Intelligent Techniques for the maintenance of constraint-based configuration systems

Florian Reinfrank, Gerald Ninaus, Franz Wotawa, Alexander Felfernig

{firstname.lastname}@ist.tugraz.at

Institute for Software Technology

Graz University of Technology

Inffeldgasse 16b, 8010 Graz, Austria

Outline

- Introduction
- Example
- Anomaly management: well-formedness violations
- Summary

Introduction

- Constraint-based configuration knowledge bases are changing over time
- Maintenance is time consuming and error prone
  - Detect anomalies
  - Understand anomalies
  - Repair anomalies
- Different techniques to reduce the maintenance effort
  - Recommendation
  - **Anomaly management**
  - Simulation
  - KB evaluation

Example: bike configuration

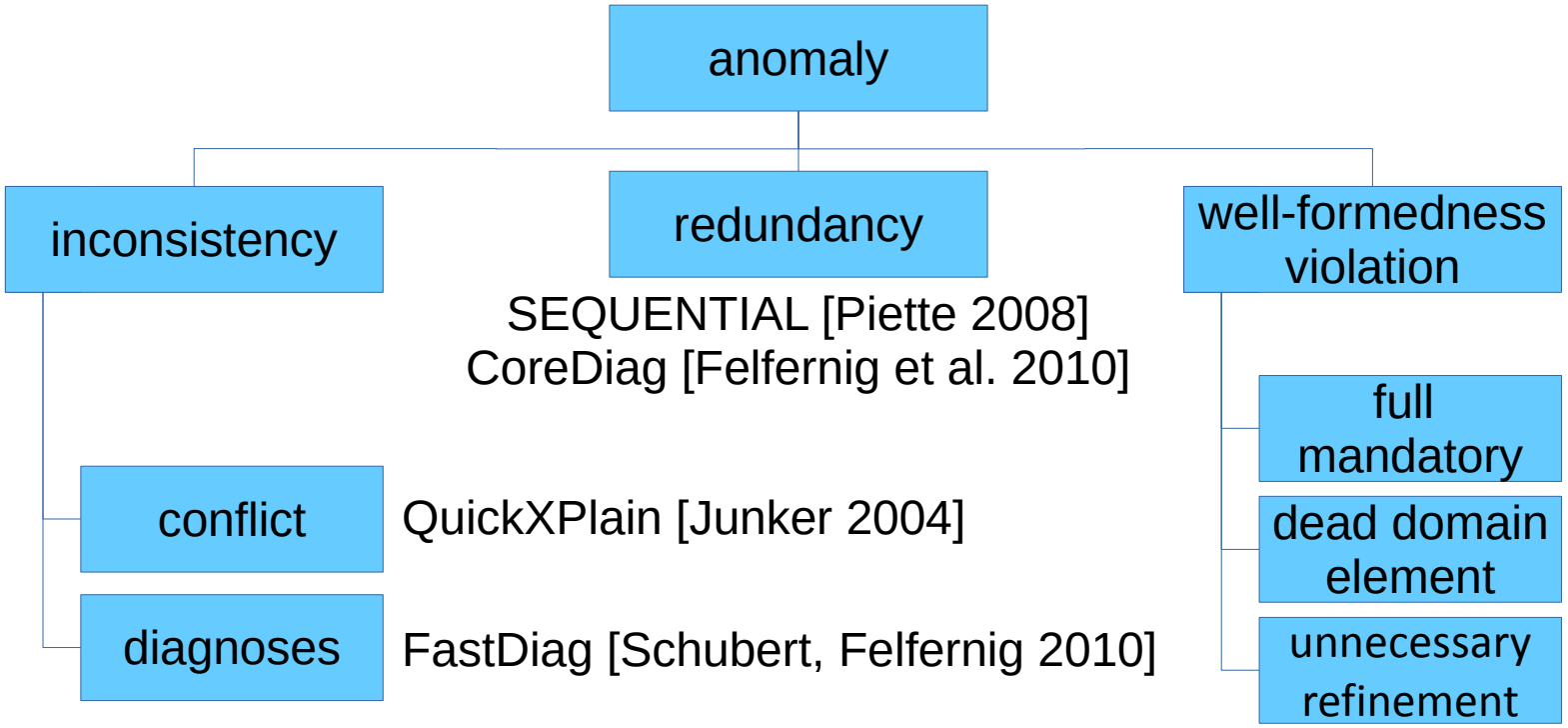
$$V = \{ BikeType, FrameSize, eBike, TireWidth, UniCycle, Usage \}$$

$$D = \{ dom(BikeType) = \{ MountainBike, CityBike, RacerBike \},$$
$$dom(FrameSize) = \{ 40\,cm, 50\,cm, 60\,cm \}, dom(eBike) = \{ true, false \},$$
$$dom(TireWidth) = \{ 23\,mm, 37\,mm, 57\,mm \}, dom(UniCycle) = \{ true, false \},$$
$$dom(Usage) = \{ Competition, EverydayLife, HillClimbing \} \}$$

$$C_{KB} = \{ c_0 = BikeType = MountainBike \rightarrow TireWidth > 37\,mm \wedge FrameSize \geq 50\,cm$$
$$c_1 = BikeType = RacerBike \rightarrow TireWidth = 23\,mm \wedge FrameSize \geq 60\,cm$$
$$c_2 = BikeType = CityBike \rightarrow TireWidth = 37\,mm \wedge FrameSize \geq 50\,cm$$
$$c_3 = \neg (BikeType \neq CityBike \wedge eBike = true)$$
$$c_4 = Usage = EveryDayLife \rightarrow BikeType = CityBike$$
$$c_5 = Usage = HillClimbing \rightarrow BikeType = MountainBike$$
$$c_6 = Usage = Competition \rightarrow BikeType = RacerBike \wedge FrameSize = 60\,cm$$
$$c_7 = eBike = true \rightarrow TireWidth = 37\,mm$$
$$c_8 = UniCycle = false \}$$

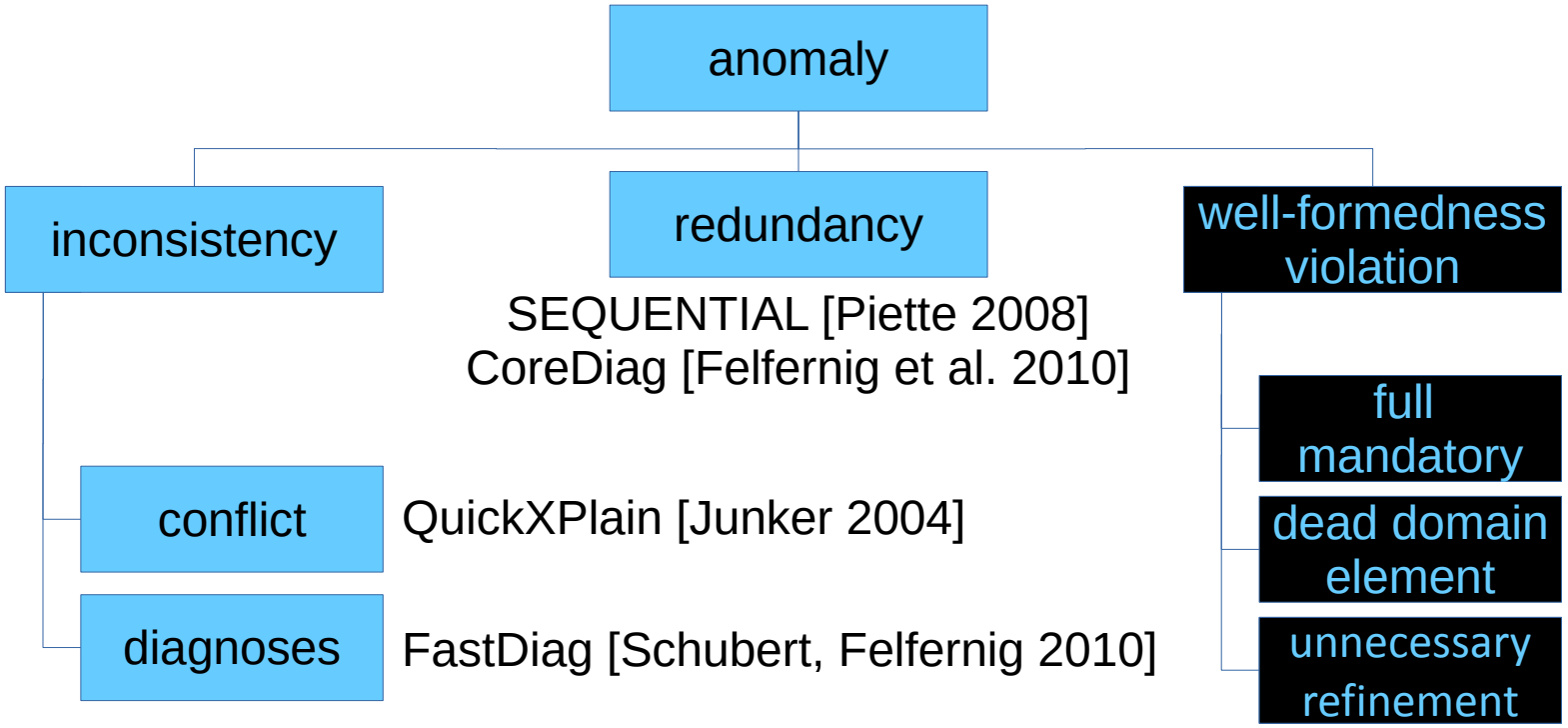
Anomaly management

- Anomalies are patterns in data that do not conform to a well defined notion of normal behaviour [Chandola et al. 2009]
- Management = detection + explanation + repair



Anomaly management

- Anomalies are patterns in data that do not conform to a well defined notion of normal behaviour [Chandola et al. 2009]
- Management = detection + explanation + repair



Anomaly management full mandatory (1)

- A domain element $val_1 \in dom(v_i)$ is full mandatory iff there is no consistent (complete or incomplete) instance without $C_{KB} \cup \{v_i = val_1\}$, s.t. $C_{KB} \cup \{v_i \neq val_1\}$ is inconsistent
- Explanation: $QuickXPlain(\{v_i \neq val_1\}, C_{KB})$
- Repair actions:
  - Remove other domain elements?
  - Remove variable?

Anomaly management full mandatory (2)

$$V = \{ BikeType, FrameSize, eBike, TireWidth, UniCycle, Usage \}$$

$$D = \{ dom(BikeType) = \{ MountainBike, CityBike, RacerBike \},$$
$$dom(FrameSize) = \{ 40\,cm, 50\,cm, 60\,cm \}, dom(eBike) = \{ true, false \},$$
$$dom(TireWidth) = \{ 23\,mm, 37\,mm, 57\,mm \}, dom(UniCycle) = \{ true, false \},$$
$$dom(Usage) = \{ Competition, EverydayLife, HillClimbing \} \}$$

$$C_{KB} = \{ c_0 = BikeType = MountainBike \rightarrow TireWidth > 37\,mm \wedge FrameSize \geq 50\,cm$$
$$c_1 = BikeType = RacerBike \rightarrow TireWidth = 23\,mm \wedge FrameSize \geq 60\,cm$$
$$c_2 = BikeType = CityBike \rightarrow TireWidth = 37\,mm \wedge FrameSize \geq 50\,cm$$
$$c_3 = \neg (BikeType \neq CityBike \wedge eBike = true)$$
$$c_4 = Usage = EveryDayLife \rightarrow BikeType = CityBike$$
$$c_5 = Usage = HillClimbing \rightarrow BikeType = MountainBike$$
$$c_6 = Usage = Competition \rightarrow BikeType = RacerBike \wedge FrameSize = 60\,cm$$
$$c_7 = eBike = true \rightarrow TireWidth = 37\,mm$$
$$c_8 = UniCycle = false \}$$

Anomaly management dead domain elements (1)

- A domain element $val \in dom(v)$ is dead iff it is never in a consistent instance, s.t. $C_{KB} \cup \{v = val\}$ is inconsistent.
- Explanation: $QuickXPlain(\{v = val\}, C_{KB})$
- Repair action: remove the domain element

Anomaly management dead domain elements (2)

$$V = \{ BikeType, FrameSize, eBike, TireWidth, UniCycle, Usage \}$$

$$D = \{ dom(BikeType) = \{ MountainBike, CityBike, RacerBike \},$$
$$dom(FrameSize) = \{ 40\,cm, 50\,cm, 60\,cm \}, dom(eBike) = \{ true, false \},$$
$$dom(TireWidth) = \{ 23\,mm, 37\,mm, 57\,mm \}, dom(UniCycle) = \{ true, false \},$$
$$dom(Usage) = \{ Competition, EverydayLife, HillClimbing \} \}$$

$$C_{KB} = \{ c_0 = BikeType = MountainBike \rightarrow TireWidth > 37\,mm \wedge FrameSize \geq 50\,cm$$
$$c_1 = BikeType = RacerBike \rightarrow TireWidth = 23\,mm \wedge FrameSize \geq 60\,cm$$
$$c_2 = BikeType = CityBike \rightarrow TireWidth = 37\,mm \wedge FrameSize \geq 50\,cm$$
$$c_3 = \neg (BikeType \neq CityBike \wedge eBike = true)$$
$$c_4 = Usage = EveryDayLife \rightarrow BikeType = CityBike$$
$$c_5 = Usage = HillClimbing \rightarrow BikeType = MountainBike$$
$$c_6 = Usage = Competition \rightarrow BikeType = RacerBike \wedge FrameSize = 60\,cm$$
$$c_7 = eBike = true \rightarrow TireWidth = 37\,mm$$
$$c_8 = UniCycle = false \}$$

Anomaly management unnecessary refinement (1)

- A knowledge base contains a variable pair v_i, v_j. For each domain element val_1 of variable v_i, we can say that variable v_j always has the same assignment $v_j = val_2$, s.t. $\forall_{val_1 \in dom(v_i)} \exists_{val_2 \in dom(v_j)} v_i = val_1 \wedge v_j \neq val_2$ is inconsistent.
- Explanation: $\forall_{val_1 \in dom(v_i)} \exists_{val_2 \in dom(v_j)}$
$$QuickXPlain\left(\left\{v_i = val_i \wedge v_j \neq val_2\right\}, C_{KB}\right)$$
- Repair actions: remove the variable v_i and change the name of variable v_j

Anomaly management unnecessary refinement (2)

$$V = \{ BikeType, FrameSize, eBike, TireWidth, UniCycle, Usage \}$$

$$D = \{ dom(BikeType) = \{ MountainBike, CityBike, RacerBike \},$$
$$dom(FrameSize) = \{ 40\,cm, 50\,cm, 60\,cm \}, dom(eBike) = \{ true, false \},$$
$$dom(TireWidth) = \{ 23\,mm, 37\,mm, 57\,mm \}, dom(UniCycle) = \{ true, false \},$$
$$dom(Usage) = \{ Competition, EverydayLife, HillClimbing \} \}$$

$$C_{KB} = \{ c_0 = BikeType = MountainBike \rightarrow TireWidth > 37\,mm \wedge FrameSize \geq 50\,cm$$
$$c_1 = BikeType = RacerBike \rightarrow TireWidth = 23\,mm \wedge FrameSize \geq 60\,cm$$
$$c_2 = BikeType = CityBike \rightarrow TireWidth = 37\,mm \wedge FrameSize \geq 50\,cm$$
$$c_3 = \neg (BikeType \neq CityBike \wedge eBike = true)$$
$$c_4 = Usage = EveryDayLife \rightarrow BikeType = CityBike$$
$$c_5 = Usage = HillClimbing \rightarrow BikeType = MountainBike$$
$$c_6 = Usage = Competition \rightarrow BikeType = RacerBike \wedge FrameSize = 60\,cm$$
$$c_7 = eBike = true \rightarrow TireWidth = 37\,mm$$
$$c_8 = UniCycle = false \}$$

Summary

- Find well-formedness violations in knowledge bases: delete not relevant variables / domain elements
- Explanations make the knowledge base more understandable
- Repair actions make the knowledge base more maintainable
- Repair actions reduce the time for consistency checks

References

- [Chandola et al. 2009] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. ACM Comput. Surv., 41:15:1–15:58. 2009.
- [Felfernig et al. 2010] Alexander Felfernig, Christoph Zehentner, and Paul Blazek. Corediag: Eliminating redundancy in constraint sets. In Martin Sachenbacher, Oskar Dressler, and Michael Hofbaur, editors, DX 2011. 22nd International Workshop on Principles of Diagnosis, pages 219 – 224, Murnau, GER, 2010.
- [Junker 2004] Ulrich Junker. Quickxplain: preferred explanations and relaxations for over-constrained problems. In Proceedings of the 19th national conference on Artifical intelligence, AAAI'04, pages 167–172. AAAI Press, 2004.
- [Piette 2008] Cedric Piette. Let the solver deal with redundancy. In Proceedings of the 2008 20th IEEE International Conference on Tools with Artificial Intelligence - Volume 01, pages 67–73, Washington, DC, USA, 2008. IEEE Computer Society.
- [Schubert, Felfernig 2010] Monika Schubert and Alexander Felfernig. A Diagnosis Algorithm for Inconsistent Constraint Sets. In Proceedings of the 21st International Workshop on the Principles of Diagnosis. 2010.