Master Essay

Eirik Halvard Sæther

February 4, 2020

Contents

Chapter 1

Notes

1.1 Reference examples

- Precise Version Control of Trees with Line-Based Version Control Systems (Asenov et al. 2017)
- A three-way merge for XML documents (Lindholm 2004). See section 2.7.1
- Fine-grained and accurate source code differencing (Falleri et al. 2014)
- Using versioned trees, change detection and node identity for three-way XML merging (Thao and Munson 2019)
- Generic Diff3 for algebraic datatypes (Vassena 2016)

Chapter 2

Introduction

Introducing feature models. what they are, why they are useful. One part is allowing different collaborators. Important for autonomy, etc.

Something about diff, three-way merge and working on line granularity. Very inaccurate, not adequate. maybe a small example.

Introduce a method of merging two feature models derived from the same feature model using a three-way merge algorithm inspired from the text-granularity algorithm and other three-way merge algorithms working on tree-structures like XML and ASTs.

These solutions are not adequate, because we want to create an algorithm that ensures soundness and takes the semantics of feature models into account when merging. This task is not trivial, and the use cases of the developers will need to be considered when deciding which conflicts to automatically merge, warn the user, etc. The algorithm will utilize properties about feature models when merging, like the fact that the nodes are unordered, and each node is uniquely identified by their ID.

This task includes defining the semantics of feature models, defining what a sound model is. The three-way merge algorithm will assume three sound feature models as input, and compute the edit script between the base and the two derived feature models. The two edit scripts will then be combined to a graph that represents what changes to be done to the base, including possible conflicts and dependencies.

2.1 Motivation

TODO

2.2 Objective (with solution requirements)

TODO

2.3 Difficulties/problems/challenges

TODO

2.4 Say why other solutions are not adequate (very short)

TODO

2.5 Describe the solution (very short)

TODO

2.6 Roadmap of the thesis

TODO

2.7 Related Work

2.7.1 A Three-way Merge for XML Documents

Lindholm describes a method of merging XML documents. The algorithm is a three-way merge taking a base XML file, T_0, and two files T_1 and T_2, derived and changed from the base file independently. The XML documents are ordered, labeled trees, and merging the trees takes this into consideration. The algorithm relies on computing *edit scripts* between T_0 and T', where T' is either T_1 or T_2. The edit scripts consist of predefined operations, which are attribute updates and insert, move and delete for XML elements.

One of the important factors of the data structure, is that the nodes are labeled with unique IDs. This makes merging significantly easier.

To study different use-cases (user stories) of XML merging, they manually merged some examples, to see how real world users would handle some tricky cases. Some rules for merging were devised, including:

-

Bibliography

Asenov, Dimitar et al. (2017). "Precise Version Control of Trees with Line-Based Version Control Systems". In: *Fundamental Approaches to Software Engineering - 20th International Conference, FASE 2017, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2017, Uppsala, Sweden, April 22-29, 2017, Proceedings*, pp. 152–169. DOI: 10.1007/978-3-662-54494-5_9. URL: https://doi.org/10.1007/978-3-662-54494-5%5C_9.

Falleri, Jean-Rémy et al. (2014). "Fine-grained and accurate source code differencing". In: *ACM/IEEE International Conference on Automated Software Engineering, ASE '14, Vasteras, Sweden - September 15 - 19, 2014*, pp. 313–324. DOI: 10.1145/2642937.2642982. URL: https://doi.org/10.1145/2642937.2642982.

Lindholm, Tancred (2004). "A three-way merge for XML documents". In: *Proceedings of the 2004 ACM Symposium on Document Engineering, Milwaukee, Wisconsin, USA, October 28-30, 2004*, pp. 1–10. DOI: 10.1145/1030397.1030399. URL: https://doi.org/10.1145/1030397.1030399.

Thao, Cheng and Ethan V. Munson (2019). "Using versioned trees, change detection and node identity for three-way XML merging". In: *Computer Science - R&D* 34.1, pp. 3–16. DOI: 10.1007/s00450-013-0253-5. URL: https://doi.org/10.1007/s00450-013-0253-5.

Vassena, Marco (2016). "Generic Diff3 for algebraic datatypes". In: *Proceedings of the 1st International Workshop on Type-Driven Development, TyDe@ICFP 2016, Nara, Japan, September 18, 2016*, pp. 62–71. DOI: 10.1145/2976022.2976026. URL: https://doi.org/10.1145/2976022.2976026.