

Well-formed Default Unification in Non-deterministic Multiple Inheritance Hierarchies

Christian Schulz, Jan Alexandersson and Tilman Becker
DFKI, Saarbrücken

1 Introduction

Default unification represents a fundamental extension of the common unification operation, where even in case of inconsistency between the information to be merged, there will always be a result. As default unification is no longer a commutative operation, a distinction is made between the argument containing default information, the *background*, and the other argument consisting of non-default information, the *cover*.

An elegant point of departure to formalizing default unification has been delivered by Carpenter's *credulous default unification* (CDU) (Carpenter, 1993):

$$F \sqsubset_c G = \{F \sqcup G' \mid G' \sqsubseteq G \text{ is maximal such that } F \sqcup G' \text{ is defined}\}$$

Unlike previous strategies to replace conflicting parts by the non-default information (Shieber, 1986; Kaplan, 1987; Bouma, 1990), Carpenter suggests in case of clashes to relax the specificity in the defeasible part to achieve consistency with the non-default structure. Since the detection of the generalized default might not be deterministic, multiple results may arise.

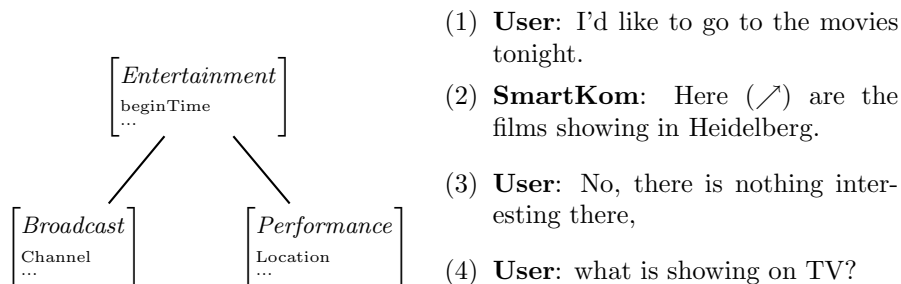


Figure 1: An excerpt from the SMARTKOM ontology and the multi-modal interaction of a dialog between the user and the system. ↗ denotes the pointing gesture by the presentation agent.

A CDU formalization in (Alexandersson & Becker, 2007) is used in the SMARTKOM system (Wahlster, 2006) on the basis of typed feature structures (TFS) and CDU as the knowledge representation and inference mechanism, in order to monitor the discourse processing, where the user’s input, the cover, updates the internal dialogue state, the background, see figure 1. The second crucial point of departure is how Alexandersson and Becker understand to exploit the information inherent to an inheritance hierarchy and henceforward to transfer Carpenter’s notion to a very effective and sound way to specify CDU, which stands in contrast to the formalization by (Grover, Brew, Manandhar, & Moens, 1994)¹ that suffers from theoretical and computational drawbacks.

This Work (Schulz, 2007) develops an extension to the CDU operation presented in (Alexandersson & Becker, 2007) for which the CDU algorithm is adapted to deal with multiple inheritance type hierarchies omitting the requirement to be bounded complete.

2 Efficient Computation of Well-formed Default Unification

The core result of our work is a refinement of the characterization of CDU as described in (Carpenter, 1993) along with a revision of the algorithm presented in (Alexandersson & Becker, 2007). By identifying basic scenarios, see figure 2, we have gained insights into the limitations of Carpenter as described below and identified the resulting deficits of the algorithm in Alexandersson and Becker. Crucial facets of our work include efficient lattice operations, for the calculation of *maximal lower bound (mlb)* and *minimal upper bound (mub)*. During CDU we suggest that the well-formedness condition on TFS formulated in (Copestake, 1992) is met. The two-stage procedural approach of the CDU mechanism is as the following:

1 **Assimilation** is a preprocessing step that first computes the target types for the cover and background; the *Type Preprocessing*. Next, the cover and the background will be translated to their appropriate target types by the operations *Specialization* and *Generalization* respectively.

2 **Overlay**’ performs type assignment by combining the information from the two TFSs returned by the assimilation and monitors the recursive invocation of Assimilation.²

¹In Grover et al. *priority union* is aimed to resolve parallelism dependent anaphora, in particular verb phrase ellipsis with a resolution mechanism based on the linguistic discourse model of (Prüst, Scha, & Berg, 1994).

²For a detailed description of Overlay’ see (Alexandersson & Becker, 2007).

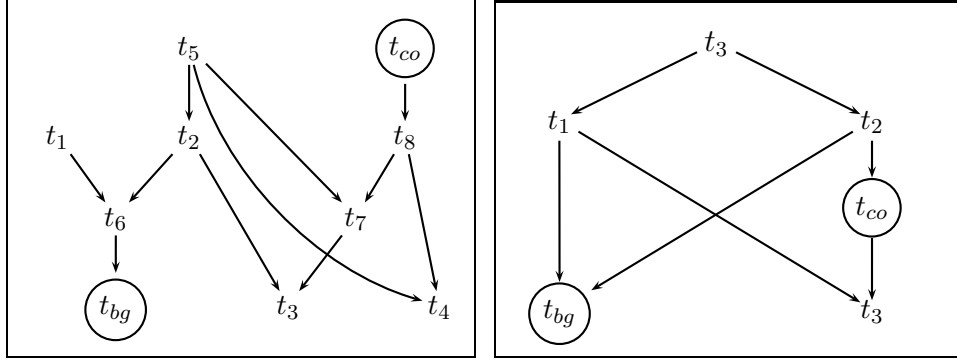


Figure 2: A CDU scenario that is not entirely covered by Carpenter’s definition (left). Determination of the required search space of all relevant type configurations (right).

A convenient concept is the definition of the *type configuration*, which enables to describe the Type Preprocessing effectively:

Definition 1 Type Configuration

Given t_{bg} , the type of the background and t_{co} , the type of the cover; a *type configuration* t_{conf} , $\langle t'_{bg}, t'_{co} \rangle$ is a pair of type identifiers where, t'_{bg} is the *abstract background type* of t_{bg} with $\sqcup_t(t'_{bg}, t_{co}) \neq \emptyset$ and $t'_{bg} \sqsubseteq t_{bg}$; t'_{co} is the *abstract cover type* of t_{co} with $t'_{co} \in \sqcup_t(t'_{bg}, t_{co})$

At this point we would like to express subsumption ordering between two type configurations $t'_{conf} = \langle t'_{bg}, t'_{co} \rangle$ and $t''_{conf} = \langle t''_{bg}, t''_{co} \rangle$ as follows. We say $t'_{conf} \sqsubseteq t''_{conf}$, if $t'_{bg} \sqsubseteq t''_{bg}$ and $t'_{co} \sqsubseteq t''_{co}$. Given some type t and $\theta(\bar{q}_{co}) = t_{co}$ we additionally propose a function, that returns the potential abstract background types located in the next level of the hierarchy $\mathcal{T}(t) = \{t' \mid t' \sqsubseteq t \text{ with } t' \text{ is maximal and } \sqcup_t(t', t_{co}) \neq \emptyset\}$.

The scenario in figure 2 (left) bears the type configuration $\langle t_2, t_3 \rangle$ which is covered by Carpenter’s definition, since t_2 corresponds to the relaxed structure of the background which according to Carpenter should contain maximal information. However, $\langle t_5, t_4 \rangle$ encloses the valid abstract background type t_5 that is more general than t_2 , which is caused by the non-deterministic inheritance behaviour among background’s supertypes; $\sqcup_t(t_5, t_{co})$ yields a non-empty set $\{t_7, t_4\}$. This example reveals the limits of Carpenter’s definition, that can be adjusted by the omission of the restriction on specificity imposed on the generalized background structure:

Definition 2 Credulous Default Unification Revised

$$F \overset{<}{\sqcup}_c G = \{F \sqcup G' \mid G' \sqsubseteq G \text{ such that } F \sqcup G' \text{ is defined and maximal}\}$$

The definition says that the result of CDU between F and G are represented by unifications between F and G' , that are defined, most specific

and G' subsumes G . For this purpose we motivate the task of Type Preprocessing to identify all type configurations that are relevant for subsequent procedures in CDU. The hierarchy in figure 2 (left) comprises the type configurations $\{\langle t_2, t_3 \rangle, \langle t_5, t_4 \rangle, \langle t_5, t_7 \rangle\}$, though $\langle t_2, t_3 \rangle$ makes $\langle t_5, t_7 \rangle$ dispensable, since $\langle t_5, t_7 \rangle$ results into a TFS that subsumes the outcome stemming from $\langle t_2, t_3 \rangle$. Type Preprocessing sorts out redundant type configurations by the subsumption check and prevents them to be input for the Specialization and the Generalization process respectively.

The algorithm of Alexandersson and Becker narrows down the search for abstract background types to the candidates that are situated exclusively on the path between the background type and the *least upper bound* of cover and background. The hierarchy in figure 2 (right), however, contains an abstract background type that is not located on any path towards elements of mub , i. e., it is justified to state that in addition to t_2 also the type t_1 is an abstract background type. In order to capture all potentially valid type configurations the search may only terminate if the currently visited type subsumes a mub element.

Algorithm 1 Assimilation

Let co and bg be two TFS such that $co = \langle Q_{co}, \bar{q}_{co}, \theta_{co}, \delta_{co} \rangle$ and $bg = \langle Q_{bg}, \bar{q}_{bg}, \theta_{bg}, \delta_{bg} \rangle$. Further we have $t_{bg} := \theta_{bg}(\bar{q}_{bg})$ and $t_{co} := \theta_{co}(\bar{q}_{co})$. The assimilation of co and bg , $\mathcal{A}(bg, co) = \Omega$, where Ω is a set of tuples of bg' and co' such that:

<p>(1) if $t_{bg} \sqsubseteq t_{co}$ then $bg' = bg, co' = co$ $\langle bg', co' \rangle \in \Omega$</p> <p>(2) if $t_{co} \sqsubseteq t_{bg}$ if $\sqcup(co, MGsat(t_{bg})) \neq \emptyset$ then $CO' = \sqcup(co, MGsat(t_{bg}))$ $bg' = bg$ $\{bg'\} \times CO' \subseteq \Omega$ else go to (3*) with $\mathcal{T}(t_{bg})$ $t_{mub} = t_{co}$</p> <p>(3) else for each $t_{mub} \in \sqcap_t(t_{bg}, t_{co})$ go to (3*) with $\mathcal{T}(t_{bg})$</p> <p>(3*) for each $t_{bg'} \in \mathcal{T}(t)$ go to (3.1)</p>	<p>(3.1) if $t'_{bg} \sqsubseteq t_{mub}$ then exit else for each $t_{mlb} \in \sqcup_t(t'_{bg}, t_{co})$ go to (3.2)</p> <p>if $\langle t'_{bg}, t_{mlb} \rangle \not\sqsubseteq t'_{conf} \forall t'_{conf};$ with $t'_{conf} = \langle \theta(\bar{q}_{bg'}), \theta(\bar{q}_{co'}) \rangle$ and $\langle bg', co' \rangle \in \Omega$ then go to (3.3) else go to (3*) with $\mathcal{T}(t'_{bg})$</p> <p>if $\sqcup(co, MGsat(t_{mlb})) \neq \emptyset$ then $CO' = \sqcup(co, MGsat(t_{mlb}))$ $bg' = \mathcal{G}(bg, t'_{bg})$ $\{bg'\} \times CO' \subseteq \Omega$ go to (3*) with $\mathcal{T}(t'_{bg})$ else go to (3*) with $\mathcal{T}(t'_{bg})$</p>
--	---

Figure 3: Along the CDU procedure Overlay' recursively assimilates the structures of the arguments one level deeper and combines the previously assimilated parts stemming from background and cover resulting in an intermediate TFS.

During CDU well-formedness of TFS is guaranteed if exclusively Specialization is well-formed, where its operational foundation refers to the ternary unification operation in (Copestake, 1992). In the algorithm specification 1 we present the Assimilation algorithm concerning multiple inheritance hierarchies featuring non-determinism.

3 Conclusion and Future Work

We have motivated and demonstrated a formalism of CDU dealing with multiple inheritance hierarchies featuring non-determinism. Thereby we have provided a reformulation of Carpenter's definition and a precisely formalized characterization of the algorithm extending the work in (Alexandersson & Becker, 2007). The non-deterministic behaviour increases considerably the degree of difficulty regarding lattice operations in order to identify all possible outcomes in CDU. To this end, we were able to boil down the efficient implementation of CDU to an efficient realization of *mlb* and *mub* computation based on the detailed discussion in (Schulz, 2007).

In this work we have gained insights considering coreferences, though we postpone the theoretical and practical elaboration as a goal for future research. A crucial contribution in (Alexandersson, Becker, & Pfeleger, 2004) is the usage of a scoring function that computes the best hypothesis among the multiple outcome of CDU. The notion of *informational distance* appears to be a reasonable device to restrain the production of multiple results in the current context as well.

References

- Alexandersson, J., & Becker, T. (2007). Efficient Computation of Overlay for Multiple Inheritance Hierarchies in Discourse Modeling. In H. Bunt & R. Muskens (Eds.), (Vol. 3, pp. 423–455). Dordrecht: Dordrecht:Kluwer.
- Alexandersson, J., Becker, T., & Pfeleger, N. (2004). Scoring for overlay based on informational distance. In *Proceedings of Konvens 2004* (pp. 1–4). Vienna, Austria.
- Bouma, G. (1990, 6–9 June). Defaults in Unification Grammar. In *Proceedings of the 28th annual meeting on Association for Computational Linguistics* (pp. 165–172). Morristown, NJ, USA: Association for Computational Linguistics.

- Carpenter, B. (1993). Skeptical and Credulous Default Unification with Applications to Templates and Inheritance. In T. Briscoe, V. de Paiva, & A. Copestake (Eds.), *Inheritance, Defaults, and the Lexicon* (pp. 13–37). Cambridge, CA: Cambridge University Press.
- Copestake, A. (1992). *The Representation of Lexical Semantic Information*. Doctoral dissertation, University of Sussex.
- Grover, C., Brew, C., Manandhar, S., & Moens, M. (1994). Priority Union and Generalization in Discourse Grammars. In *32nd. Annual Meeting of the Association for Computational Linguistics* (pp. 17 – 24). Las Cruces, NM: Association for Computational Linguistics.
- Kaplan, R. M. (1987). Three Seductions of Computational Psycholinguistics. In P. Whitelock, H. Somers, P. Bennett, R. Johnson, & M. M. Wood (Eds.), *Linguistic Theory and Computer Applications* (pp. 149–188). London: Academic Press.
- Prüst, H., Scha, R., & Berg, M. van den. (1994). Discourse Grammar and Verb Phrase Anaphora. In *Linguistics and Philosophy 17* (pp. 261–327). Amsterdam, Netherlands: Springer.
- Schulz, C. H. (2007). *Well-formed Default Unification in Multiple Inheritance Hierarchies*. Diploma thesis, Saarland University, Saarbrücken, Germany.
- Shieber, S. M. (1986). A Simple Reconstruction of GPSG. In *Proc. of the 11th COLING* (pp. 211–215). Morristown, NJ, USA: Association for Computational Linguistics.
- Wahlster, W. (2006). Dialogue Systems Go Multimodal: The Smartkom Experience. In *Smartkom - Foundations of Multimodal Dialogue Systems* (pp. 3–27). Heidelberg, Germany: Springer.