# Retrospective on Clio: Schema Mapping and Data Exchange in Practice

Renée J. Miller

Bell University Labs Chair of Information Systems
Department of Computer Science
University of Toronto
miller@cs.toronto.edu

**Abstract.** Clio is a joint research project between the University of Toronto and IBM Almaden Research Center started in 1999 to address both foundational and systems issues related to the management of heterogeneous data. In this talk, I will take a look back over the last eight years of this project to review its achievements, the lessons learned, and the challenges that remain.

**Key words:** Schema mapping, mapping generation, mapping compilation and execution, data exchange, data integration

The Clio project was founded to tackle the challenging issues raised by the proliferation of independently developed data sources that are heterogeneous in their design and content. Heterogeneous data sets contain data that have been represented using different data models, different structuring primitives, or different modeling assumptions. Such data sets often have been developed and modeled with different requirements in mind. As a consequence, different schemas may have been used to represent the same or related data. To manage heterogeneous data, we must be able to manage these schemas and mappings between the schemas. Clio is a management system for heterogeneous data that couples traditional data management solutions with additional tools for creating, using and maintaining mappings between schemas. Our first results on schema mapping generation [MHH00] were first demonstrated at SIGMOD 2001 [HMH01]. The main contributions of the Clio project include the following. .

*Schema mapping generation.* In many integration applications, data that conforms to a (*source*) schema (also called a local schema), may be queried or viewed through another (*target*) schema (also called a global schema). The relationship between the source and the target schema is modeled through a set of artifacts called *schema mappings.* Prior to Clio, most work on automating schema mapping generation focused on finding simple attribute-attribute correspondences or matches (using text similarity or natural language techniques). Clio introduced a novel interactive mapping creation paradigm [MHH00] for creating mappings that represent structural transformations of data. These mappings are created using the semantics encoded in the schemas, and represent the semantic relationship between schemas. We have investigated how to automate the mapping creation process, and how to effectively solicit user input when the semantics of the schemas are ambiguous or incomplete [MHH00,PVM$^+$02]. This work was demonstrated at ICDE 2002 [HPV$^+$02]. In addition, we have developed a data-driven visualization tool to help users understand and refine mappings as they are generated [YMHF01]. We use carefully chosen data examples to explain mappings and alternative mappings to help a user arrive at a correct and complete mapping. Most recently, we have considered the generation of mappings that may be correlated with other mappings [FHH$^+$06], work that was demonstrated at ICDE 2007 [HHP$^+$07]

*Schema mapping specification.* Mappings are specified using an expressive declarative language with a formal semantics. The mapping language is suitable for relational schemas and for nested structures including XML schema, DTDs, and concept hierarchies. The expressiveness of this mapping language enables a wide range of transformations and makes the language suitable for use in a variety of mapping applications [PVM$^+$02,FHH$^+$06].

In creating a mapping from a source schema to a target schema, we do not assume that the schemas represent the same data. Certainly, there may be source data that are not represented in the target. Additionally, there may be target data that are not represented in the source. An accurate mapping must be able to represent missing (unmapped) source and target data [PVM$^+$02].

*Using Mappings.* The mappings produced by Clio can be used for both *data integration* (where the target data is virtual) [YP04] and for *data exchange* (where the target data is materialized) [PVM$^+$02]. For data exchange, we faced the challenge of being able to generate new values for unspecified (unmapped) target data that are essential for ensuring the consistency of the target database. Clio presented the first algorithm for data exchange in 2002 using a solution that is guaranteed to generate a valid target database [PVM$^+$02].

We have gone on to investigate some of the foundations of data exchange [FKMP03,FKMP05] with a formal study of how data exchange differs from data integration. Importantly, this formal study uses a mapping language (tuple-generating dependencies) that are inspired by the Clio system. We have considered how to characterize the best data exchange solution [FKP05], how to do data exchange in networks of peer schemas [FKMT06], how to compose mappings [FKPT05], and a number of other issues surrounding the use of mappings [Fag06].

*Mapping compilation and execution.* Mappings can be compiled into executable queries or programs which perform data exchange [PVM$^+$02]. Depending on the application environment, Clio can generate SQL queries, Xquery, XSLT, among other formats, for execution. Alternatively, Clio also provides its own optimized execution engine for efficient evaluation of data exchange programs [JHPH07].

Our declarative mapping formalism is sufficiently expressive to capture the semantics of a wide variety of integration and exchange applications, and as a result, the ability to compile mappings into different runtime environments for efficient execution is critical.

*Managing and maintaining mappings.* Our declarative mappings can be adapted and reused more easily than low-level procedural mapping scripts. We have presented solutions for adapting mapping as schemas evolve [VMP03,VMP04,YP05], demonstrated at ICDE 2004 [VMPM04].

*Industrial impact.* Clio technology has been transferred into IBM's product lines, and forms a core component of IBM's Rational Data Architect [HHH$^+$05]. The use of declarative mappings (over hand-coded procedural scripts) is now considered "best-practice" in commercial products and Clio has lead the way in changing the culture. In addition, Clio has been a leader in demonstrating that both schemas and mappings are dynamic artifacts that must be managed effectively.

## References

[Fag06] Ronald Fagin. Inverting Schema Mappings. In *Proc. of the ACM Symp. on Principles of Database Systems (PODS)*, pages 50–59, 2006.

[FHH$^+$06] Ariel Fuxman, Mauricio A. Hernández, Howard Ho, Renée J. Miller, Paolo Papotti, and Lucian Popa. Nested Mappings: Schema Mapping Reloaded. In *Proc. of the Int'l Conf. on Very Large Data Bases (VLDB)*, pages 67–78, 2006.

[FKMP03] Ronald Fagin, Phokion G. Kolaitis, Renée J. Miller, and Lucian Popa. Data Exchange: Semantics and Query Answering. In *Proc. of the Int'l Conf. on Database Theory (ICDT)*, pages 207–224, 2003.

[FKMP05] Ronald Fagin, Phokion G. Kolaitis, Renée J. Miller, and Lucian Popa. Data Exchange: Semantics and Query Answering. *Theoretical Computer Science*, 336(1):89–124, May 2005.

[FKMT06] Ariel Fuxman, Phokion G. Kolaitis, Renée J. Miller, and Wang-Chiew Tan. Peer Data Exchange. *ACM Trans. on Database Systems (TODS)*, 31(4):1454–1498, 2006.

[FKP05] Ronald Fagin, Phokion G. Kolaitis, and Lucian Popa. Data exchange: getting to the core. *ACM Trans. on Database Systems (TODS)*, 30(1):174–210, 2005.

[FKPT05] Ronald Fagin, Phokion G. Kolaitis, Lucian Popa, and Wang-Chiew Tan. Composing schema mappings: Second-order dependencies to the rescue. *ACM Trans. on Database Systems (TODS)*, 30(4):994–1055, 2005.

[HHH$^+$05] Laura M. Haas, Mauricio A. Hernández, Howard Ho, Lucian Popa, and Mary Roth. Clio grows up: from research prototype to industrial tool. In *ACM SIGMOD Int'l Conf. on the Management of Data*, pages 805–810, 2005.

[HHP$^+$07] Mauricio A. Hernández, Howard Ho, Lucian Popa, T. Fukuda, Ariel Fuxman, Renée J. Miller, and Paolo Papotti. Creating Nested Mappings with Clio. In *IEEE Proc. of the Int'l Conf. on Data Engineering (ICDE)*, 2007. System Demonstration.

[HMH01] Mauricio A. Hernández, Renée J. Miller, and Laura Haas. Clio: A Semi-Automatic Tool for Schema Mapping. *ACM SIGMOD Int'l Conf. on the Management of Data*, 30(2):607, 2001. System Demonstration.

[HPV$^+$02] Mauricio A. Hernández, Lucian Popa, Yannis Velegrakis, Renée J. Miller, F. Naumann, and C.-T. Ho. Mapping XML and Relational Schemas with Clio. In *IEEE Proc. of the Int'l Conf. on Data Engineering (ICDE)*, pages 498–499, 2002. System Demonstration.

[JHPH07] Haifeng Jiang, Howard Ho, Lucian Popa, and Wook-Shin Han. Mapping-Driven XML Tranformation. In *International World Wide Web Conference*, 2007.

[MHH00] Renée J. Miller, L. M. Haas, and Mauricio A. Hernández. Schema Mapping as Query Discovery. In *Proc. of the Int'l Conf. on Very Large Data Bases (VLDB)*, pages 77–88, 2000.

[MHH$^+$01] Renée J. Miller, Mauricio A. Hernández, L. M. Haas, Ling-Ling Yan, Howard Ho, Ronald Fagin, and Lucian Popa. The Clio Project: Managing Heterogeneity. *SIGMOD Record*, 30(1):78–83, March 2001.

[PVM$^+$02] Lucian Popa, Yannis Velegrakis, Renée J. Miller, Mauricio A. Hernández, and Ronald Fagin. Translating Web Data. In *Proc. of the Int'l Conf. on Very Large Data Bases (VLDB)*, pages 598–609, 2002.

[VMP03] Yannis Velegrakis, Renée J. Miller, and Lucian Popa. Mapping Adaptation under Evolving Schemas. In *Proc. of the Int'l Conf. on Very Large Data Bases (VLDB)*, pages 584–595, 2003.

[VMP04] Yannis Velegrakis, Renée J. Miller, and Lucian Popa. On Preserving Mapping Consistency under Schema Changes. *The Int'l Journal on Very Large Data Bases*, 13(3):274–293, 2004.

[VMPM04] Yannis Velegrakis, Renée J. Miller, Lucian Popa, and John Mylopoulos. ToMAS: A System for Adaptin Mappings as Schemas Evolve. In *IEEE Proc. of the Int'l Conf. on Data Engineering (ICDE)*, page 882, 2004. System Demonstration.

[YMHF01] Ling-Ling Yan, Renée J. Miller, Laura Haas, and Ronald Fagin. Data-Driven Understanding and Refinement of Schema Mappings. *ACM SIGMOD Int'l Conf. on the Management of Data*, 30(2):485–496, May 2001.

[YP04] Cong Yu and Lucian Popa. Constraint-Based XML Query Rewriting For Data Integration. In *ACM SIGMOD Int'l Conf. on the Management of Data*, pages 371–382, 2004.

[YP05] Cong Yu and Lucian Popa. Semantic Adaptation of Schema Mappings when Schemas Evolve. In *Proc. of the Int'l Conf. on Very Large Data Bases (VLDB)*, pages 1006–1017, 2005.