `\getcurrentrow[box] or \getcurrentrow[note]`. This coordinate will be stored in `(currentrow)`, which can be used to specify the location of the next block node.

Block Nodes in the Second Column

To start the second column or the third column use commands `\startsecondcolumn`, and `\startthirdcolumn`. If the number of columns is 2, then the last command will not have effect. You can also start a new column with an arbitrary x -coordinate by specifying explicitly the coordinate of the new block node as follows:
`\blocknode[$(5\text{fstraw})+(\text{shiftr})+(\text{shiftr})$]{Block Title}{Block Content}`

Colored Boxes Inside Block Nodes

There are three types of colored boxes/blocks that you can use inside `\blocknode` to highlight information.

Theorem	<code>\innerblock{Theorem}{Statement}</code>
Text	<code>\innerblockplain[colorone]{Text}</code>
Text	<code>\coloredbox[colorthree]{Text}</code>

There are also callout nodes that allow for a more interesting layout of the poster:
`\callout[rotate angle][from coordinate][coordinate]{Node Width}{Node Content}`
The alias for such nodes is `note`.

Plain nodes These nodes are similar to callout nodes. They allow for specifying the title of the node.
`\plainnode[rotate angle][coordinate]{Node Width}{Node Title}{Node Content}`

Personalizing the Poster

It is possible to adjust the layout of the poster. To impose your own setting, you can use these macros:

- Macros for changing sizes
`\setmargin[4]`, `\setheadheight[14]`, `\setintitushift[10]`, `\setblockspacing[2]`, `\setblocktitleheight[3]`
- Other structural macros
`\setcolumnnumber[3]`, `\usetemplate[5]`, `\usecolortemplate[4]`, `\usebackgroundtemplate[5]`, `\usetitletemplate[2]`, `\useblocknode[3]`, `\useinnernode[3]`, `\useplainnode[3]`
- Macro for adding logos to the title block
`\addlogo[south west]{ $(0,0)$ }{ fstraw }{filename}`
- Macros for the basic colors
`\setfirstcolor[green70!]`, `\setsecondcolor[gray80!]`, `\setthirdcolor[red80!black]`
- Macros for specific colors:
`\setbackgrounddarkcolor[colorone 70!black]`, `\setbackgroundlightcolor[colorone 70!white]`,
`\setfirsttextcolor[textcolor]`, `\setfirstfillcolor[white]`, `\setfirstrowcolor[colortwo]`,
`\setblocktextcolor[textcolor]`, `\setblockfillcolor[white]`,
`\setblocktitletextcolor[colorone]`, `\setblocktitlefillcolor[colortwo]`,
`\setplainblocktextcolor[textcolor]`, `\setplainblockfillcolor[colorthree 40]`,
`\setplainblocktitletextcolor[textcolor]`, `\setplainblocktitlefillcolor[colorthree 60]`,
`\setinnerblocktextcolor[textcolor]`, `\setinnerblockfillcolor[white]`,
`\setinnerblocktitetextcolor[white]`, `\setinnerblocktitlefillcolor[colorthree]`

TikZposter template
It is a template for scientific posters based on `aoposter` and `TikZ` only. The current version contains five different templates (see my posters [here](#) and [here](#)). The sources of this pdf file can be found [here](#).

The title block

Blocknode

Innerblock

Calloutnode

Titleblock and **blocknode** can be placed automatically (i.e., coordinate is optional), **calloutnode** and **plainnode** require explicit coordinates and can be rotated. These four nodes are on the level of TikZ with slightly different sets of parameters.

I think it is **IMPORTANT** that each node can have *arbitrary position and width*, just the same way as if I had a graphical interface to do so (like Powerpoint). Moreover, after a node is drawn, it might be useful to have the alias of that node (e.g., to relatively place a callout node). In general, with these nodes one should have the full power of TikZ, in my opinion.

Innerblock behaves like a normal latex environment. Should be similar to Theorem or Block in Beamer.

Plainnode

2. Nodes' color schemes

Scientific Poster with TikZ version 2.0

Elena Boteva
KRDB Research Centre, Free University of Bozen-Bolzano, Italy
boteva@inf.unibz.it

Starting

Start with the following document:

```
\documentclass[aposter]{usepackage{tikzposter}} % here most of the things are defined% change parameters only after this line\usepackage[margin=cm, paperwidth=84.1cm, paperheight=118.9cm]{geometry}\title{Title}\author{Author}\institution{\texttt{email}}\begin{document}\AddToShipoutPicture{\BackgroundPicture}\margin\begin{tikzpicture}\initialsizeandshifts\titleblock[50]{1}\blocknode[Block Title]{Block Content}\startsecondcolumn\blocknode[Block Title 2]{Block Content 2}\end{tikzpicture}\end{document}
```

Macro for creating a block node:
`\blocknode[Block Title]{Block Content}`
Macro `\blocknode` has three parameters. The first one is optional and it is the position of the block. The first block will be automatically placed to `(5){lstraw}{xshft}{yshft}`, which is the left corner below the title block. In most of the templates, `(lstraw)` is set to `(left south)`, where `left` is the alias for the title block. Each subsequent block is automatically placed to `(lstraw south){yshft}`, i.e. below the previous block aligned box. You can also use an explicit parameter, e.g. `(-10:20)` (note that `(0,0)` is the center of the poster). The second parameter is the title of the block. Finally, the last parameter is the actual content.

Block Nodes in the Second Column

To start the second column or the third column use commands `\startsecondcolumn`, and `\startthirdcolumn`. If the number of columns is 2, then the last command will not have effect. You can also start a new column with an arbitrary x-coordinate by specifying explicitly the coordinate of the new block node as follows:
`\blocknode[5]{lstraw}{yshft}+(x,0){Block Title}{Block Content}`

Colored Boxes Inside Block Nodes

There are three types of colored boxes/blocks that you can use inside a block node to highlight information.

Theorem	<code>\innerblock[Theorem]{Statement}</code>
Text	<code>\innerblockplain[colorone!80]{Text}</code>
Text	<code>coloredbox[threethree!50]{Text}</code>

Making Title

To make title, use the standard commands `\title` and `\author` in the preamble, and then the following macro:
`\titleblock[50]{1}`
Macro `\titleblock` has three parameters. The first one is optional and it specifies the shift of the title block w.r.t. its default position, which is set to `(50.5*0){paperheight}-(0,0,margin)`. The second parameter is the width of the title block, and the third parameter is the scaling ratio (to make the title bigger or smaller).

The syntax for specifying authors is similar to the one in `aa.sty`. Author information can be set in various styles. For several authors from the same institution:
`\author{Author 1 \and \and Author n}`
`Address line \ - \ Address line`
If the names do not fit well on one line use
`\author{Author 1}{\M{Author 2}}{\ - \}{\M{Author n}}`
`Address line \ - \ Address line`
For authors from different institutions:
`\author{Author 1}{\ Address line \ - \}{\ Address line}`
`\and \ - \ Address line \ - \}{\ Address line}`
to start a separate "row" of authors use `\AND` as in
`\author{Author 1}{\ Address line \ - \}{\ Address line \AND}`
`Author 2}{\ Address line \ - \}{\ Address line \AND}`
`Author \}{\ Address line \ - \}{\ Address line}`
(though, I must say `\and \ - \and` did not work for me with more than 2 authors, so just use commas where you need it if it does not work for you either).

Variable Width Block Nodes

You can also create blocks of arbitrary width
`\blocknode[coordinate]{Block width}{Block Title}{Block Content}`
In this case it is better to specify coordinate manually if you want to have blocks aligned vertically.
Note that `(xshft)` and `(yshft)` are coordinates created in macro `\initialsizeandshifts`, and they allow to have relative positioning of block nodes in an automatic fashion. If you want to define your own shifts, set new values for `(xshft)` and `(yshft)` using commands `\setxshft` and `\setyshft`.
Also, it might be useful to know the y-coordinate of the south border of the previous block. You can retrieve it by using the command `\getcurrentrow{box}` or `\getcurrentrow{note}`. This coordinate will be stored in `(currentrow)`, which can be used to specify the location of the next block node.

There are also callout nodes that allow for a more interesting layout of the poster:
`\calloutnode[rotate angle]{from coordinate}{coordinate}{Node Width}{Node Content}`
The alias for such nodes is `note`.

Plain nodes These nodes are similar to callout nodes - they allow for specifying the title of the node.
`\plainnode[rotate angle]{coordinate}{Node Width}{Node Title}{Node Content}`

Personalizing the Poster

It is possible to adjust the layout of the poster. To impose your own style, you can use these macros:
• Macros for changing sizes
`\setmargin[4]{\setheaddrawingheight[14]{\setminititleshift[10]{\setblockspacing[2]{\setblocktitleheight[3]`
• Other structural macros
`\setcolumnnumber[3]{\usetemplate[5]{\usecolortemplate[4]{\usebackgroundtemplate[5]{\settitledtemplate[2]{\useblocknodetemplate[3]{\useinnernodetemplate[3]{\useplainnodetemplate[4]`
• Macro for adding logos to the title block
`\addlogo[south west]{(0,0){6cm}{filename}`
• Macros for the basic colors
`\setfirstcolor{green!70!}{\setsecondcolor{gray!80!}{\setthirdcolor{red!80!black}`
• Macros for specific colors:
`\setbackgrounddarkcolor{colorone!70!black}{\setbackgroundlightcolor{colorone!70!}{\setfirsttextcolor{textcolor}{\settitlefillcolor{white}{\setfirstrowcolor{colortwo}{\setblocktextcolor{textcolor}{\setblockfillcolor{white}{\setblocktitletextcolor{colorone}{\setblocktitlefillcolor{colortwo}{\setplainblocktextcolor{textcolor}{\setplainblockfillcolor{colorthree!40}{\setplainblocktitletextcolor{textcolor}{\setplainblocktitlefillcolor{colorthree!60}{\setinnerblocktextcolor{textcolor}{\setinnerblockfillcolor{white}{\setinnerblocktitletextcolor{white}{\setinnerblocktitlefillcolor{colorthree,`

TikZposter template
It is a template for scientific posters based on `aposter` and TikZ only. The current version contains five different templates (see my posters [here](#) and [here](#)). The sources of this pdf file can be found [here](#).

The title block

Blocknode

Innerblock

Calloutnode

Each of these nodes have 4 color parameters:

- textcolor** -- the color of the main text
- fillcolor** -- the color of the background of the main text
- titletextcolor** -- the color of the title text
- titlefillcolor** -- the color of the title background/the color of the frame

Plainnode

2. Nodes' templates (themes)

The title block

Blocknode

Innerblock

Calloutnode

Plainnode

Scientific Poster with TikZ version 2.0



Elena Botoeva
KRDB Research Centre, Free University of Bozen-Bolzano, Italy
botoeva@inf.unibz.it

Starting

Start with the following document:

```
\documentclass{aoposter}
\usepackage{tikzposter} % here most of the things are defined
% change parameters only after this line
\usepackage{margin}[margin cm, paperwidth=84.1cm, paperheight=118.9cm]{geometry}
\title{Title}
\author{Author \institute{\texttt{email}}
\begin{document}
\AddToShipoutPicture{\BackgroundPicture}
\independent
\begin{tikzpicture}
\initialsizesandshifts
\titleblock{50}{1}
\blocknode{Block Title}[Block Content]
\startsecondcolumn
\blocknode{Block Title 2}[Block Content 2]
\end{tikzpicture}
\end{document}
```

Macro for creating a block node:
`\blocknode{Block Title}[Block Content]`
Macro `\blocknode` has three parameters. The first one is optional and it is the position of the block. The first block will be automatically placed to `(\firstrow-\cofirst\qfirst)5`, which is the left corner below the title block. In most of the templates, `(firstrow)` is set to `title south`, where `title` is the alias for the title block. Each subsequent block is automatically placed to `(\Block south-\qfirst)5`, i.e. below the previous block aligned box. You can also use an explicit parameter, e.g., `(-10,30)` (note that `(0,0)` is the center of the poster). The second parameter is the title of the block. Finally, the last parameter is the actual content.

Block Nodes in the Second Column

To start the second column or the third column use commands `\startsecondcolumn`, and `\startthirdcolumn`. If the number of columns is 2, then the last command will not have effect.

You can also start a new column with an arbitrary x-coordinate by specifying explicitly the coordinate of the new block node as follows:
`\blocknode{5(\firstrow-\qfirst)+(x,0)}[Block Title][Block Content]`

Colored Boxes Inside Block Nodes

There are three types of colored boxes/blocks that you can use inside block nodes to highlight information:

```
Theorem \innerblock{Theorem}[Statement]
Statement
Text \innerblockplain{colorone!80!}[Text]
Text \coloredbox{colorthree!50!}[Text]
```

There are also callout nodes that allow for a more interesting layout of the poster:
`\calloutnode[rotate angle][from coordinate]{coordinate}[Node Width][Node Content]`
The alias for such nodes is `note`.

Plain nodes These nodes are similar to callout nodes. They allow for specifying the title of the node.
`\plainnode[rotate angle][coordinate][Node Width][Node Title][Node Content]`

Making Title

To make title, use the standard commands `\title` and `\author` in the preamble, and then the following macro:
`\titleblock{50}{1.5}`

Macro `\titleblock` has three parameters. The first one is optional and it specifies the shift of the title block w.r.t. its default position, which is set to `(50.5*(0,paperheight)-0)(margin)5`. The second parameter is the width of the title block, and the third parameter is the scaling ratio (to make the title bigger or smaller).

The syntax for specifying authors is similar to the one in `aaastty`. Author information can be set in various styles. For several authors from the same institution:

```
\author{Author 1 \and \and Author n \institute{Institution}}
Address line \institute{Institution}
If the names do not fit well on one line use
\author{Author 1 \institute{Institution} \and Author 2 \institute{Institution} \and Author n \institute{Institution}}
Address line \institute{Institution}
For authors from different institutions:
\author{Author 1 \institute{Institution 1} \and Author 2 \institute{Institution 2} \and Author n \institute{Institution n}}
Address line \institute{Institution 1} \and \institute{Institution 2} \and \institute{Institution n}
To start a separate 'row' of authors use \AND as in
\author{Author 1 \institute{Institution 1} \and Author 2 \institute{Institution 2} \AND Author 3 \institute{Institution 3} \and Author 4 \institute{Institution 4}}
Address line \institute{Institution 1} \and \institute{Institution 2} \and \institute{Institution 3} \and \institute{Institution 4}
(though I must say \and ... \and did not work for me with more than 2 authors, so just use commas where you need it if it does not work for you either)
```

Variable Width Block Nodes

You can also create blocks of arbitrary width
`\blocknode[coordinate][Block width][Block Title][Block Content]`
In this case it is better to specify coordinate manually if you want to have blocks aligned vertically.

Note that `(south)` and `(qfirst)` are coordinates created in macro `\initialsizesandshifts`, and they allow to have relative positioning of block nodes in an automatic fashion. If you want to define your own shifts, set new values for `(south)` and `(qfirst)` using commands `\setsouth` and `\setqfirst`.

Also, it might be useful to know the y-coordinate of the south border of the previous block. You can retrieve it by using the command `\getcurrentrow[box] \getcurrentrow[rate]`
This coordinate will be stored in `(currentrow)`, which can be used to specify the location of the next block node.

Personalizing the Poster

It is possible to adjust the layout of the poster. To impose your own setting, you can use these macros:

- Macros for changing sizes
`\setmargin[4]`, `\sethaddrawingheight[14]`, `\setinstituteshift[10]`, `\setblockspacing[2]`, `\setblocktitleheight[3]`
- Other structural macros
`\setcolumnnumber[3]`, `\usetemplate[5]`, `\usecolortemplate[4]`, `\usebackgroundtemplate[5]`, `\usetitletemplate[2]`, `\useblocknodetemplate[5]`, `\useinnernodetemplate[3]`, `\useplainnodetemplate[4]`
- Macro for adding logos to the title block
`\addlogo[south west]{(0,0)[6cm]{filename}}`
- Macros for the basic colors
`\settextcolor{green!70!}`, `\setsecondcolor{gray!80!}`, `\setthirdcolor{red!80!black}`
- Macros for specific colors
`\setbackgrounddarkcolor{colorone!70!black}`, `\setbackgroundlightcolor{colorone!70!}`, `\settextcolor{textcolor}`, `\setfillcolor{white}`, `\settitlecolor{colortwo}`, `\setblocktitlecolor{textcolor}`, `\setblockfillcolor{white}`, `\setblocktitlecolor{colorone}`, `\setblockfillcolor{colortwo}`, `\setplainblocktextcolor{textcolor}`, `\setplainblockfillcolor{colorthree!40!}`, `\setplainblocktitlecolor{textcolor}`, `\setplainblocktitlefillcolor{colorthree!60!}`, `\setinnerblocktextcolor{textcolor}`, `\setinnerblockfillcolor{white}`, `\setinnerblocktitlecolor{textcolor}`, `\setinnerblocktitlefillcolor{colorthree}`.

TikZposter template
It is a template for scientific posters based on `aoposter` and `TikZ` only. The current version contains five different templates (see my posters [here](#) and [here](#)). The sources of this pdf file can be found [here](#).

I have
- 5 different **blocknode** templates,
- 4 different **plainnode/calloutnode** templates,
- 2 different **innerblock** templates.

Note that for some of the node templates you should NOT be able to change roundedness, shadows, etc.

The code for some of the nodes is a bit dirty (I was using some tricks to draw, e.g., the plainnodes/calloutnodes on this slide)

3. Colors

In my implementation I have THREE basic colors, called very simply :-)
colorone (default blue), **colortwo** (default gray) and **colorthree** (default orange). We could call them similarly to beamer, they have **structure**, etc.

In my opinion there should not be more than 3 basic colors (bear in mind white and black). This number of colors should be enough for a good design.

All other colors (like titlefillcolor of nodes, the colors of the background, etc.) are defined from the basic three colors. So by changing these three colors, the user can quickly change the colors of the whole poster.

Then the individual colors can be adjusted in arbitrary ways.

4. Background

In my implementation the header (or footer) drawing is a part of the background

Scientific Poster with TikZ version 2.0
Elena Botoeva
KRDB Research Centre, Free University of Bozen-Bolzano, Italy
botoeva@inf.unibz.it

Starting Block Nodes in the Second Column

Scientific Poster with TikZ version 2.0
Elena Botoeva
KRDB Research Centre, Free University of Bozen-Bolzano, Italy
botoeva@inf.unibz.it

Starting Block Nodes in the Second Column

Scientific Poster with TikZ version 2.0
Elena Botoeva
KRDB Research Centre, Free University of Bozen-Bolzano, Italy
botoeva@inf.unibz.it

Starting Block Nodes in the Second Column

Colored Boxes Inside Block Nodes

Personalizing the Poster

Variable Width

Variable Width Block Nodes

TikZposter template

5. Structure

Scientific Poster with TikZ version 2.0



Elena Botoeva
 KRDB Research Centre, Free University of Bozen-Bolzano, Italy
 botoeva@inf.unibz.it

In my implementation I have the following 'structural' parameters

1) Margin

2) Blockspacing

3) Headdrawingheight

Starting

Start with the following document:

```
\documentclass[a0poster]
\usepackage{tikz} % here most of the things are defined
% change parameters only after this line
\usepackage[margin= cm, paperwidth=84.1cm, paperheight=118.9cm]{geometry}

\title{Title}
\author{Author\Institution\\\texttt{email}}
\begin{document}
\AddToShipoutPicture{\BackgroundPicture}

\noinvent
\begin{tikzpicture}
\initializesizeandshifts

\titleblock{50}{1}
\blocknode{Block Title}{Block Content}
\startsecondcolumn
\blocknode{Block Title 2}{Block Content 2}
\end{tikzpicture}
\end{document}
```

Macro for creating a block node:
`\blocknode{Block Title}{Block Content}`
 Macro `\blocknode` has three parameters. The first one is optional and it is the position of the block. The first block will be automatically placed to $(\$(firstrow)-(\$shift)+(\$x,0)\$)$, which is the left corner below the title block. In most of the templates, $(firstrow)$ is set to $(title.south)$, where $title$ is the alias for the title block. Each subsequent block is automatically placed to $(\$(box.south)-(\$shift)\$)$, i.e., below the previous block aliased *box*. You can also use an explicit parameter, e.g., $(-10, 30)$ (note that $(0,0)$ is the center of the poster). The second parameter is the title of the block. Finally, the last parameter is the actual content.

Block Nodes in the Second Column

To start the second column or the third column use commands `\startsecondcolumn`, and `\startthirdcolumn`.
 If the number of columns is 2, then the last command will not have effect.
 You can also start a new column with an arbitrary x-coordinate by specifying explicitly the coordinate of the new block node as follows:
`\blocknode{(\$(firstrow)-(\$shift)+(x,0)\$)}{Block Title}{Block Content}`

Colored Boxes Inside Block Nodes

There are three types of colored boxes/blocks that you can use inside block nodes to highlight information.

```
\innerblock{Theorem}{Statement}
\innerblockplain[color=orange!80]{Text}
\coloredbox{colorthree!50}{Text}
```

There are also callout nodes that allow for a more interesting layout of the poster.
`\calloutnode[rotate angle]{from coordinate}{coordinate}{Node Width}{Node Content}`
 The alias for such nodes is *note*.

Plain nodes These nodes are similar to callout nodes. They allow for specifying the title of the node.
`\plainnode[rotate angle][coordinate]{Node Width}{Node Title}{Node Content}`

Making Title

To make title, use the standard commands `\title` and `\author` in the preamble, and then the following macro:

```
\titleblock{50}{1.5}
```

Macro `\titleblock` has three parameters. The first one is optional and it specifies the shift of the title block w.r.t. its default position, which is set to $(\$(0.5*(0,paperheight)-(0,margin)\$)$. The second parameter is the width of the title block, and the third parameter is the scaling ratio (to make the title bigger or smaller).

The syntax for specifying authors is similar to the one in `aaai.sty`. Author information can be set in various styles: For several authors from the same institution:

```
\author{Author 1 \and ... \and Author n \\\
Address line \\\ ... \\\ Address line}

If the names do not fit well on one line use
\author{Author 1 \\\ {\bf Author 2} \\\ ... \\\ {\bf Author n} \\\
Address line \\\ ... \\\ Address line}
```

For authors from different institutions:
`\author{Author 1 \\\ Address line \\\ ... \\\ Address line`
`\And ... \And`
`Author n \\\ Address line \\\ ... \\\ Address line}`

To start a separate "row" of authors use `\AND`, as in
`\author{Author 1 \\\ Address line \\\ ... \\\ Address line \AND`
`Author 2 \\\ Address line \\\ ... \\\ Address line \AND`
`Author 3 \\\ Address line \\\ ... \\\ Address line}`

(though, I must say `\and` and `\AND` did not work for me with more than 2 authors, so just use commas where you need if it does not work for you either).

Variable Width Block Nodes

You can also create blocks of arbitrary width
`\blocknodew[coordinate]{Block width}{Block Title}{Block Content}`

Personalizing the Poster

It is possible to adjust the layout of the poster. To impose your own setting, you can use these macros:

- Macros for changing sizes
`\setmargin{4}`, `\setheaddrawingheight{14}`, `\setinstituteshift{10}`,
`\setblockspacing{2}`, `\setblocktitleheight{3}`
- Other structural macros
`\setcolumnnumber{3}`, `\setemplate{5}`,
`\usecolortemplate{4}`, `\usebackgroundtemplate{5}`, `\usetitletemplate{2}`,
`\useblocknodetemplate{5}`, `\useinnodetemplate{3}`, `\useplainnodetemplate{4}`
- Macro for adding logos to the title block
`\addlogo[south west]((0,0)){6cm}{filename}`
- Macros for the basic colors

The default positions of nodes and their widths are calculated automatically taking into account these parameters.

Examples of my posters

Description Logic Knowledge Base Exchange

Elena Botova supervised by Diego Calvanese
KROB Research Centre, Free University of Bozen-Bolzano, Italy
botova@inf.unibz.it

Data Exchange vs. Knowledge Base Exchange

Knowledge Base Exchange is a special case of Data Exchange with incomplete information.

Data Exchange

Each database instance is complete, i.e., every fact is either true or false. Therefore, represents exactly one possible instance (full).

Data Exchange Example

Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$
 $S_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$
 $S_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$
 Then M is a mapping from S_1 to S_2 .
 S_1 is a universal solution for M .

Knowledge Base Exchange

A Description Logic (DL) Knowledge Base consists of

- **concept** dots given as an **ABox** and
- **structural rules** for completing data given as a **TBox**, therefore, is a compact representation of (possibly infinitely) many actual instances.

Knowledge Base Exchange Example

Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$, $S_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$, $S_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$.
 Then M is a mapping from S_1 to S_2 .
 S_1 is a universal solution for M .

We consider exchange of Description Logic KBs: each KB is constituted by a TBox and an ABox, and mapping is a set of DL inclusions from the source signature to the target signature. We start with lightweight DLs: \mathcal{EL} , \mathcal{EL}_H and \mathcal{EL}_H^* .

(Computing) KB Solutions

The basic reasoning problem is to compute a universal solution.

- K_2 is said to be a **universal solution** for K_1 under a mapping M if $\text{MBox}(K_1) \subseteq \text{MBox}(K_2)$.
- $K_2 = (T_2, A_2)$ is a universal solution for $K_1 = (T_1, A_1)$ under M if $A_2 \subseteq \text{chase}_{\text{MBox}(K_1)}(A_1)$.

- Deciding existence of a universal solution for the first of DL-Lite in \rightarrow is PTime (this problem has undecidability of QBF) and is in PTime (using two-way alternating automata).

We are also interested in query solutions for a class of queries Q .

- K_2 is said to be a **universal Q-solution** for K_1 under M if Q is Q over S_2 $\text{entails}_M(Q)$ over S_1 .
- $K_2 = (T_2, A_2)$ is a **univ. UCO-solution** for $K_1 = (T_1, A_1)$ under M if $\text{chase}_{\text{MBox}(K_1)}(A_1) \subseteq \text{chase}_{\text{MBox}(K_2)}(A_2)$.
- The same complexity results hold for universal UCO-solutions.

Universal Solutions vs. Universal UCO-Solutions

Universal UCO-Solutions are the preferred solutions in our setting.

Example	Universal Solutions	Universal UCO-Solutions
<p>emptiness</p> <p>Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$ $S_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$ $S_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$ $A_1 = \{ \text{Author}(\text{Diana}) \}$ $A_2 = \{ \text{Author}(\text{Diana}), \text{Editor}(\text{Elena}) \}$</p>	<p>• universal solutions cannot have a non-empty TBox</p> <p>Some source $D_1 \subseteq S_1$ is not related by T_1 to $(T_1, M(A_1))$, E_1 is not related to $(T_1, M(A_1))$.</p>	<p>• universal UCO-solutions have a non-empty TBox</p> <p>$K_2 = (T_2, A_2)$ with $T_2 = \{ \text{Author}(\text{Diana}), \text{Editor}(\text{Elena}) \}$ is a univ. UCO-solution for K_1 under M.</p>
<p>reflexive TBox-chase</p> <p>Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$ $S_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$ $S_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$ $A_1 = \{ \text{Author}(\text{Diana}) \}$</p>	<p>• universal solutions do not exist</p> <p>Some $\text{chase}_{\text{MBox}(K_1)}(A_1)$ is not related to $(T_1, M(A_1))$ and there are no ABox inclusions only dependent to T_1-infix.</p>	<p>• universal UCO-solutions exist</p> <p>Some $K_2 = (T_2, A_2)$ with $T_2 = \{ \text{Author}(\text{Diana}), \text{Editor}(\text{Elena}) \}$ is a univ. UCO-solution for K_1 under M.</p>
<p>full joining tree</p> <p>Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$ $S_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$ $S_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$ $A_1 = \{ \text{Author}(\text{Diana}) \}$</p>	<p>• universal solutions are exponential in the size of K_1 and M</p> <p>Some $\text{chase}_{\text{MBox}(K_1)}(A_1)$ is not full joining tree.</p>	<p>• universal UCO-solutions are of polynomial size</p> <p>Some $K_2 = (T_2, A_2)$ with $T_2 = \{ \text{Author}(\text{Diana}), \text{Editor}(\text{Elena}) \}$ is a univ. UCO-solution for K_1 under M.</p>

Open Problems and Future Work

Problems that remain open:

- the exact computational complexity of computing (universal) solutions,
- computing a universal solution in presence of disjunctions in the mapping,
- computing a universal UCO-solution,
- computing the minimal mapping M' for a weakly representable T_2 , such that T_2 is representable under M' .

Plans for future work:

- implement the representability algorithm,
- implement a simple prototype for KB exchange,
- study KB exchange for more expressive languages, such as DL-Lite with GRC, DL-Lite_{ind}, and \mathcal{EL} ,
- study composition and inversion of mappings.

Description Logic Knowledge Base Exchange

Elena Botova supervised by Diego Calvanese
KROB Research Centre, Free University of Bozen-Bolzano, Italy
botova@inf.unibz.it

Data Exchange vs. Knowledge Base Exchange

Knowledge Base Exchange is a special case of Data Exchange with incomplete information.

Data Exchange

Each database instance is complete, i.e., every fact is either true or false. Therefore, represents exactly one possible instance (full).

Data Exchange Example

Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$
 $S_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$
 $S_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$
 Then M is a mapping from S_1 to S_2 .
 S_1 is a universal solution for M .

Knowledge Base Exchange

A Description Logic (DL) Knowledge Base consists of

- **concept** dots given as an ABox and
- **structural rules** for completing data given as a TBox, therefore, is a compact representation of (possibly infinitely) many actual instances.

Knowledge Base Exchange Example

Let $T_1 = \{ \langle \text{Author} \sqsubseteq \text{Editor}, \text{Author} \sqsubseteq \text{Editor} \rangle \}$, $A_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$, $T_2 = \{ \langle \text{Author} \sqsubseteq \text{Editor}, \text{Author} \sqsubseteq \text{Editor} \rangle \}$, $A_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$.
 Then A_2 is a universal solution for (T_1, A_1) under M .

We consider logics from the DL-Lite family of lightweight DLs.

Representability Problem

We want to maximize implicit knowledge in the target.

T_2 is **representable** under M if there exists T_1 s.t. for each ABox A_1 , T_2 $\text{chase}_{\text{MBox}(A_1)}$ is a universal UCO-solution for (T_1, A_1) under M , and T_2 is a minimal representation of T_1 under M .

Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$, $S_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$, $S_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$.
 $T_1 = \{ \langle \text{Author} \sqsubseteq \text{Editor}, \text{Author} \sqsubseteq \text{Editor} \rangle \}$ is representable under M if $T_2 = \{ \langle \text{Author} \sqsubseteq \text{Editor}, \text{Author} \sqsubseteq \text{Editor} \rangle \}$ is representable under M .

The representability problem for DL-Lite, TBoxes

T_2 is **representable** under M iff there exists an algorithm to construct universal UCO-solutions of polynomial size. Otherwise, a more related representability can be used.

T_2 is **weakly representable** under M iff there exists T_1 such that $M \subseteq M'$, $M'(T_1) = M'$ and T_1 is representable under M' .

Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$, $S_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$, $S_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$.
 $T_1 = \{ \langle \text{Author} \sqsubseteq \text{Editor}, \text{Author} \sqsubseteq \text{Editor} \rangle \}$ is weakly representable under M since $M' = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$ is representable under M' .

The weak representability problem for DL-Lite, TBoxes

is decidable in PTime.

Computational Problems. Summary of Results

We study the computational complexity of two decision problems associated to each kind of solution.

- The membership problems have as input a mapping M , and source and target KBs K_1 and K_2 (or TBoxes T_1 and T_2).
- The non-emptiness problems have as input a mapping M , and a source KB K_1 (or TBox T_1).

Below are results for DL-Lite, a prominent member of the DL-Lite family.

Membership	Simple ABoxes	Extended ABoxes	Non-emptiness	Simple ABoxes	Extended ABoxes
Universal solutions	in NP	in ExpTime	Universal solutions	in NP	in ExpTime
Universal UCO-solutions	in PSpace-hard	in PSpace-hard	Universal UCO-solutions	in PSpace-hard	in PSpace-hard
UCO-representations	NLogSPACE-complete	NLogSPACE-complete	UCO-representations	NLogSPACE-complete	NLogSPACE-complete

Exchanging Description Logic Knowledge Bases

Marcello Arenas, Elena Botova, Diego Calvanese, and Vladislav Righilov
Dept. of Computer Science, PUC Chile, KROB Research Centre, Free Univ. of Bozen-Bolzano, Italy, Institute of Informatics, Leibniz Univ. Hannover, Germany, and Eindhoven University of Technology, The Netherlands

Our Framework: Knowledge Base Exchange

Given a mapping M and a source knowledge base (KB) K_1 , compute a target KB K_2 that is a solution for K_1 under M .

We consider exchange of Description Logic (DL) KBs: each KB is constituted by a TBox and an ABox, and mapping is a set of DL inclusions.

Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$ be a mapping, K_1 and K_2 KBs over S_1 and S_2 .

- K_1 is a **universal solution** for K_1 under M if $\text{MBox}(K_1) = \text{Sat}_{\text{MBox}(K_1)}$.
- K_2 is a **universal UCO-solution** if $\forall Q \in \mathcal{Q}$, $\text{entails}_M(Q)$ over S_2 .

In this paper we study the problem of KB exchange for DL-Lite and DL-Lite_{ind}.

Why Knowledge Base Exchange

Because we brought show and clutter goods described in an ontology. Goods. We want to open an online shop selling the goods. But first we want to change the information in the ontology for reordering the goods. But first we want to change the ontology vocabulary as it will be deployed to the user.

An ontology over the new vocabulary can be obtained as a solution of the UCO-solution problem for M and (T_1, A_1) .

Universal and UCO-Solutions

There are two types of possible solutions to the KB exchange problem: with the empty TBox and with a non-empty TBox.

The universal solution has the empty TBox and it is the ABox A_1^{univ} of the form:

The universal solution has several drawbacks:

- universal solution for DL-Lite does not always exist.
- if it exists, then its TBox is empty.
- the smallest universal solutions are exponential in the size of the input.

An alternative to the universal solution could be the universal UCO-solution (T_2, A_2) depicted below:

Universal UCO-solutions are the preferred solutions.

Why Universal UCO-Solutions

The website should provide the following functionality:

- describe different categories, e.g. to show all sandals, or sandals on platforms.
- filter the listed products according to several criteria, such as color, brand, etc.
- recommend items that fit the currently viewed product.

Such functionality can be implemented querying the target KB with UCOs:

Universal UCO-solutions are the preferred solutions.

New Problem: Representability of a TBox

To minimize the exchange of ABox information, we are interested in solutions that contain as much implicit knowledge as possible: given a mapping M and a source TBox T_1 , compute a target TBox T_2 of exists, such that for each ABox A_1 , T_2 $\text{chase}_{\text{MBox}(A_1)}$ is a universal UCO-solution for (T_1, A_1) under M .

The problem of UCO-representability for DL-Lite_{ind} TBoxes can be solved in polynomial time.

Description Logic Knowledge Base Exchange

Elena Botova supervised by Diego Calvanese
KROB Research Centre, Free University of Bozen-Bolzano, Italy
botova@inf.unibz.it

Data Exchange vs. Knowledge Base Exchange

Knowledge Base Exchange is a special case of Data Exchange with incomplete information.

Data Exchange

Each database instance is complete, i.e., every fact is either true or false. Therefore, represents exactly one possible instance (full).

Data Exchange Example

Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$
 $S_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$
 $S_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$
 Then M is a mapping from S_1 to S_2 .
 S_1 is a universal solution for M .

Knowledge Base Exchange

A Description Logic (DL) Knowledge Base consists of

- **concept** dots given as an ABox and
- **structural rules** for completing data given as a TBox, therefore, is a compact representation of (possibly infinitely) many actual instances.

Knowledge Base Exchange Example

Let $T_1 = \{ \langle \text{Author} \sqsubseteq \text{Editor}, \text{Author} \sqsubseteq \text{Editor} \rangle \}$, $A_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$, $T_2 = \{ \langle \text{Author} \sqsubseteq \text{Editor}, \text{Author} \sqsubseteq \text{Editor} \rangle \}$, $A_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$.
 Then A_2 is a universal solution for (T_1, A_1) under M .

We consider logics from the DL-Lite family of lightweight DLs.

Representability Problem

We want to maximize implicit knowledge in the target.

T_2 is **representable** under M if there exists T_1 s.t. for each ABox A_1 , T_2 $\text{chase}_{\text{MBox}(A_1)}$ is a universal UCO-solution for (T_1, A_1) under M , and T_2 is a minimal representation of T_1 under M .

Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$, $S_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$, $S_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$.
 $T_1 = \{ \langle \text{Author} \sqsubseteq \text{Editor}, \text{Author} \sqsubseteq \text{Editor} \rangle \}$ is representable under M if $T_2 = \{ \langle \text{Author} \sqsubseteq \text{Editor}, \text{Author} \sqsubseteq \text{Editor} \rangle \}$ is representable under M .

The representability problem for DL-Lite, TBoxes

T_2 is **representable** under M iff there exists an algorithm to construct universal UCO-solutions of polynomial size. Otherwise, a more related representability can be used.

T_2 is **weakly representable** under M iff there exists T_1 such that $M \subseteq M'$, $M'(T_1) = M'$ and T_1 is representable under M' .

Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$, $S_1 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}) \}$, $S_2 = \{ \text{Author}(\text{Diana}), \text{Author}(\text{Elena}), \text{Editor}(\text{Diego}), \text{Editor}(\text{Elena}) \}$.
 $T_1 = \{ \langle \text{Author} \sqsubseteq \text{Editor}, \text{Author} \sqsubseteq \text{Editor} \rangle \}$ is weakly representable under M since $M' = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$ is representable under M' .

The weak representability problem for DL-Lite, TBoxes

is decidable in PTime.

Computational Problems. Summary of Results

We study the computational complexity of two decision problems associated to each kind of solution.

- The membership problems have as input a mapping M , and source and target KBs K_1 and K_2 (or TBoxes T_1 and T_2).
- The non-emptiness problems have as input a mapping M , and a source KB K_1 (or TBox T_1).

Below are results for DL-Lite, a prominent member of the DL-Lite family.

Membership	Simple ABoxes	Extended ABoxes	Non-emptiness	Simple ABoxes	Extended ABoxes
Universal solutions	in NP	in ExpTime	Universal solutions	in NP	in ExpTime
Universal UCO-solutions	in PSpace-hard	in PSpace-hard	Universal UCO-solutions	in PSpace-hard	in PSpace-hard
UCO-representations	NLogSPACE-complete	NLogSPACE-complete	UCO-representations	NLogSPACE-complete	NLogSPACE-complete

Description Logic Knowledge Base Exchange

Marcello Arenas, Elena Botova, and Diego Calvanese
Dept. of Computer Science, PUC Chile, KROB Research Centre, Free Univ. of Bozen-Bolzano, Italy, Institute of Informatics, Leibniz Univ. Hannover, Germany, and Eindhoven University of Technology, The Netherlands

Our Framework: Knowledge Base Exchange

Given a mapping M and a source knowledge base (KB) K_1 , compute a target KB K_2 that is a solution for K_1 under M .

We consider exchange of Description Logic (DL) KBs: each KB is constituted by a TBox and an ABox, and mapping is a set of DL inclusions.

Let $M = \{ \langle \text{Author}, \text{Author} \rangle, \langle \text{Editor}, \text{Editor} \rangle \}$ be a mapping, K_1 and K_2 KBs over S_1 and S_2 .

- K_1 is a **universal solution** for K_1 under M if $\text{MBox}(K_1) = \text{Sat}_{\text{MBox}(K_1)}$.
- K_2 is a **universal UCO-solution** if $\forall Q \in \mathcal{Q}$, $\text{entails}_M(Q)$ over S_2 .

In this paper we study the problem of KB exchange for DL-Lite and DL-Lite_{ind}.

Why Knowledge Base Exchange

Because we brought show and clutter goods described in an ontology. Goods. We want to open an online shop selling the goods. But first we want to change the information in the ontology for reordering the goods. But first we want to change the ontology vocabulary as it will be deployed to the user.

An ontology over the new vocabulary can be obtained as a solution of the UCO-solution problem for M and (T_1, A_1) .

Universal and UCO-Solutions

There are two types of possible solutions to the KB exchange problem: with the empty TBox and with a non-empty TBox.

The universal solution has the empty TBox and it is the ABox A_1^{univ} of the form:

The universal solution has several drawbacks:

- universal solution for DL-Lite does not always exist.
- if it exists, then its TBox is empty.
- the smallest universal solutions are exponential in the size of the input.

An alternative to the universal solution could be the universal UCO-solution (T_2, A_2) depicted below:

Universal UCO-solutions are the preferred solutions.

Why Universal UCO-Solutions

The website should provide the following functionality:

- describe different categories, e.g. to show all sandals, or sandals on platforms.
- filter the listed products according to several criteria, such as color, brand, etc.
- recommend items that fit the currently viewed product.

Such functionality can be implemented querying the target KB with UCOs:

Universal UCO-solutions are the preferred solutions.

New Problem: Representability of a TBox

To minimize the exchange of ABox information, we are interested in solutions that contain as much implicit knowledge as possible: given a mapping M and a source TBox T_1 , compute a target TBox T_2 of exists, such that for each ABox A_1 , T_2 $\text{chase}_{\text{MBox}(A_1)}$ is a universal UCO-solution for (T_1, A_1) under M .

The problem of UCO-representability for DL-Lite_{ind} TBoxes can be solved in polynomial time.

[1] M. Arenas, E. Botova, and D. Calvanese. Knowledge base exchange. In *Proc. of DL 2011*, volume 745 of CEUR, ceur-ws.org, 2011.

[2] M. Arenas, E. Botova, D. Calvanese, V. Righilov, and E. Sherstov. Exchanging description logic knowledge bases. In *Proc. of KR 2012*, 2012.

[3] M. Arenas, E. Botova, D. Calvanese, V. Righilov, and E. Sherstov. Representability in dl-lite knowledge base exchange. In *Proc. of the 25th Int. Workshop on Description Logics (DL 2012)*, 2012.

[4] E. Botova. Description logic knowledge base exchange. In *Proc. of the 6th International Conference on Web Reasoning and Rule Systems (RR 2012)*, pages 266–271, 2012.

On Flexibility

While, I agree, it is very useful to have a structured poster with all the functionality concerning columns, automatic coordinates, width and so on, I believe that the user should have the possibility to have an unstructured poster, for instance like this one



Department of Informatics and Telecommunications
University of Athens, Greece

Thodoris Georgiou¹
t.georgiou@di.uoa.gr

Undergraduate Poster
SIGMOD Conference 2010



Extracting Topics of Debate between Users on Web Discussion Boards

1

Problem definition

Extraction of the topics users discuss about: Given the small size of posts and the casual style of the language used, it is hard to indicate topics from the information of a post solely.

Our approach

A clustering algorithm that creates sets of posts sharing same thematic context.

2

Thread posts

#1
#2
#3
#4
#5
#6

Time

Post example

Utilizing the quotation system, timestamps and the position of each post in the thread, post connections are recovered.

These connections can either be quotes or direct replies.

A typical forum post shows the quoted text, the username of the post author, the timestamp etc.

Direct reply is a post that replies to the exactly previous post, in a very short time and without using a quotation.

3

For every post (#i) of the thread we obtain its neighboring nodes on the graph G:

Start from node i

Add all nodes connected with i, in set N

For every node in set N, as j:

For every node connected with j, as k:

If the total weight of the edges between i and k is more than w:

Note: w is not fixed

Continue

Else:

Add k in the end of set N

Set all visited nodes as neighbors of i

4

In the current state every post has been paired with its neighbors and topic terms.

Set contains all neighbors of post #i

Author of post #i

Terms

Topic is extracted using terms from this cluster of posts.

Topics are represented as sets of topic terms (-10).

5

Topics are extracted for every post and linked with the corresponding users.

Users sharing many common terms have discussed about the same topic.

6

Future work

Knowledge of user agreement can be used to estimate opinion relations on the extracted topics.

It has been observed that when a user replies to another, it will most certainly be due to disagreement!

ARE YOU COMING TO BED?

I CAN'T. THIS IS IMPORTANT. SOMEBODY IS ARGUING ON THE INTERNET.

www.AKCE.com

Thread Graph G

Vertices: posts
Edges: connections

Weighted using two metrics:

- Post distance
- Time distance

Post distance: The distance in nodes between two connected posts.

Time distance: The distance in minutes between two connected posts.



Web discussion board (forum)

All forum data are downloaded with a web crawler...

...and an HTML tag cleaning is applied.

Database

Web discussion board (forum)

AlchemyAPI Online term extraction service

TFIDF

Post text

Post terms

Acknowledgments: I am grateful to my colleague *Manos Karvounis¹* and my advisor *Yannis Ioannidis¹* for their precious help.