Visual Comparison of Graphical Models

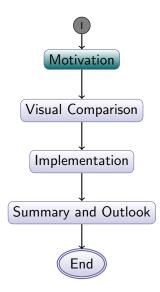
Arne Schipper Hauke Fuhrmann Reinhard von Hanxleden

Real-Time Systems and Embedded Systems Group, Department of Computer Science, Christian-Albrechts-Universität zu Kiel

{ars,haf,rvh}@informatik.uni-kiel.de

UML&AADL'09

Outline

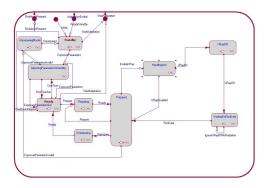


Problem:

 Graphical models very often used, quite easy to create and browse, but pain to compare.

Problem:

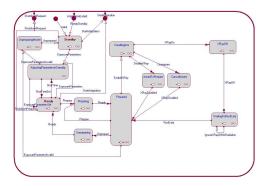
 Graphical models very often used, quite easy to create and browse, but pain to compare.



Screenshot of a real model from a project, Version 1

Problem:

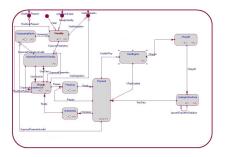
 Graphical models very often used, quite easy to create and browse, but pain to compare.



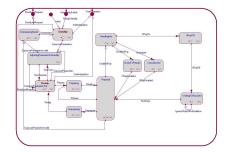
Screenshot of a real model from a project, Version 2

Problem:

 Graphical models very often used, quite easy to create and browse, but pain to compare.



Screenshot of a real model from a project, Version 1



Screenshot of a real model from a project, Version 2

Problem cont'd:

- ▶ Means exist to compare graphical models textually, but ...
- User has to switch between different abstraction levels.

Problem cont'd:

- ▶ Means exist to compare graphical models textually, but ...
- User has to switch between different abstraction levels.

Solution:

- Develop means to aid the user in performing a real visual comparison of graphical models.
- Some tools exist, but have drawbacks.

Problem cont'd:

- Means exist to compare graphical models textually, but ...
- User has to switch between different abstraction levels.

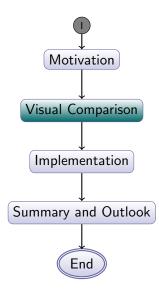
Solution:

- Develop means to aid the user in performing a real visual comparison of graphical models.
- Some tools exist, but have drawbacks.

Method:

- Identify and improve those drawbacks.
- Implement as Eclipse plug-in using existing techniques where appropriate.
- Use generic approach to cope with various graphical languages.

Outline



Textual diff/comparison:

- ► Known to everyone, compare texts side by side.
- One-dimensional or linear arrangement with *holes* in the texts.

Textual diff/comparison:

- Known to everyone, compare texts side by side.
- One-dimensional or linear arrangement with *holes* in the texts.

fertility rate]] is relatively low and stood at 1.96 in 2009, comparable to those of the [[United States]] and [[France]].ref>{{cic web} [tite=Total fertility rate | url=https://www.cia.gov/library /publications/the-world-factbook/rankorder /2127rank.html | work=The World Factbook -Country Comparisons | publisher=CIA| year=2009 | accessdate=2009-05-01}) The country maintains a high [[List of countries by literacy rate][iteracy rate]] of 99%, comparable to most [[developed countries]].ref>Set [[List of countries by literacy rate]]. [[Total fertility rate]] is relatively low and stood at 1.96 in 2009, comparable to those of the [[United States]] and [[France]]-ref>{{cite web | itile=Total fertility rate | url=https://www.cia.gov /library/publications/the-world-factbook/rankorder /2127rank.html | work=The World Factbook --Country Comparisons | publisher=ClA | year=2009 | accessdate=2009-05-01] ></ref>

	Lar	igua	age=	
--	-----	------	------	--

Line 424:

[[File:An3 man.jpg]thumb|right|A drawing in one of the chambers of the [[Complex of Goguryeo Tombs|Goguryeo tombs]].]]

===Language===

Line 425:

[[File:An3 man.jpg]thumb|right|A drawing in one of the chambers of the [[Complex of Goguryeo Tombs|Goguryeo tombs]].]]

Figure: Two article versions in Wikipedia

Common comparison of graphical models:

- Generate a textual description of the changes.
- Is sometimes structured, but ...
- User has to find these changes in the graphical representation.

Common comparison of graphical models:

- Generate a textual description of the changes.
- Is sometimes structured, but ...
- User has to find these changes in the graphical representation.

	elect the components to comp Follow selection from Content of P					
	Follow selection from Concern or P Select file to compare mensalty	royec	Expert	ontro		
				Gradit		
L	C/Mijden.red		(100.)			
Ø	Use selected version of project					
	Select file to compare manually					
	C:Wijden2.mdl	(10.00)	<u>س</u>			
				Diff Nov4		
isuit Vi	ener Mode	н		Nodel 2		
. 0	don		4 (9) don2			
1	OurrentScope	n	OurrentScope			
4 - DC-Motor		=	a - DC-Notor	E		
	-(1) TU(1)	ш				
	- (c) (c)		- 💿 Ur(t)			
	i dectrical part of DCM		electrical part of E	CM .		
	 Implementation part of DCM 		 Internet of Depart of Depart of Departs 	м		
	- (0) TL(R)		- 00 TL(2)			
			-0 TD(r)			
	 Eriction characteristic 		- Constant			
	Integrator		Priction chara	taristic.		
	KI		Integrator			
	Sum of Torques		5			
	b do(1.Theta)	÷	- Sum of Torma			
< 1.00	· · · · · · · · · · · · · · · · · · ·		< _m_	*		
Show Diffs only			Filter Diffs 🛄 Show selection	In MATLAS		
Type	Element Path	Description				
-	(dcm/DC-Motor/electrical par	The Models have different values (Gain) in Bloc				
-	(dcn/DC-Motor mechan part	The Models have different values (InputValues)				
-	(donuDC-Motor/mechanipart	The	Models have different values (Initia	s/Conditi		
-	(dcn/DC-Motor/mechan part		Block 'Integrator' has different pre-			
1.	(dcn/DC-Motor/mechan part	The Models have different Positions for Block X11				
<i>.</i>	(dcn/DC-Motor/mechan part	The	 Models have different Positions for 	Block E		
1*1	(dongMun1		Models have different Foreground			
<	(dcm2)DC-Motor/mechan par		de 'Constant' doesn't exist in Model I			
<	(dcm2)DC-Motor/mechan.par		e between Blocks 'Constant' and 'Int			

Figure: Model diff of Expert Control

Visual comparison:

- Show the changes in the graphical model itself.
- Prevents the user from switching between text and graphical model.

Visual comparison:

- Show the changes in the graphical model itself.
- Prevents the user from switching between text and graphical model.

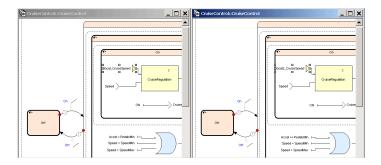


Figure: Scade model diff

Challenges:

Graphical models at least two-dimensional, in contrast to text.

- Graphical models at least two-dimensional, in contrast to text.
- ▶ No trivial solution for *holes* like in textual diff.

- Graphical models at least two-dimensional, in contrast to text.
- ▶ No trivial solution for *holes* like in textual diff.
- Some models have information which is not shown visually.

- Graphical models at least two-dimensional, in contrast to text.
- ▶ No trivial solution for *holes* like in textual diff.
- Some models have information which is not shown visually.
- Large models.

- Graphical models at least two-dimensional, in contrast to text.
- ▶ No trivial solution for *holes* like in textual diff.
- Some models have information which is not shown visually.
- Large models.
- Mental map of the user.

Challenges:

- Graphical models at least two-dimensional, in contrast to text.
- ▶ No trivial solution for *holes* like in textual diff.
- Some models have information which is not shown visually.
- Large models.
- Mental map of the user.
- Difference detection. However, solved by an existing engine and we focus on graphical presentation

Challenges:

- Graphical models at least two-dimensional, in contrast to text.
- ▶ No trivial solution for *holes* like in textual diff.
- Some models have information which is not shown visually.
- Large models.
- Mental map of the user.
- Difference detection. However, solved by an existing engine and we focus on graphical presentation

Questions:

Use just the structure of the graphical model or also the layout information of the elements?

Challenges:

- Graphical models at least two-dimensional, in contrast to text.
- ▶ No trivial solution for *holes* like in textual diff.
- Some models have information which is not shown visually.
- Large models.
- Mental map of the user.
- Difference detection. However, solved by an existing engine and we focus on graphical presentation

- Use just the structure of the graphical model or also the layout information of the elements?
- Use one model or both versions to display the changes?

Challenges:

- Graphical models at least two-dimensional, in contrast to text.
- ▶ No trivial solution for *holes* like in textual diff.
- Some models have information which is not shown visually.
- Large models.
- Mental map of the user.
- Difference detection. However, solved by an existing engine and we focus on graphical presentation

- Use just the structure of the graphical model or also the layout information of the elements?
- Use one model or both versions to display the changes?
- Alter the layout or leave it intact?

Challenges:

- Graphical models at least two-dimensional, in contrast to text.
- ▶ No trivial solution for *holes* like in textual diff.
- Some models have information which is not shown visually.
- Large models.
- Mental map of the user.
- Difference detection. However, solved by an existing engine and we focus on graphical presentation

- Use just the structure of the graphical model or also the layout information of the elements?
- Use one model or both versions to display the changes?
- Alter the layout or leave it intact?
- Does a readable automatic layout help?

The two versions of the model:

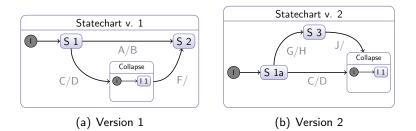


Figure: The two original versions of the example diagram.

Possible representation of the changes 1:

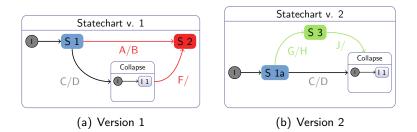


Figure: Plain visual diff. Color legend: green/additions, red/deletions, blue/changes.

Possible representation of the changes 3:

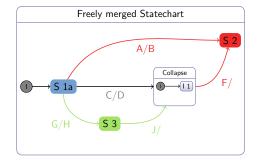


Figure: Freely merged visual diff.

Which representation?

▶ Manual tests showed that *plain* visual diff is best.

- ▶ Manual tests showed that *plain* visual diff is best.
- Additional textual description of changes is also given.

- Manual tests showed that *plain* visual diff is best.
- ► Additional textual description of changes is also given.
- No problems with/recomputation of layout, but a good layout of the original models is helpful.

- Manual tests showed that *plain* visual diff is best.
- ► Additional textual description of changes is also given.
- No problems with/recomputation of layout, but a good layout of the original models is helpful.
- Mental map of user is preserved.

- Manual tests showed that *plain* visual diff is best.
- ► Additional textual description of changes is also given.
- No problems with/recomputation of layout, but a good layout of the original models is helpful.
- Mental map of user is preserved.
- Additional means like panning, zooming and folding needed to cope with large models.

Which representation?

- Manual tests showed that *plain* visual diff is best.
- ► Additional textual description of changes is also given.
- No problems with/recomputation of layout, but a good layout of the original models is helpful.
- Mental map of user is preserved.
- Additional means like panning, zooming and folding needed to cope with large models.

Other issues:

The diff is performed just against the structural/domain model.

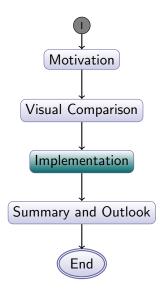
Which representation?

- Manual tests showed that *plain* visual diff is best.
- ► Additional textual description of changes is also given.
- No problems with/recomputation of layout, but a good layout of the original models is helpful.
- Mental map of user is preserved.
- Additional means like panning, zooming and folding needed to cope with large models.

Other issues:

- The diff is performed just against the structural/domain model.
- Non graphical changes (e.g. of properties) are also displayed; blue in the previous slides.

Outline



Developed as Eclipse plug-in within a project called KIELER (Kiel Integrated Environment for Layout, for Eclipse RCP).

• EMF to create the domain models.

- EMF to create the domain models.
- GMF to build the corresponding graphical editor.

- EMF to create the domain models.
- GMF to build the corresponding graphical editor.
- EMF Compare to compute the differences of the EMF model.

- EMF to create the domain models.
- GMF to build the corresponding graphical editor.
- EMF Compare to compute the differences of the EMF model.
- KiViK (Kieler Visual Comparison) to get EMF Compare output into GMF.

- EMF to create the domain models.
- GMF to build the corresponding graphical editor.
- EMF Compare to compute the differences of the EMF model.
- KiViK (Kieler Visual Comparison) to get EMF Compare output into GMF.
 - Use original layout of diagrams and display them side by side.

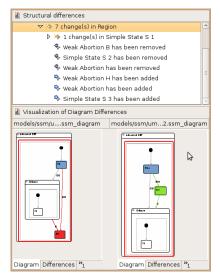
- EMF to create the domain models.
- GMF to build the corresponding graphical editor.
- EMF Compare to compute the differences of the EMF model.
- KiViK (Kieler Visual Comparison) to get EMF Compare output into GMF.
 - Use original layout of diagrams and display them side by side.
 - Annotate the structural changes with different colors.

- EMF to create the domain models.
- GMF to build the corresponding graphical editor.
- EMF Compare to compute the differences of the EMF model.
- KiViK (Kieler Visual Comparison) to get EMF Compare output into GMF.
 - Use original layout of diagrams and display them side by side.
 - Annotate the structural changes with different colors.
 - Use third panel on top to display just the structural changes textually (like EMF Compare).

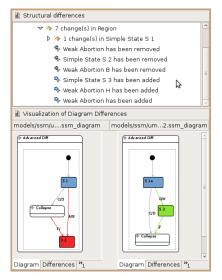
- EMF to create the domain models.
- GMF to build the corresponding graphical editor.
- EMF Compare to compute the differences of the EMF model.
- KiViK (Kieler Visual Comparison) to get EMF Compare output into GMF.
 - Use original layout of diagrams and display them side by side.
 - Annotate the structural changes with different colors.
 - Use third panel on top to display just the structural changes textually (like EMF Compare).
 - Equip the comparison view with means to navigate and zoom.

- EMF to create the domain models.
- GMF to build the corresponding graphical editor.
- EMF Compare to compute the differences of the EMF model.
- KiViK (Kieler Visual Comparison) to get EMF Compare output into GMF.
 - Use original layout of diagrams and display them side by side.
 - Annotate the structural changes with different colors.
 - Use third panel on top to display just the structural changes textually (like EMF Compare).
 - Equip the comparison view with means to navigate and zoom.
 - Collapse composite items with no changes inside (a layout algorithm is needed then).

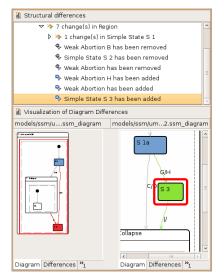
General implementation:



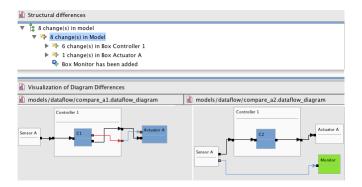
Example of collapsing:



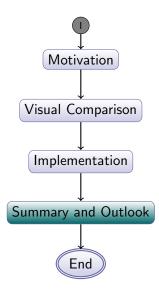
Example of automatic zoom:



Comparison of Dataflow models:



Outline



Summary and Outlook

Feedback:

- Students and professionals gave an overall positive feeback for this approach.
- Representation directly in the diagram seen as benefit.
- Visualization of small (or invisible) changes very useful.
- User interface with collapsing, panning and zooming intuitive.
- Generic approach enables support for various diagrams with none or little adaption.

Summary and Outlook

Outlook:

- Large models are still challenging; time for comparison as well as navigation.
- Next step would be to support merging graphically.
- Maybe implement also the other approaches presented to see how they perform.



Visual Comparison of Graphical Models



Visual Comparison of Graphical Models Thanks!