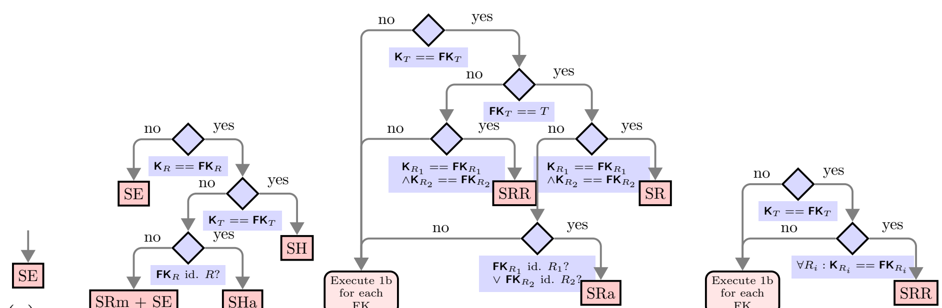
Appendix of "ADaMaP: Automatic Alignment of Relational Data Sources using Mapping Patterns"

Diego Calvanese Avigdor Gal Naor Haba
Davide Lanti Alessandro Mosca
Marco Montali
Roee Shraga

March 5, 2021

1 The ADaMaP algorithm



(a) 0 (b) 1 (c) 2 (d) ≥ 3

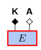
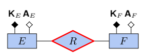
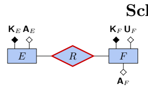
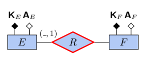
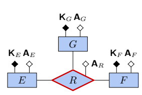
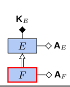
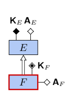
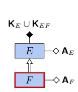
Figure 1: ADaMaP inference for a table T by the number of foreign keys it contains.

Algorithm 1 ADaMaP Inference

Input: a schema Σ and a set of patterns $\mathcal{P}$
Output: an alignment M
$M \leftarrow \emptyset$
for $T \in \Sigma$ **do**
  apply Figure 1 with T to obtain a set of correspondences m
  $M \leftarrow M \cup m$
end for
Return: M

2 Table of Schema-driven Patterns

Table 1: Schema-driven Patterns. For patterns yielding views, we show the views together with the DB schema, separating them from the original tables using a thick horizontal bar.

E-R DIAGRAM	DB SCHEMA	MAPPING PATTERN	ONTOLOGY
Schema/Data Entity (SE/DE)			
	$T_E(\underline{\mathbf{K}},\mathbf{A})$	s: T_E t: $C_E(\mathsf{t}_E(\mathbf{K}))$, $\{d_A(\mathsf{t}_E(\mathbf{K}),A)\}_{A\in\mathbf{K}\cup\mathbf{A}}$	$\{\exists d_A \sqsubseteq C_E\}_{A\in\mathbf{K}\cup\mathbf{A}}$
Schema/Data Relationship (SR/DR)			
	$T_E(\underline{\mathbf{K}_E},\mathbf{A}_E)$ $T_F(\underline{\mathbf{K}_F},\mathbf{A}_F)$ $T_R(\underline{\mathbf{K}_{RE},\mathbf{K}_{RF}})$	s: T_R t: $p_R(\mathsf{t}_E(\mathbf{K}_{RE}),\mathsf{t}_F(\mathbf{K}_{RF}))$	$\exists p_R \sqsubseteq C_E$ $\exists p_R^- \sqsubseteq C_F$
In case of $(\_,1)$ cardinality on role R_E (resp., R_F), the primary key for T_R is restricted to the attributes $\mathbf{K}_{RE}$ (resp., $\mathbf{K}_{RF}$).			
Schema/Data Relationship with Identifier Alignment (SRa/DRa)			
	$T_E(\underline{\mathbf{K}_E},\mathbf{A}_E)$ $T_F(\underline{\mathbf{K}_F},\mathbf{U}_F,\mathbf{A}_F)$ $T_R(\underline{\mathbf{K}_{RE},\mathbf{U}_{RF}})$ $\mathsf{unique}_{R_F}(\mathbf{U}_F)$	s: $T_R \ltimes_{\mathbf{U}_{RF}=\mathbf{U}_F} T_F$ t: $p_R(\mathsf{t}_E(\mathbf{K}_{RE}),\mathsf{t}_F(\mathbf{K}_F))$	$\exists p_R \sqsubseteq C_E$ $\exists p_R^- \sqsubseteq C_F$
In case of $(\_,1)$ cardinality on role R_E (resp., R_F), the primary key for T_R is restricted to the attributes $\mathbf{K}_{RE}$ (resp., $\mathbf{U}_{RF}$).			
Schema/Data Relationship with Merging (SRm/DRm)			
	$T_F(\underline{\mathbf{K}_F},\mathbf{A}_F)$ $T_E(\underline{\mathbf{K}_E},\mathbf{K}_{EF},\mathbf{A}_E)$	s: T_E t: $p_{EF}(\mathsf{t}_E(\mathbf{K}_E),\mathsf{t}_F(\mathbf{K}_{EF}))$	$\exists p_{EF} \sqsubseteq C_E$ $\exists p_{EF}^- \sqsubseteq C_F$
Schema/Data Reified Relationship (SRR/DRR)			
	$\mathbf{K}_R := \mathbf{K}_{RE}\cup\mathbf{K}_{RF}\cup\mathbf{K}_{RG}$ $T_G(\underline{\mathbf{K}_G},\mathbf{A}_G)$ $T_R(\underline{\mathbf{K}_{RE},\mathbf{K}_{RF},\mathbf{K}_{RG}},\mathbf{A}_R)$ $T_E(\underline{\mathbf{K}_E},\mathbf{A}_E)$ $T_F(\underline{\mathbf{K}_F},\mathbf{A}_F)$	s: T_R t: $C_R(\mathsf{t}_R(\mathbf{K}_R))$, $\{d_A(\mathsf{t}_R(\mathbf{K}_R),A)\}_{A\in\mathbf{K}_R\cup\mathbf{A}_R}$, $p_{RE}(\mathsf{t}_R(\mathbf{K}_R),\mathsf{t}_E(\mathbf{K}_{RE}))$, $p_{RF}(\mathsf{t}_R(\mathbf{K}_R),\mathsf{t}_F(\mathbf{K}_{RF}))$, $p_{RG}(\mathsf{t}_R(\mathbf{K}_R),\mathsf{t}_G(\mathbf{K}_{RG}))$	$\exists p_{RE} \sqsubseteq C_R$ $\exists p_{RE}^- \sqsubseteq C_E$ $\exists p_{RF} \sqsubseteq C_R$ $\exists p_{RF}^- \sqsubseteq C_F$ $\exists p_{RG} \sqsubseteq C_R$ $\exists p_{RG}^- \sqsubseteq C_G$ $\{\exists d_A^- \sqsubseteq C_R\}_{A\in\mathbf{K}_R\cup\mathbf{A}_R}$
SRR applies whenever there are three or more participating roles, or when the relationship has attributes. • If R is identified by a strict subset X of the participating roles, then the primary key of table T_R (and the corresponding set of attributes $\mathbf{K}_R$) is restricted to the foreign keys corresponding to the roles in X. For instance, if R is identified by roles R_E and R_G, then the primary key of T_R (and the corresponding set of attributes $\mathbf{K}_R$) is $\mathbf{K}_{RE}\cup\mathbf{K}_{RG}$. • If one of the foreign keys is to a non-primary key set of attributes, the object property relative to that foreign key is dealt with as in SRa.			
Schema/Data Hierarchy (SH/DH)			
	$T_E(\underline{\mathbf{K}_E},\mathbf{A}_E)$ $T_F(\underline{\mathbf{K}_{FE}},\mathbf{A}_F)$	s: T_F t: $C_F(\mathsf{t}_E(\mathbf{K}_{FE}))$, $\{d_A(\mathsf{t}_E(\mathbf{K}_{FE}),A)\}_{A\in\mathbf{A}_F}$	$C_F \sqsubseteq C_E$ $\{\exists d_A^- \sqsubseteq C_F\}_{A\in\mathbf{A}_F}$
Schema/Data Hierarchy with Identifier Alignment (SHa/DHa)			
	$T_E(\underline{\mathbf{K}_E},\mathbf{A}_E)$ $\mathsf{unique}_{T_F}(\mathbf{U}_F)$ $T_F(\underline{\mathbf{K}_F},\mathbf{U}_F,\mathbf{A}_F)$ ——— $T_E(\underline{\mathbf{K}_E},\mathbf{A}_E)$ $\mathsf{unique}_{V_F}(\mathbf{K}_F)$ $V_F(\mathbf{K}_F,\underline{\mathbf{U}_F},\mathbf{A}_F) = T_F$	s: T_F t: $C_F(\mathsf{t}_E(\mathbf{U}_F))$, $\{d_A(\mathsf{t}_E(\mathbf{U}_F),A)\}_{A\in\mathbf{K}_F\cup\mathbf{A}_F}$	$C_F \sqsubseteq C_E$ $\{\exists d_A^- \sqsubseteq C_F\}_{A\in\mathbf{K}_F\cup\mathbf{A}_F}$
In this pattern, the "alignment" is meant to align the primary identifier used in the child entity to the primary identifier used in the parent entity. We here depict the most common scenario, where the foreign key points to the primary key of the parent entity. The other two possiblities for the application of the pattern are: • the foreign key in the child entity coincides with the primary key of that entity, and references a non-primary key of the parent entity; • the foreign key in the child entity does not coincide with the primary key of that entity, and references a non-primary key of the parent entity.			
Schema/Data Hierarchy with Advanced Alignment (SHaa/DHaa)			
 $\mathbf{K}_E$ in F is a constant c, $\mathbf{K}_{EF}$ alone identifies F.	$T_E(\underline{\mathbf{K}_E,\mathbf{K}_{EF}},\mathbf{A}_E)$ $T_F(\underline{\mathbf{K}_F},\mathbf{A}_F)$	s: $\pi_{\mathsf{c},\mathbf{K}_F}(T_F)$ t: $C_F(\mathsf{t}_E(\mathsf{c},\mathbf{K}_F))$, $\{d_A(\mathsf{t}_E(\mathsf{c},\mathbf{K}_F),A)\}_{A\in\mathbf{A}_F}$	$C_F \sqsubseteq C_E$ $\{\exists d_A^- \sqsubseteq C_F\}_{A\in\mathbf{A}_F}$
$\mathbf{K}_E$ is not inherited by the child table, because it is a constant value c. The dependency between T_F and T_E is an inclusion dependency, rather than a foreign key dependency.			