Microsoft Research

Model Management 2.0: Manipulating Richer Mappings

Phil Bernstein & Sergey Melnik
Microsoft Research

June 12, 2007

© 2007 Microsoft Corporation

Data Programmability

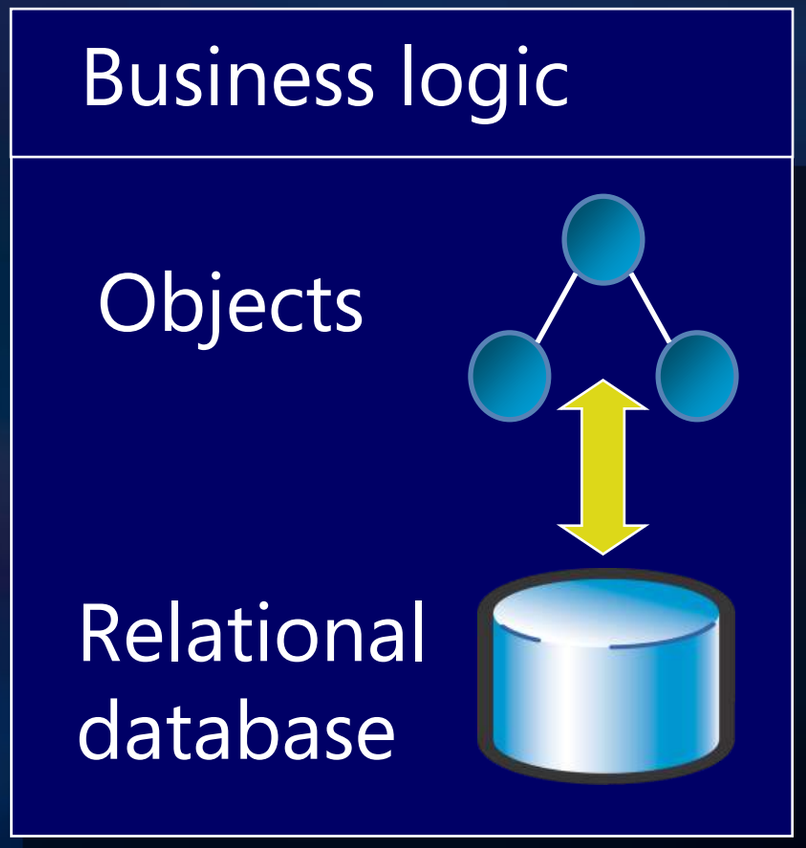
- Make it easier to write programs that access databases
- Traditionally, for large IT departments
- Much progress, but it's still ~40% of the work
- Core problem is developing and using complex mappings between schemas

Mapping Problems are Pervasive
And it's a Growth Industry

- Data translation
- XML message mapping
- Data warehouse loading
- Query mediators
- Forms managers
- Report writers
- Query designers
- Object-relational wrappers
- Portal generation from DB
- OLAP databases
- Application integration
- Composing web services

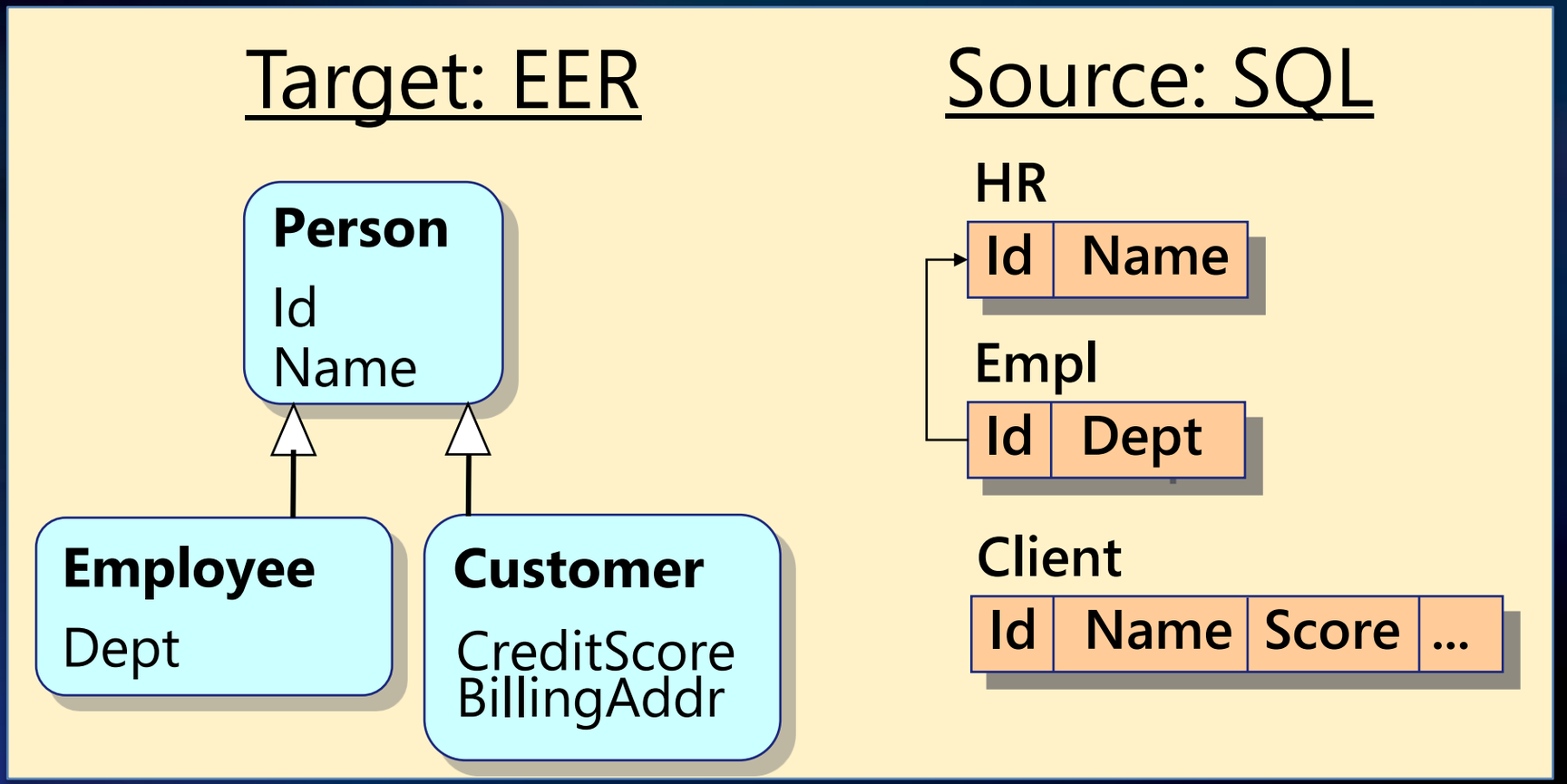
Object-Relational Wrappers

- Most packaged business apps need to access an OO view of relational data
- Requires an OR mapping



An Example Mapping

- Person = HR $\cup$ $\pi_{ID,Name}$(Client)
- Employee = HR $\bowtie$ Empl
- Customers = Client



Constructing Persons

[Melnik, Adya, Bernstein, SIGMOD 07]

```
SELECT VALUE
    CASE
      WHEN (T5._from2 AND NOT(T5._from1))  THEN Person(T5.Person_Id, T5.Person_Name)
      WHEN (T5._from1 AND T5._from2)
        THEN Employee(T5.Person_Id, T5.Person_Name,  T5.Employee_Dept)
        ELSE Customer(T5.Person_Id, T5.Person_Name,  T5.Customer_CreditScore,
             T5.Customer_BillingAddr)
    END
FROM (  (SELECT T1.Person_Id, T1.Person_Name,  T2.Employee_Dept,
                CAST(NULL AS SqlServer.int) AS Customer_CreditScore,
                CAST(NULL AS SqlServer.nvarchar) AS Customer_BillingAddr, False AS _from0,
                (T2._from1 AND T2._from1 IS NOT NULL) AS _from1,  T1._from2
           FROM  ( SELECT  T.Id AS Person_Id, T.Name AS Person_Name, True AS _from2
                   FROM HR AS T) AS T1
                 LEFT OUTER JOIN (
                   SELECT  T.Id AS Person_Id,  T.Dept AS Employee_Dept, True AS _from1
                   FROM dbo.Empl AS T) AS T2
                 ON T1.Person_Id = T2.Person_Id )
           UNION ALL (
                SELECT  T.Id AS Person_Id,  T.Name AS Person_Name,
                        CAST(NULL AS SqlServer.nvarchar) AS Employee_Dept,
                        T.Score AS Customer_CreditScore,  T.Addr AS Customer_BillingAddr,
                        True AS _from0,  False AS _from1,  False AS _from2
                FROM Client AS T)
      ) AS T5
```

Why is mapping hard?

[Haas, ICDT 07]

- Heterogeneity
- Impedance mismatch
- Insufficient abstraction
- Potpourri of tools

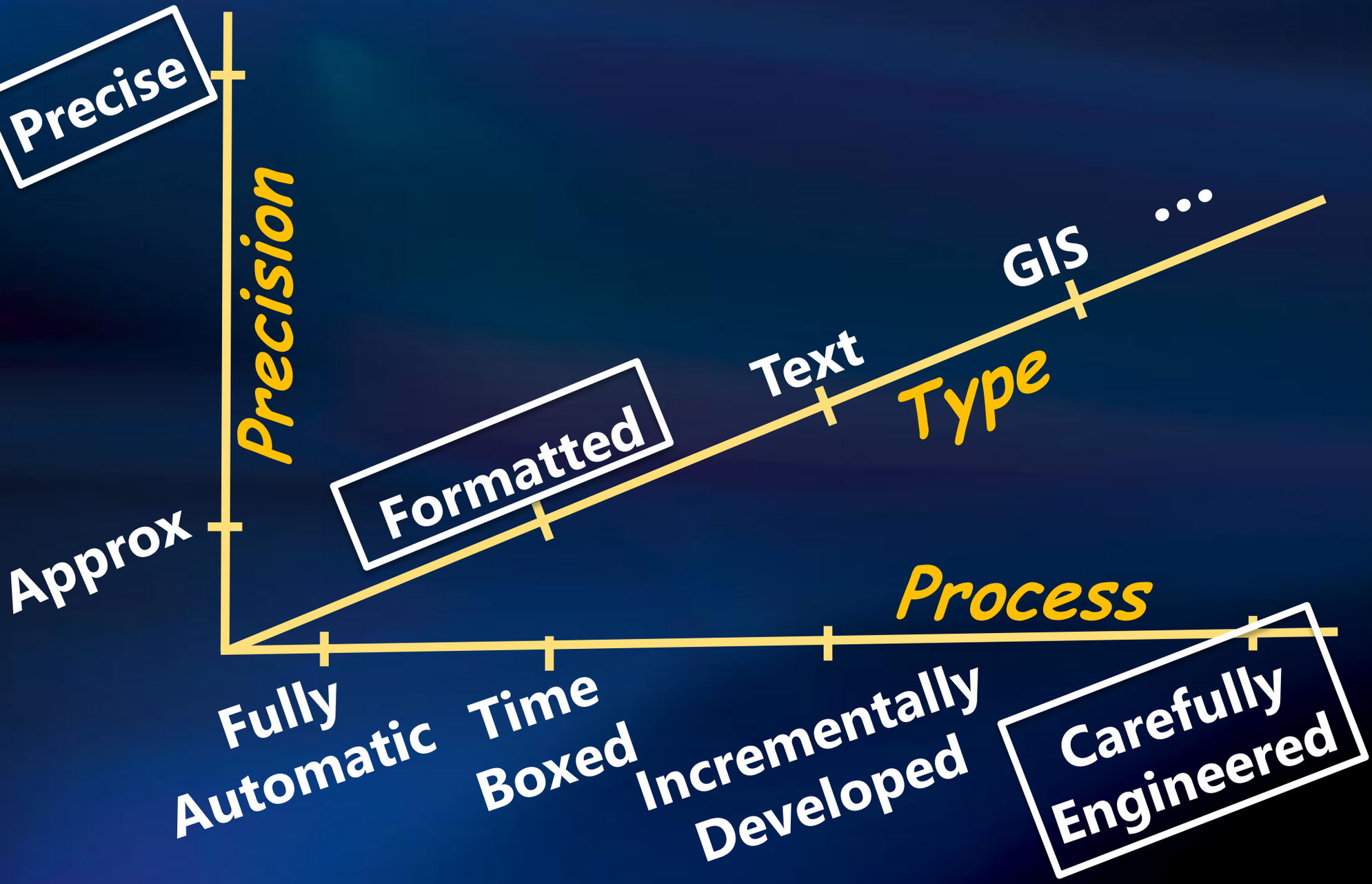


And It's Getting Harder

- More data models
  - Java, ODMG, XSD, .NET
  - RDF, OWL, EDM, SML
- More programming languages
- More types of tools
- More schema sources
  - Web site wrappers
  - Google Base
  - Generic info extractors [Gubanov, Bernstein WebDB 06]

Mapping Space

Info Integration Workshop
http://db.cis.upenn.edu/iiworkshop/



Model Management 1.0

[Bernstein, Halevy, Pottinger SIGMOD Record 00]

Manipulate
models & mappings
as bulk objects



Meta-model independent
- relational, ER, OO, XML, RDF, OWL, SML, …

Operations
- Match
- Compose
- Merge
- Diff/Extract
- ModelGen
- Inverse

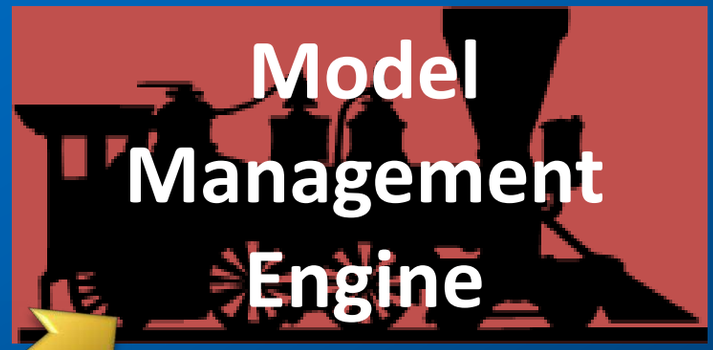
Tools

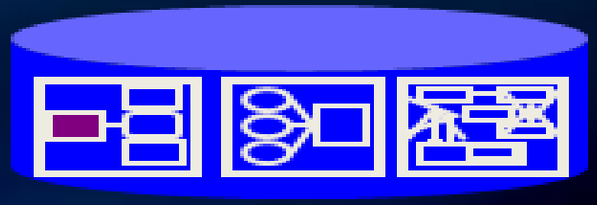


Wrapper Generator

Query Mediator

ETL





Metadata Repository

Model Management 1.0 Benefits

- More research focus on primary operations
  - More powerful operations
  - Hence better tools
- More leverage from tool investments
- More uniform behavior across tools

Good News / Bad News

- Good News
  - Lots of progress on operations
  - Some practical applications
  - A lot has been learned
- Bad News
  - Still waiting for the first reasonably-complete practical implementation
- Good news
  - A lot of research left to do

Outline

- What has changed: Use richer mappings
- What has changed: Include the runtime
- Model Management 2.0 in detail

What Has Changed?
Use Richer Mappings

2000
Structural mappings

- Mappings are structural
  - Relate schemas, *not* data
- Operations oblivious to mapping semantics
- Semantics is a plug-in

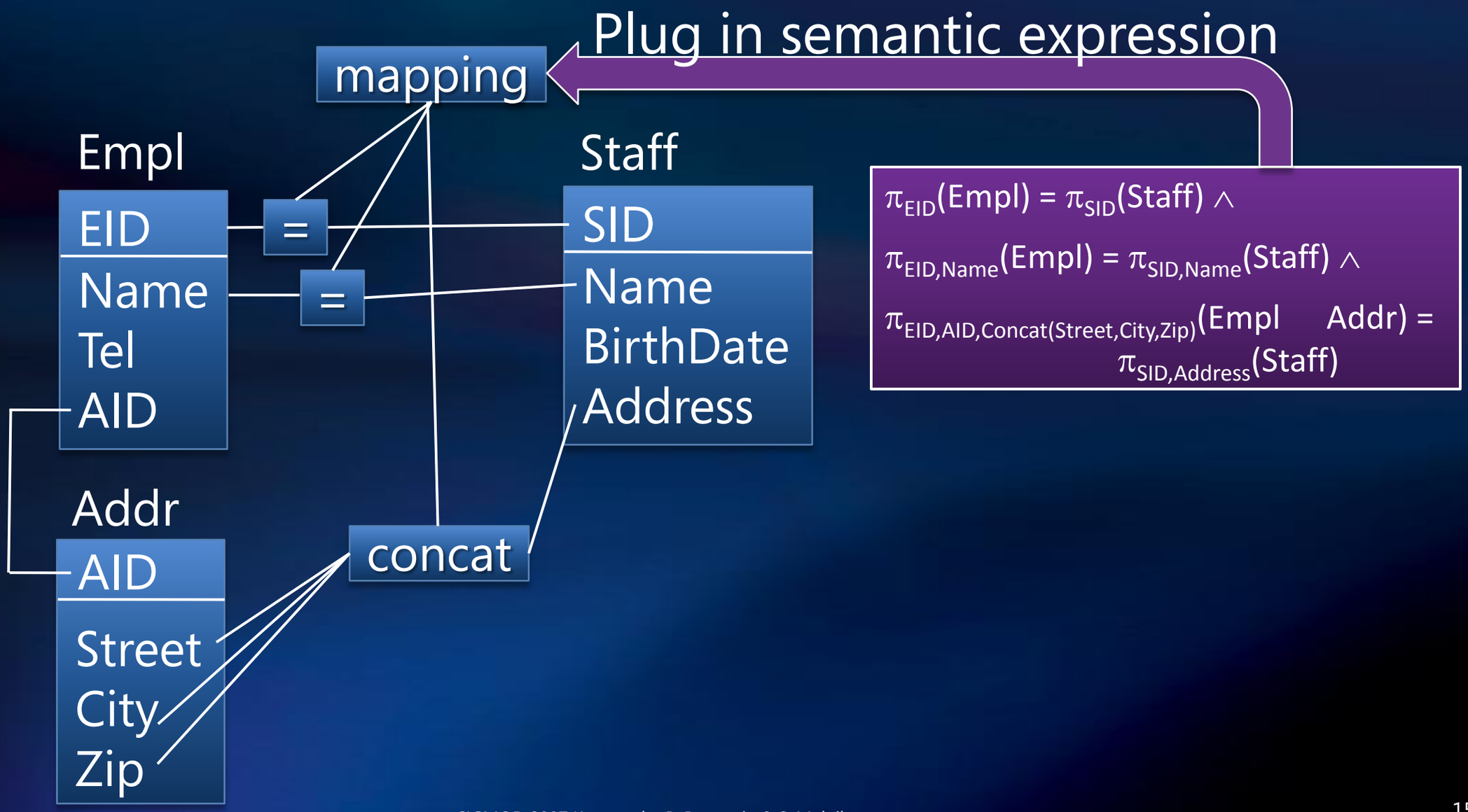
2007
Semantic mappings

- Mappings are semantic
  - Relate schemas *and* data
- Operations sensitive to mapping semantics
- Semantics is built-in

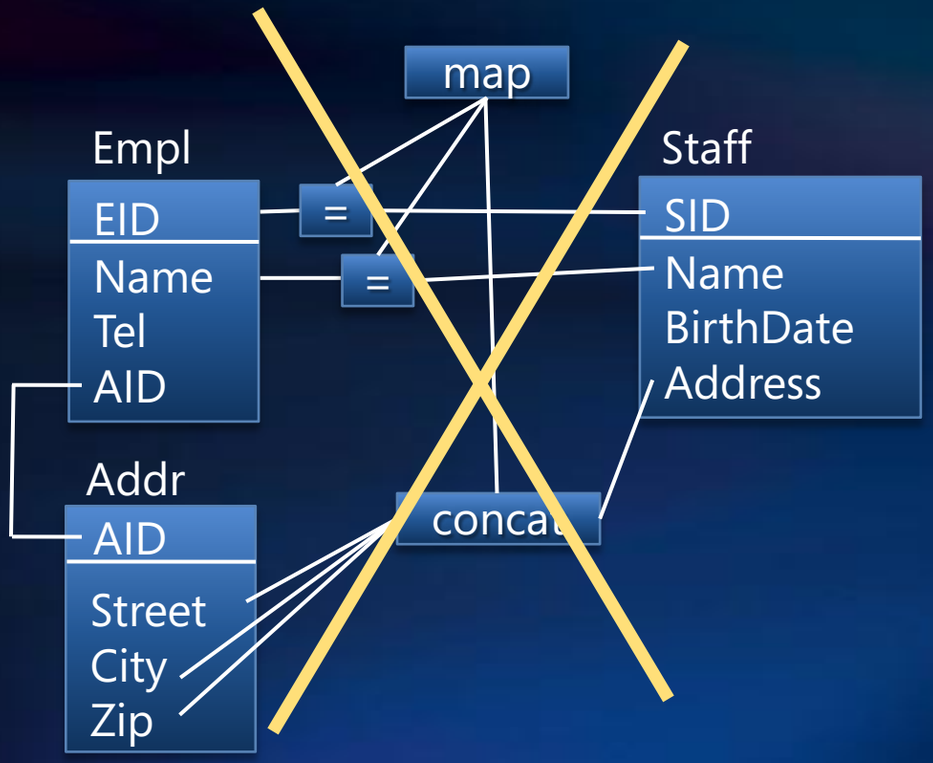
Example

[Bernstein. CIDR 03]

In 2000, mapping is a structure



Example
In 2007, just use the expression



$\pi_{EID}(Empl) = \pi_{SID}(Staff) \wedge$

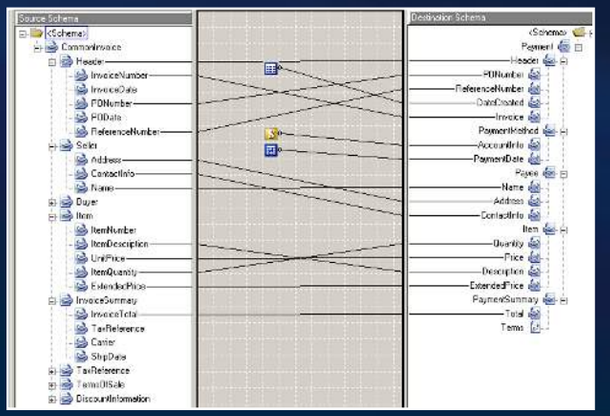
$\pi_{EID,Name}(Empl) = \pi_{SID,Name}(Staff) \wedge$

$\pi_{EID,AID,Concat(Street,City,Zip)}(Empl \quad Addr) =$
$\pi_{SID,Address}(Staff)$

Mappings

[Casanova, Vidal. PODS 83] [Batini, Lenzerini, Navathe. CSUR 86] [Motro. TSE 89] [Dayal, Hwang. TSE 84] [Katz, Wong. TSE 83] [Navathe, Gadgil. SIGMOD 82] [Smith, Bernstein, Blaustein, Goodman, Landers, Rosenthal, Wong. VLDB 80] [Smith, Bernstein, Fox, Landers, Reiner, Rothnie, Wong, Wood. CCA 83] [Smith, Bernstein, Landers. VLDB 81] [Smith, Bernstein, Lum. CACM 83] [Smith, Bernstein, Lum. VLDB 82] [Smith, Bernstein, Navathe. SIGMOD 81] [Smith, Bernstein, Wong. VLDB 79] [Smith, Fox, Landers. ICDE 83] [Smith, Fox, Landers. IJCOOPIS 93] [Biskup, Convent. SIGMOD 86] [Miller, Haas, Hernandez. VLDB 00]

- Element correspondences
  - First step in aligning schemas
  - For lineage & impact analysis
  - Weak or no formal semantics



- Mapping constraints relate instances of schemas
  - E.g., equality of relational expressions

```
SELECT Id, Name, Dept  =  SELECT Id, Name, Dept
FROM   Employee           FROM HR JOIN Empl ON Id
```

- Transformation is an executable mapping constraint
  - Constructs target instances from source instances
  - E.g., SQL query, XSLT, C# program

Mapping Expressiveness

- What we want: first-order logic with
  - aggregation
  - set and bag semantics
  - regular expressions
  - nested collections and lists
  - rich type constructors (e.g., to construct XML fragments),
  - user-defined functions
  - deduplication and other heuristic functions
- What can we handle? … Much less.

Parallel Evolution

Clio Project

- IBM, Univ. of Toronto, U.C. Santa Cruz
- Miller, Haas, Hernandez, Fagin, Ho, Popa, Tan, …

Model Management

- Microsoft, Univ. of Washington, Univ. of Leipzig
- Bernstein, Halevy, Pottinger, Rahm, Madhavan, Melnik, …

Build a design tool for semantic mappings

Build model management operations with plug-in semantics

Study model management operations with semantics

What Has Changed?
Include the Runtime

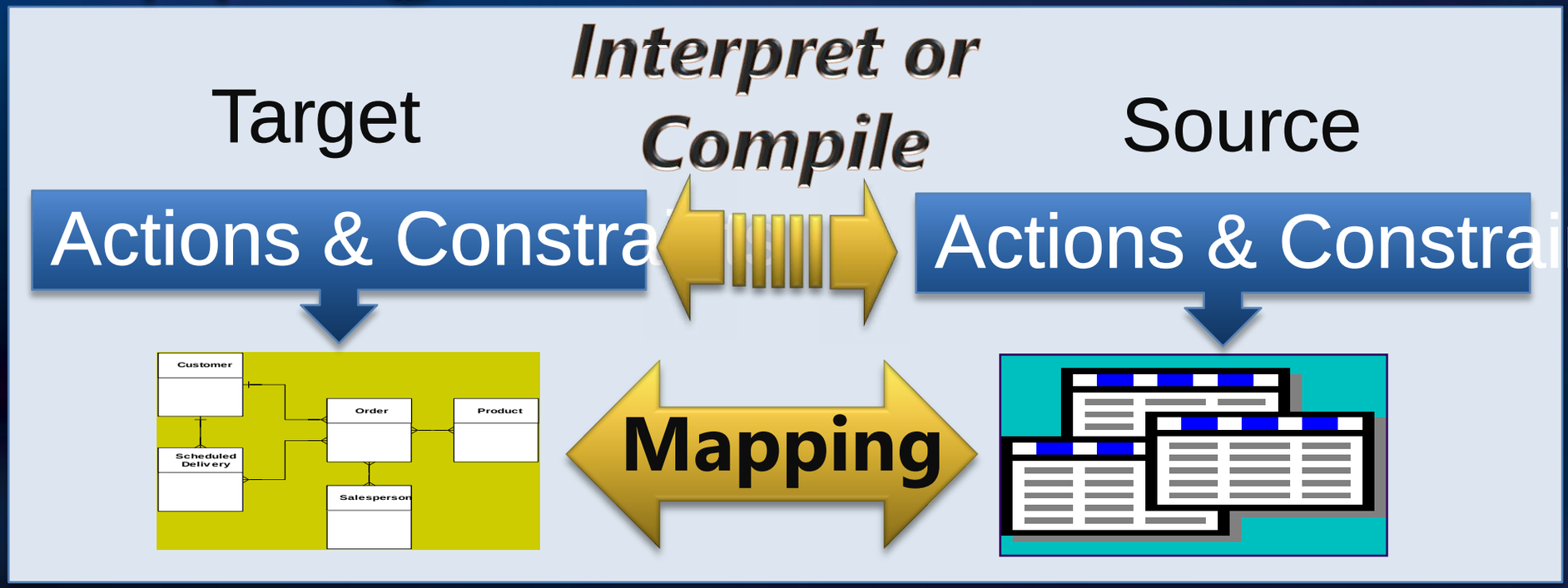
2000
Design-time

- Model management is independent of run-time
- No special run-time functionality

2007
Run-time

- Model management is tied to a run-time
- Run-time functions are sensitive to mapping expressiveness and model mgt capabilities

Mapping Runtime



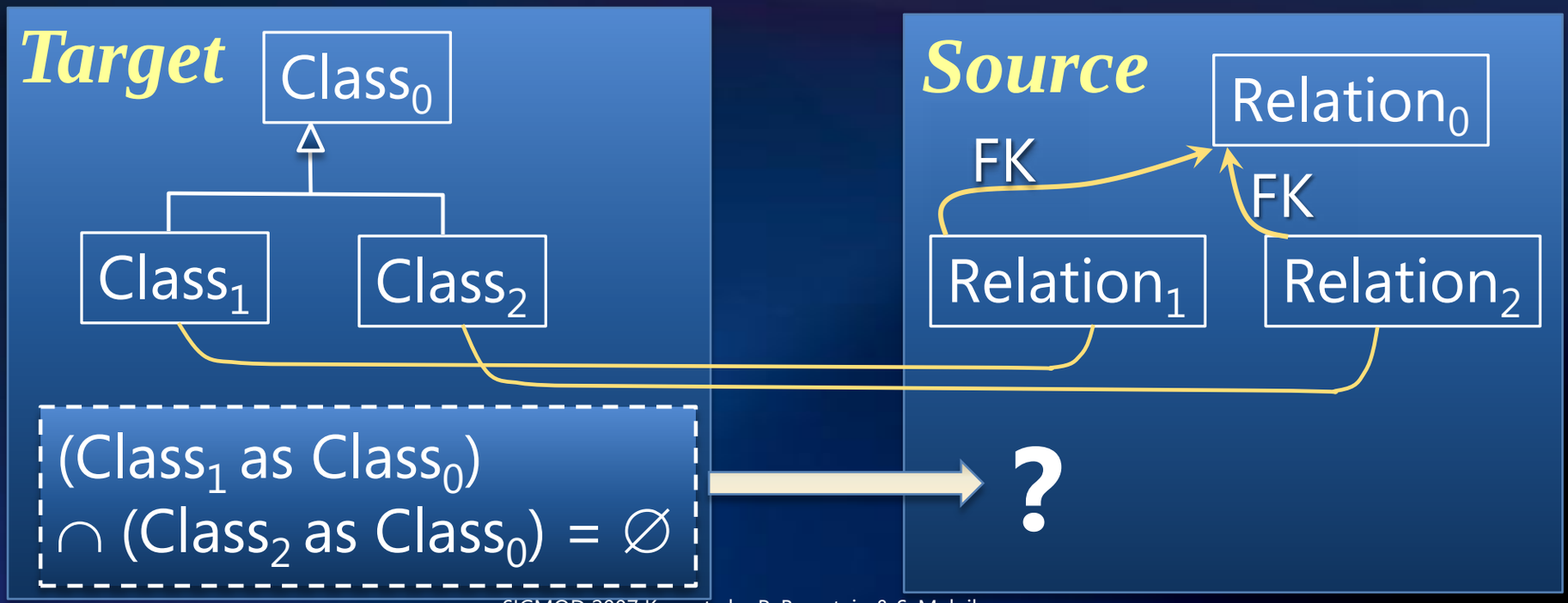
- Queries
- Updates
- Peer-to-peer
- Provenance
- Access Control

- Integrity constraints
- Synch logic
- Business logic
- Debugging

- Errors
- Indexing
- Notifications
- Batch loading
- Data exchange

Mapping Runtime Examples

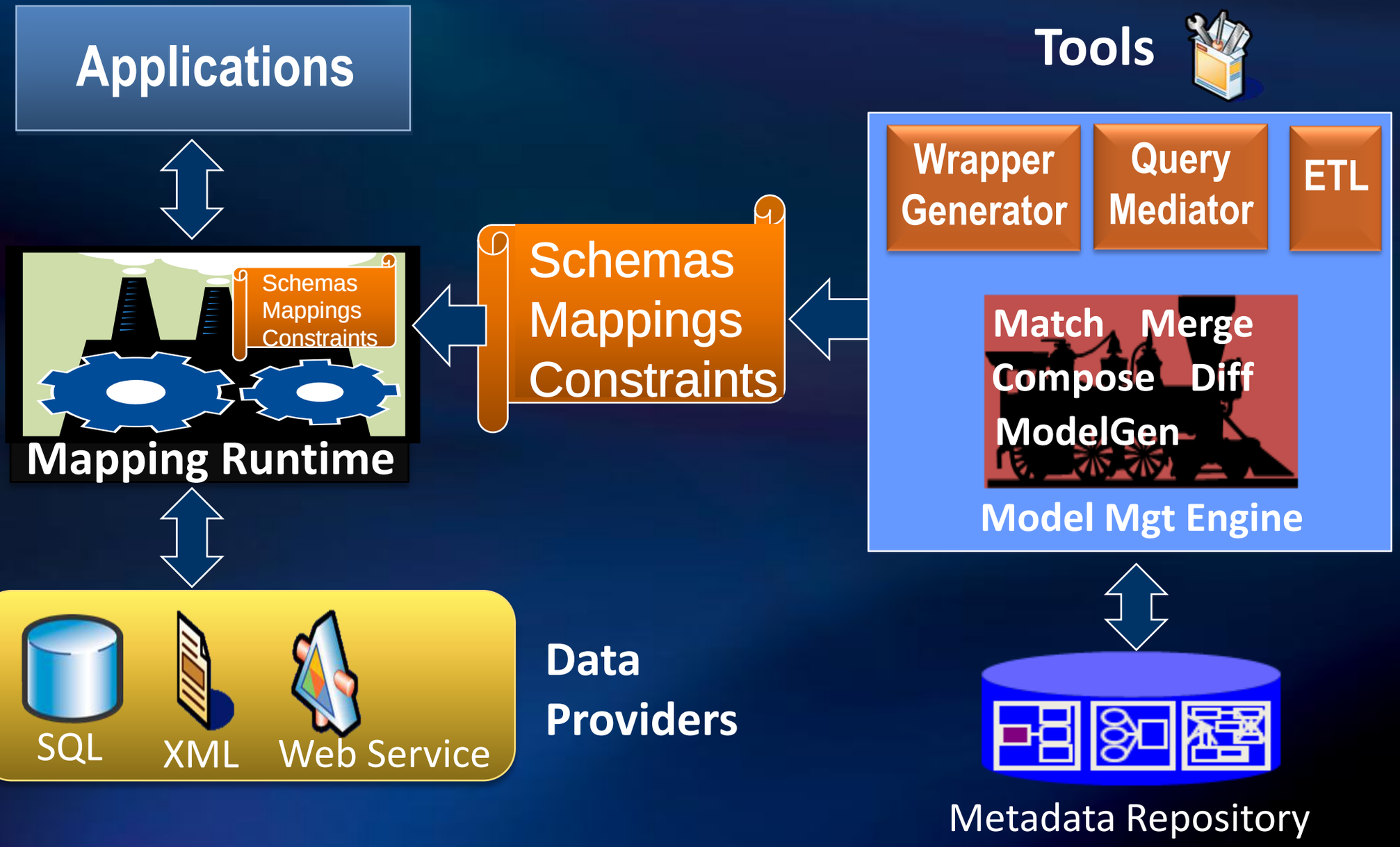
- Integrity constraints
  - Integrity constraints on target T are enforced by a combination of constraints enforced by the source and by the target runtime.
  - Feasibility - some constraints on T may not be expressible in source S.



Mapping Runtime Examples (2)

- Provenance [Cui, Widom, Weiner. TODS 00] [Baghwat, Chiticariu, Tan, Vijayvargia. VLDB J. 05] [Buneman, Chapman, Cheney. SIGMOD 06]
  - User moves data from source S to target T
  - Which source data contributed to a particular target data item?
- Errors
  - A data access via T is translated into an access on S that generates an error
  - The error needs to be passed back through the mapping in a form that is understandable in the context of T.

Model Management 2.0



Scenarios

1. **Create mappings**
   - Match
   - ConstraintGen
   - TransGen
   - ModelGen

2. **Evolve mappings**
   - Compose
   - Diff
   - Merge
   - Inverse

Schema Matching

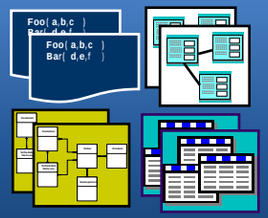


- Exploit lexical analysis of element names, schema structure, data types, thesauri, value distributions, ontologies, instances, and previous matches
- Past Goal - improved precision & recall
  - Big productivity gains are unlikely
- Better goals
  - Return top-k, not best overall match
  - Avoid the tedium. Manage work.
  - HCI – handle large schemas.
  - User studies – what would improve productivity?

Cast of Thousands

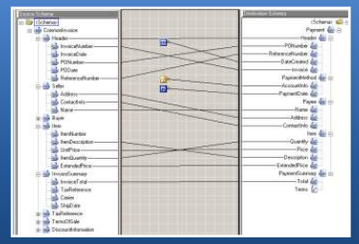
- AnHai Doan
- Alon Halevy
- Pedro Domingos
- Phil Bernstein
- Erhard Rahm
- Sergey Melnik
- Jayant Madhavan
- Jeffrey Naughton
- Jaewoo Kang
- Tova Milo
- Pavel Shvaiko
- Fausto Giunchiglia
- Sonia Bergamaschi
- Silvana Castano
- Bin He
- Kevin Chang
- Namyoun Choi
- Il-Yeol Song
- Hyoil Han
- Domenico Ursino
- Luigi Palopoli
- Dominico Sacca
- Georgio Terracina
- David Embley
- David Jackman
- Li Xu
- Yihong Ding
- Jacob Berlin
- Amihai Motro
- Hong Hai Do
- Fabien Duchateau
- Zohra Mellahsene
- Ela Hunt
- Toralf Kirsten
- Andreas Thor
- Alexander Bilke
- Avigdor Gal
- Michalis Petropoulos
- Christoph Quix
- Chris Clifton
- Arnie Rosenthal
- Wen-Syan Li
- Hector Garcia-Molina
- Sagit Zohar
- Gio Wiederhold
- Anna Zhdanova
- Jerome Euzenat
- Prasenjit Mitra
- Natasha Noy
- Anuj Jaiswal
- Mikalai Yatskevich
- Nuno Silva
- Joao Rocha
- David Aumueller
- Sabine Massmann
- Felix Naumann

Code Generation Scenarios [Miller, Haas, & Hernandez, VLDB 0

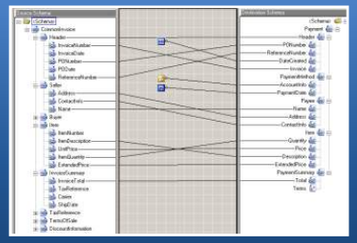


Models

Match



Correspondences



Correspondence

ConstraintGen

```
Select ord#, prod#, cust#
From  Shipped
=
Select ord#, prod#, cust#
From Order Join Item
      on ord#
```

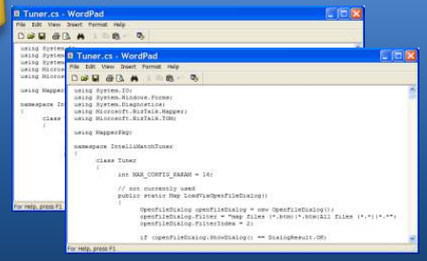
Constraints

?

```
Select ord#, prod#, cust#
From  Shipped
⊆
Select ord#, prod#, cust#
From Order Join Item
      on ord#
```

Constraints

TransGen



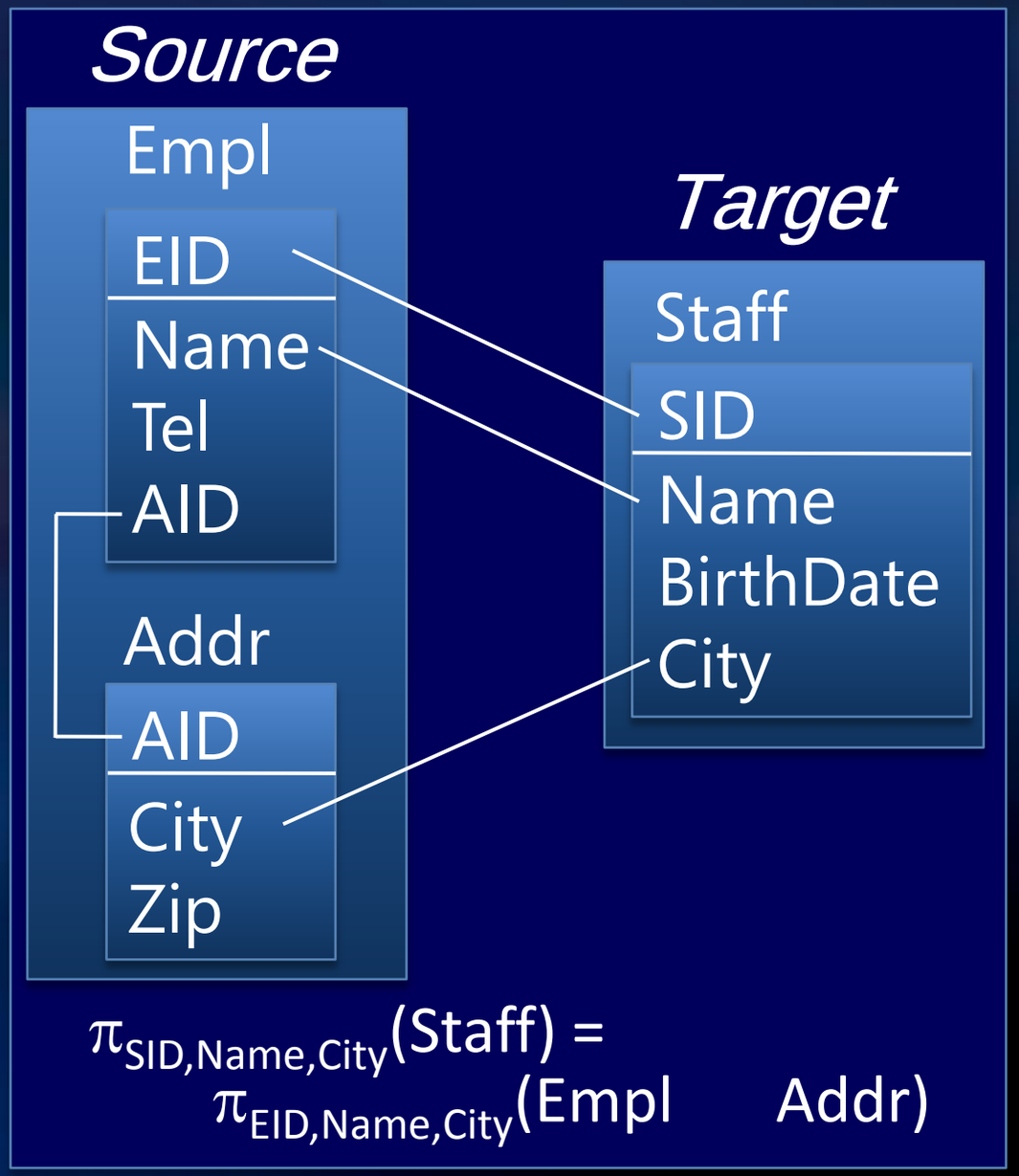
Transformations

Correspondences → Transformations

[Popa, Velegrakis, Miller, Hernandez, Fagin. VLDB 02]
[Velegrakis. PhD thesis 2005]

For a given target element

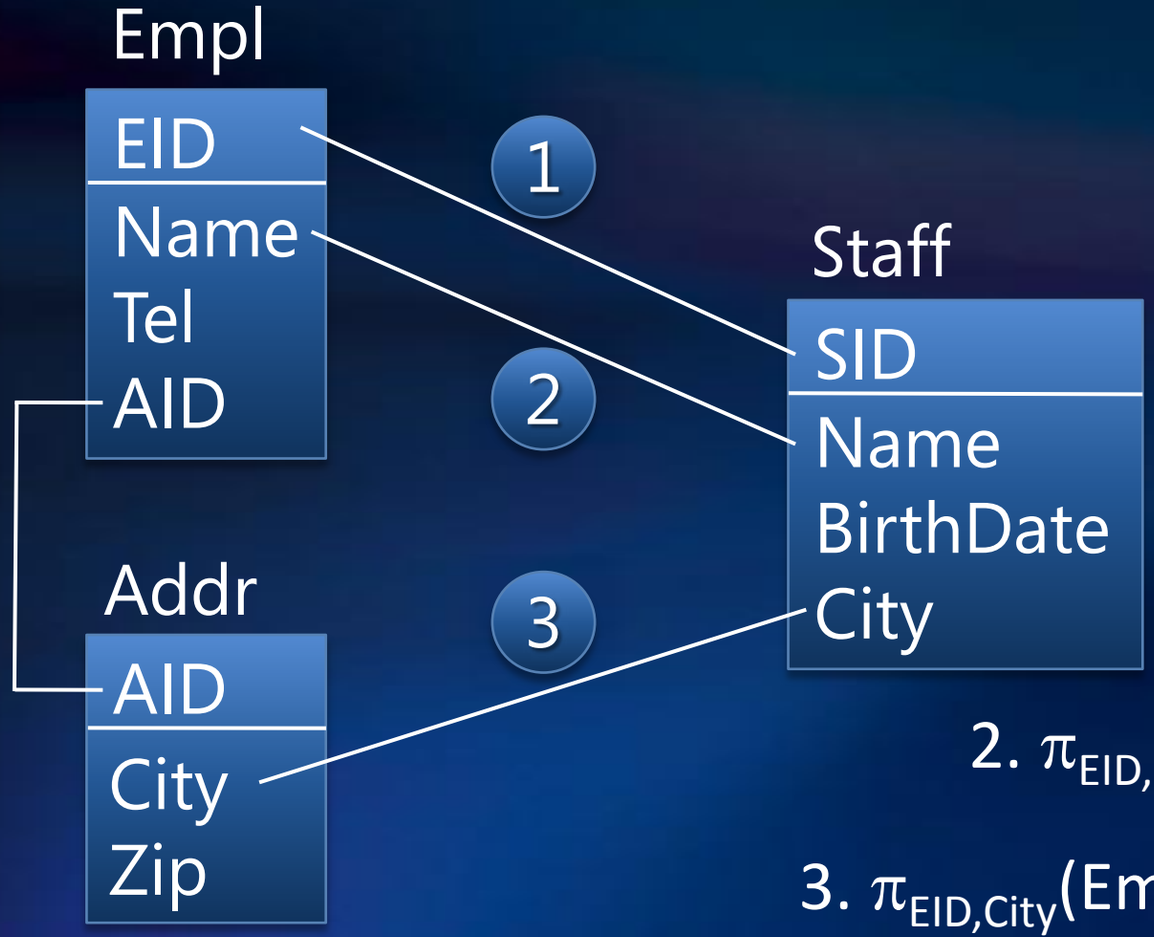
- Find all source elements linked by correspondences
- Find all ways that source elements are related
- Choose one of them and generate the transformation



Correspondences → Constraints

[Melnik, Bernstein, Halevy, Rahm, SIGMOD 05]

- Directly interpret correspondences as mapping constraints
- If it's a tree schema and keys correspond

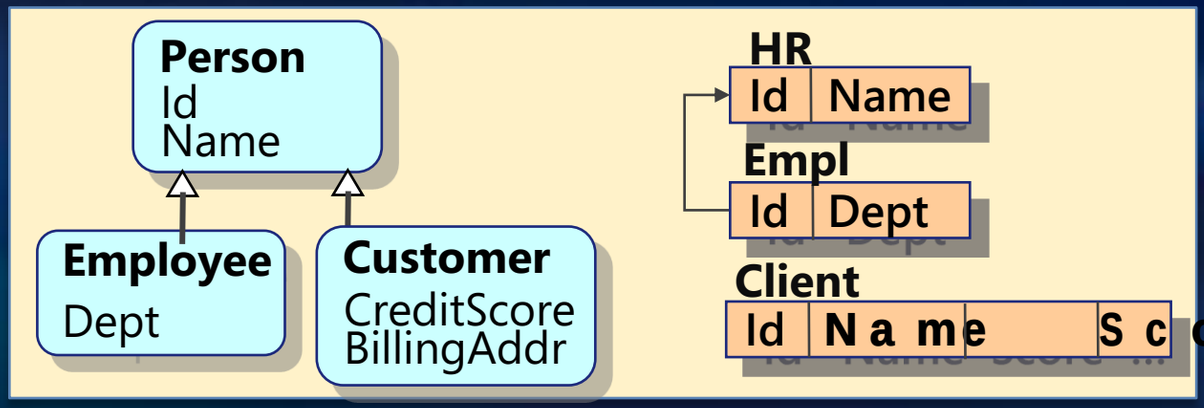


1. $\pi_{EID}(Empl) = \pi_{SID}(Staff)$

2. $\pi_{EID,Name}(Empl) = \pi_{SID,Name}(Staff)$

3. $\pi_{EID,City}(Empl \quad Addr) = \pi_{SID,City}(Staff)$

Source: EER
Target: SQL



Mapping Constraints

```
SELECT p.Id, p.Name
FROM Persons AS p
WHERE p IS OF (ONLY Person)
  OR p IS OF (ONLY Employee)
```

=

```
SELECT Id, Name
FROM dbo.HR
```

```
SELECT e.Id, e.Dept
FROM Persons AS e
WHERE e IS OF Employee
```

=

```
SELECT Id, Dept
FROM dbo.Empl
```

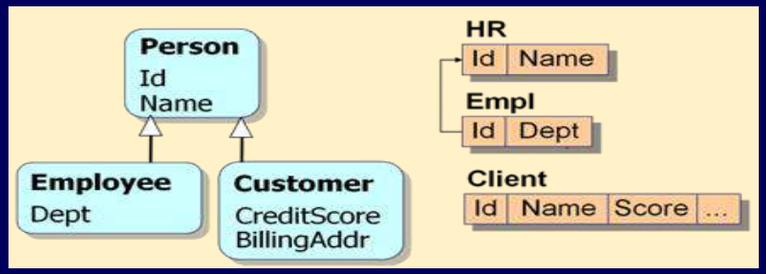
```
SELECT c.Id, c.Name,
   c.CreditScore, c.BillingAddr
FROM Persons AS c
WHERE c IS OF Customer
```

=

```
SELECT Id, Name,
          Score, Addr
FROM dbo.Client
```

Constraints → Transformations

[Melnik, Adya, Bernstein, SIGMOD 07]




```
SELECT p.Id, p.Name
FROM Persons AS p
WHERE p IS OF (ONLY Person)
  OR p IS OF (ONLY Employee)
```

=

```
SELECT Id, Name
FROM dbo.HR
```

```
SELECT e.Id, e.Dept
FROM Persons AS e
WHERE e IS OF Employee
```

=

```
SELECT Id, Dept
FROM dbo.Empl
```

```
SELECT c.Id, c.Name,
   c.CreditScore, c.BillingAddr
FROM Persons AS c
WHERE c IS OF Customer
```

=

```
SELECT Id, Name,
       Score, Addr
FROM dbo.Client
```

```
SELECT VALUE
    CASE
        WHEN (T5._from2 AND NOT(T5._from1))  THEN Person(T5.Person_Id,
T5.Person_Name)
        WHEN (T5._from1 AND T5._from2)
            THEN Employee(T5.Person_Id, T5.Person_Name,  T5.Employee_Dept)
            ELSE Customer(T5.Person_Id, T5.Person_Name,  T5.Customer_CreditScore,
                T5.Customer_BillingAddr)
    END
  FROM (   (SELECT T1.Person_Id, T1.Person_Name,  T2.Employee_Dept,
                CAST(NULL AS SqlServer.int) AS Customer_CreditScore,
                CAST(NULL AS SqlServer.nvarchar) AS Customer_BillingAddr, False AS
                 _from0,
                (T2._from1 AND T2._from1 IS NOT NULL) AS _from1,  T1._from2
            FROM  ( SELECT  T.Id AS Person_Id, T.Name AS Person_Name, True AS
                 _from2
                FROM HR AS T) AS T1
                LEFT OUTER JOIN (
                  SELECT  T.Id AS Person_Id,  T.Dept AS Employee_Dept, True AS
                   _from1
                  FROM dbo.Empl AS T) AS T2
                ON T1.Person_Id = T2.Person_Id )
            UNION ALL (
                SELECT  T.Id AS Person_Id,  T.Name AS Person_Name,
                    CAST(NULL AS SqlServer.nvarchar) AS Employee_Dept,
                    T.Score AS Customer_CreditScore,  T.Addr AS
Customer_BillingAddr,
                    True AS _from0,  False AS _from1,  False AS _from2
                FROM Client AS T)
        ) AS T5
```

Constraints → Transformations (2)

- Difficulty depends on
  - Whether the constraints are functions
  - The transformation language (e.g., SQL, XSLT)
  - Expressiveness of constraints
  - Optimization required

Scenarios

1. Create mappings

- ✓ Match
- ✓ ConstraintGen
- ✓ TransGen
- **ModelGen**

2. Evolve mappings

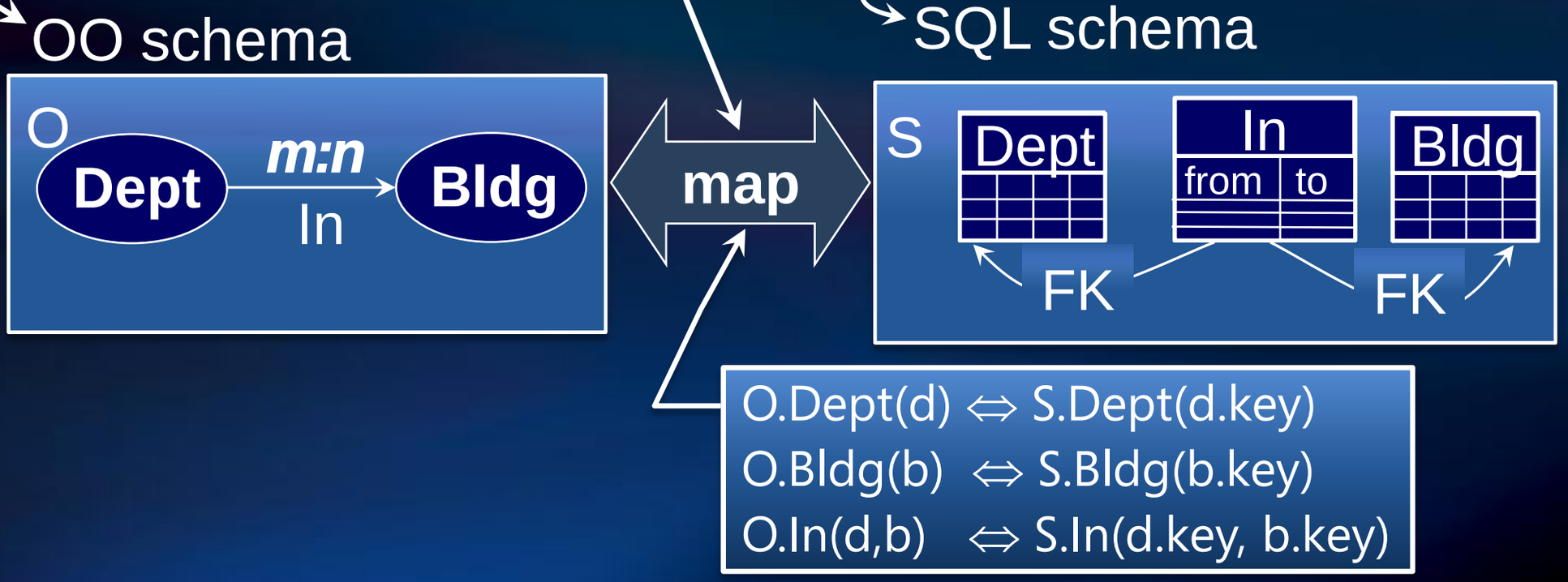
- Compose
- Diff
- Merge
- Inverse

ModelGen: Schema Translation

Input
- source model
- target metamodel

Output
- target model
- constraints

[Atzeni, Torlone. EDBT 96]
[Bernstein, Melnik, Mork. VLDB 05]
[Atzeni, Cappellari, Bernstein. EDBT 06]



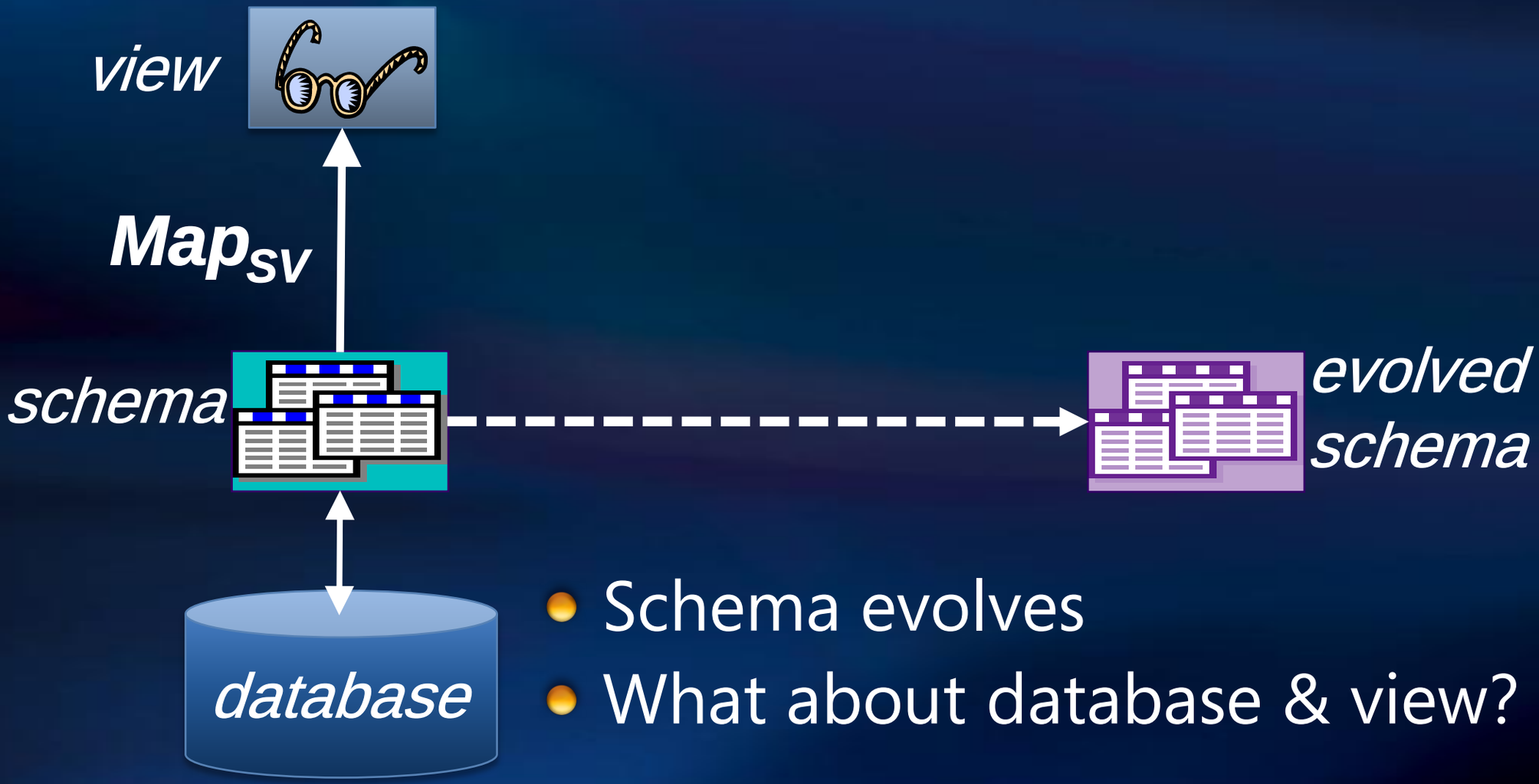
- There are several credible prototypes
  - Don't know of products, yet

Scenarios

1. Create mappings
   - Match
   - ConstraintGen
   - TransGen
   - ModelGen

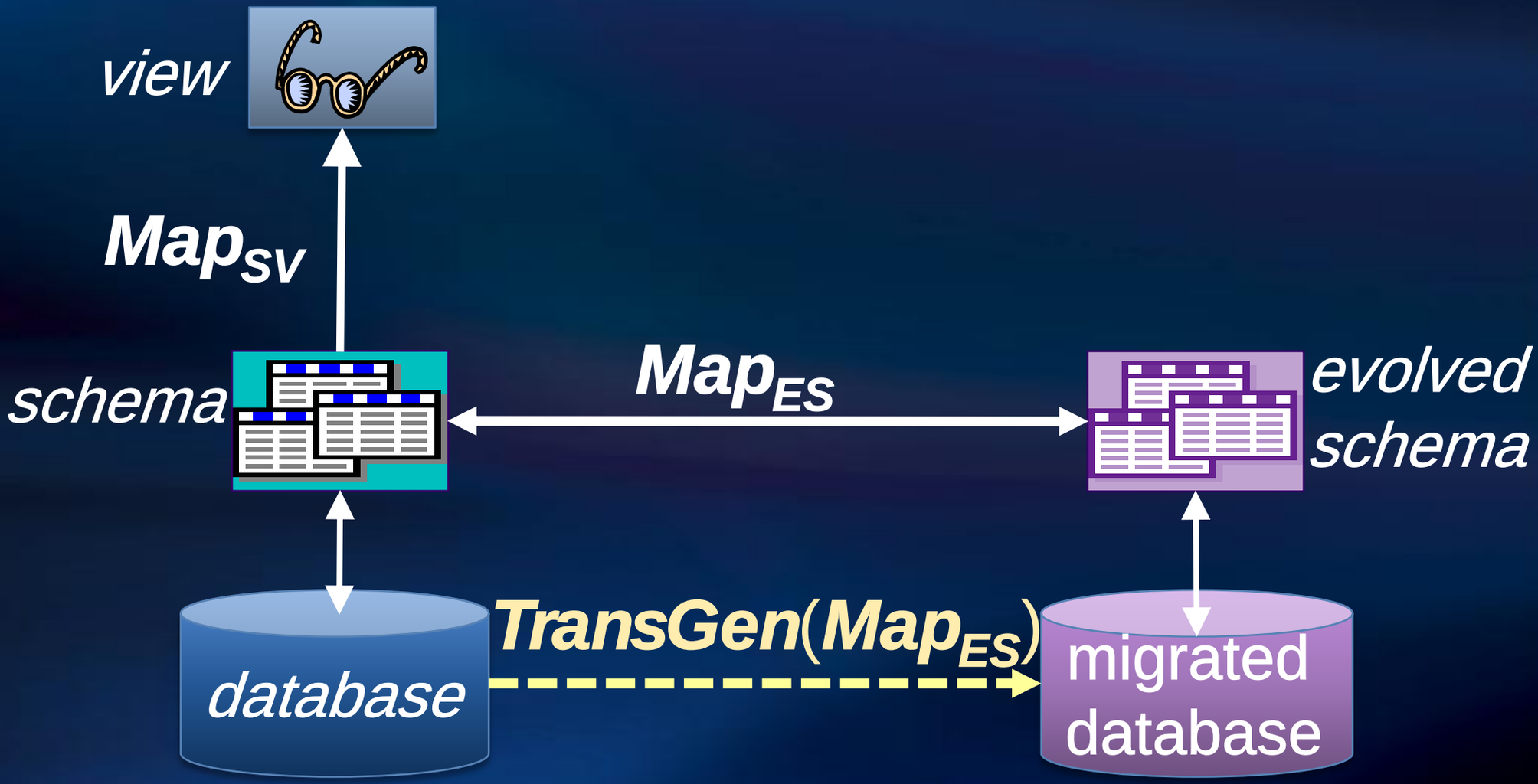
2. **Evolve mappings**
   - Compose
   - Diff
   - Merge
   - Inverse

Schema Evolution [Rahm, Bernstein. SIGMOD Rec. Dec 06]



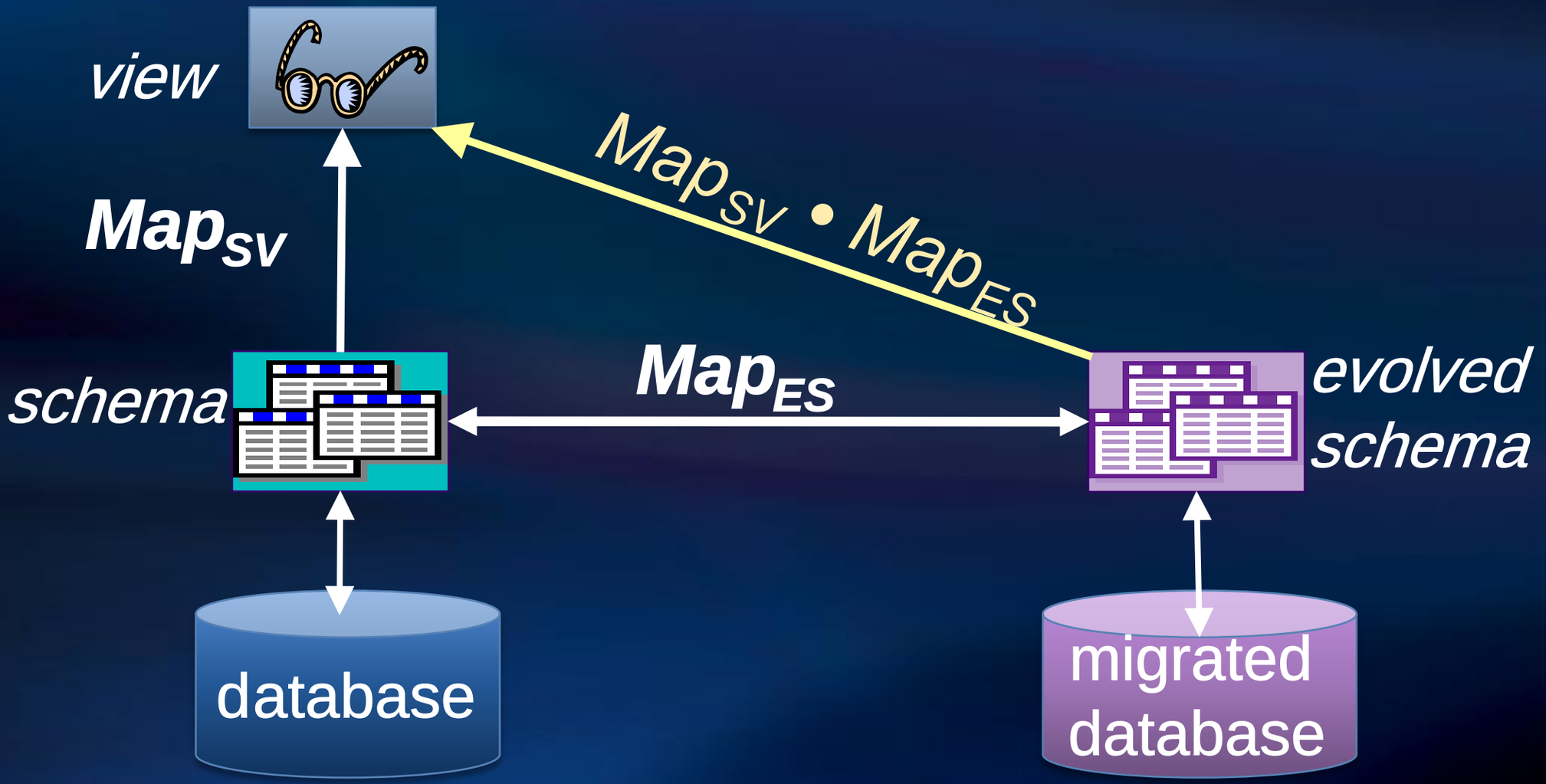
- Schema evolves
- What about database & view?

Data Migration



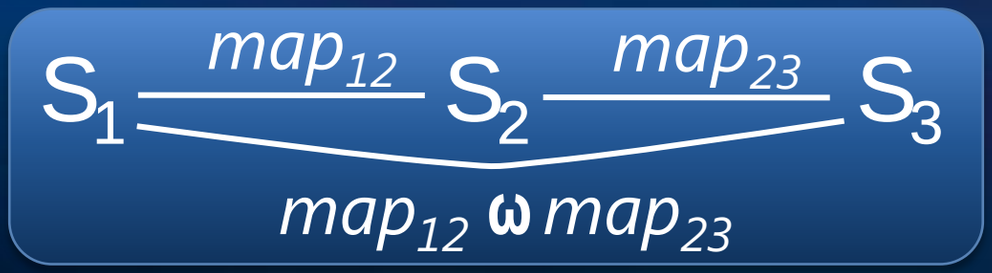
1. Create mapping: *schema* ⇔ *evolved schema*
2. Generate a transformation

View Migration



- Compose Map_{SV} and Map_{ES} to connect *view* to *evolved schema*

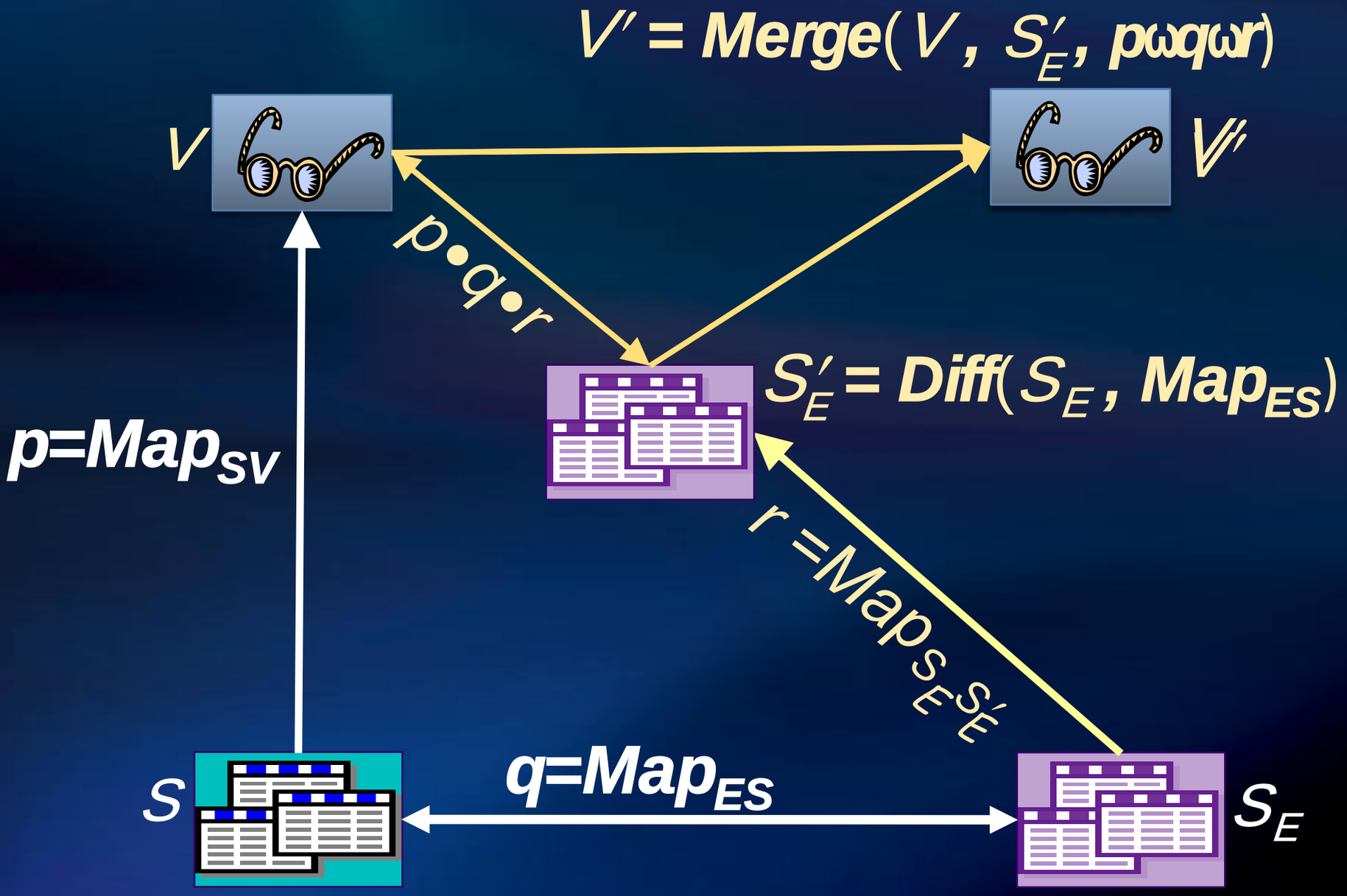
Composition



[Fagin, Kolaitis, Popa, Tan. TODS 05]
[Nash, Bernstein, Melnik. TODS 07]
[Yu, Popa. VLDB 05]
[Bernstein, Green, Melnik, Nash. VLDB 06]

- Some natural 1st-order mapping languages are not closed under composition
  - Sometimes, it's undecidable whether the composition is expressible in the input language
  - Can settle for a partial solution over 1st-order mappings
- Or you can use a 2nd-order mapping language that's closed under composition
  - There's a composition algorithm to compute it
- Some prototype implementations reported
  - Practical applications needed

Augment View with S_E's new data



Extract & Diff

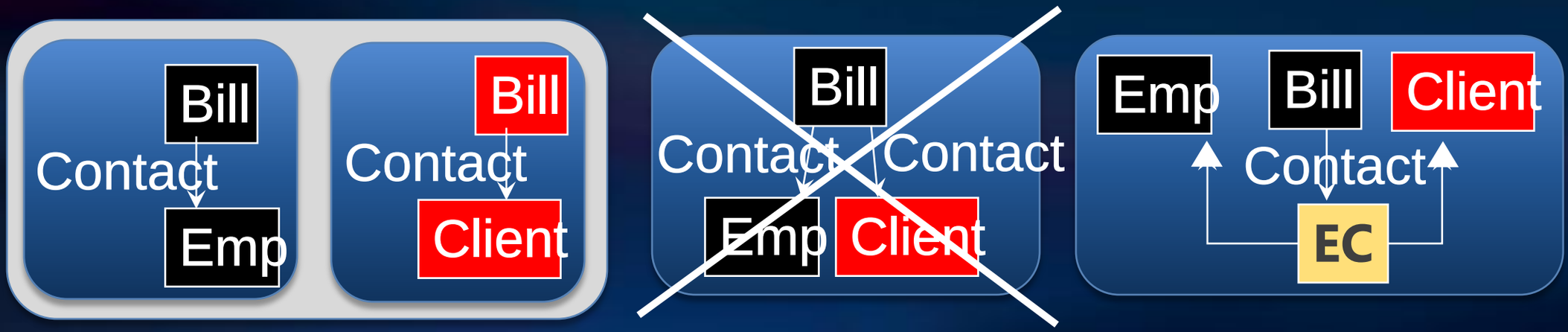


- [S, $map_{S,S'}$] = Extract(S′, $map_{S'-S}$)
  - S is a maximal sub-schema of S′ that can be populated with data from S via $map_{S'-S}$
  - Related to the materialized view selection problem: S is the minimal view needed to populate S
- Diff(S′, $map_{S'-S}$) is the complement of Extract
  - An algorithm for select-join views is in [Lechtenbörger, Vossen. TODS 03]

Merge

[Casanova, Vidal. PODS 83] [Spaccapietra, Parent. TKDE 94]
[Biskup, Convent. SIGMOD 86] [Pottinger, Bernstein. VLDB 03]
[Buneman, Davidson, Kosky. EDBT 92]

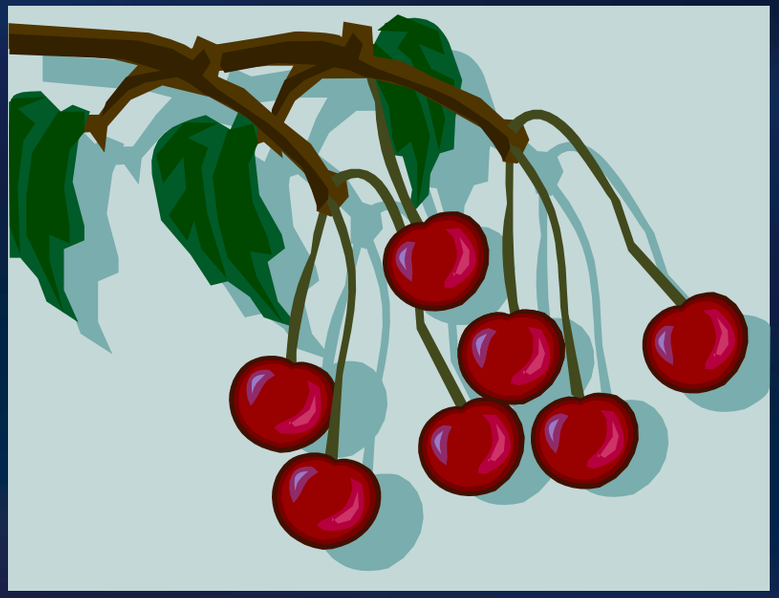
- Take disjoint union of schemas and constraints and then optimize
- Merge algorithms for structural mappings



- Extension: input map is a first-class model
- Not much known for semantic mappings

Low Hanging Fruit

- More surveys
  - Solutions to data programability problems
  - Products that address these problems (e.g. runtimes)
- More case studies
  - Using published solutions and products to solve mapping problems



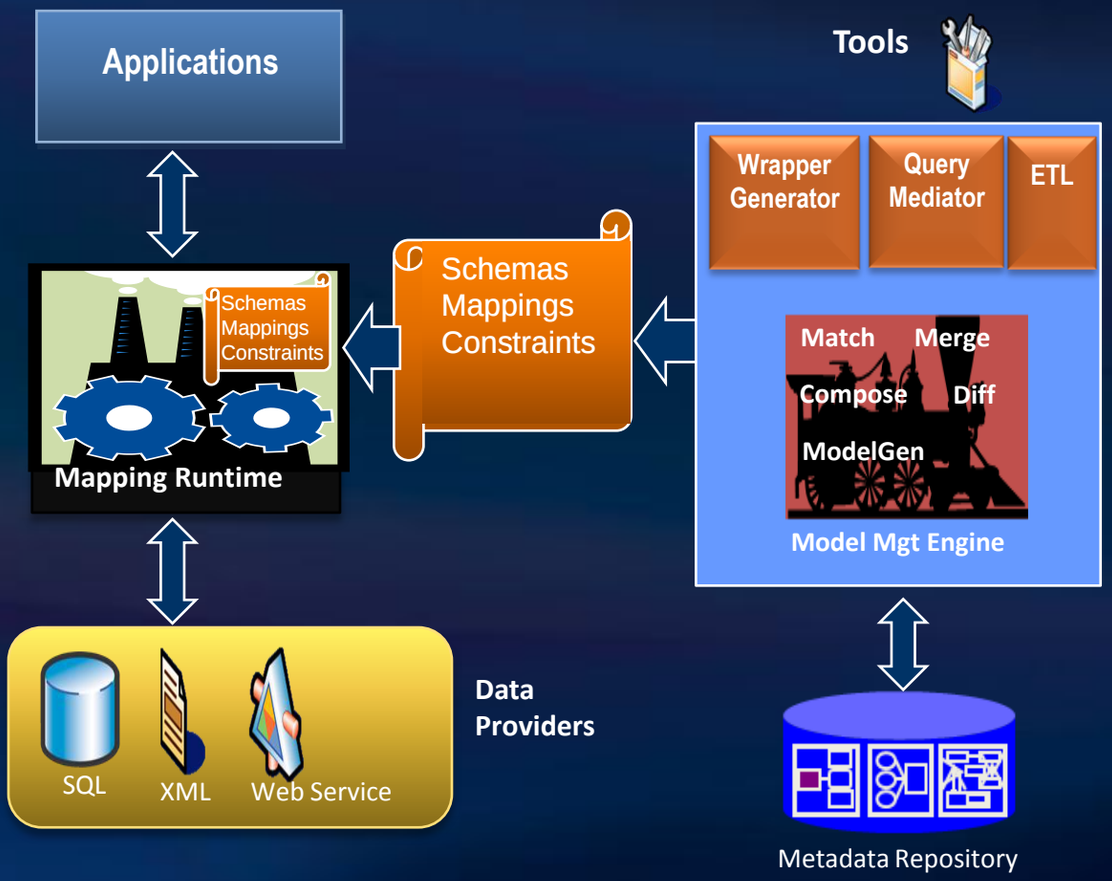
Other Challenges

- Semantics and algorithms of operators with more expressive mappings
- Translating behavior on target via mapping to behavior on source



Model Management System

- Is it still a goal to build a MMS?
- Or is it just a set of techniques to be applied?



Summary

- There's a big market looking for solutions
- Limited known about run-time scenarios
  - Mostly just for queries
  - Some updates, provenance, integrity constraints
  - Much work needed for synch logic, errors, indexing, notifications, batch loading, ….
- There's progress on many operators
  - But it's incomplete
  - For mappings with limited expressiveness
  - Little known about merge, diff, extract

Microsoft Research

Other Sessions

- Industrial Session 2: **Tuesday, 1:30pm**, Room 201-B
  - Anatomy of the ADO.NET Entity Framework
  - Atul Adya, José Blakeley, Sergey Melnik, S. Muralidhar, the ADO.NET Team (Microsoft)
- Demo Group 1: **Tuesday, 3:30-5:30pm**, Convent'n Hall No.2 2E
  - ADO.NET Entity Framework : Raising the Level of Abstraction in Data Programming
  - Pablo Castro, Sergey Melnik, Atul Adya (Microsoft)
- Research Session 11: **Wednesday, 10AM**, Room 308
  - Compiling Mappings to Bridge Applications and Databases
  - Sergey Melnik, Atul Adya, Philip A. Bernstein (Microsoft)
- Industry Session 7: **Thursday, 3:30 ð 5:30**, Room 201-B
  - The Microsoft Data Platform
  - David Campbell, Anil Nori



© 2007 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.