Intelligent Systems (AI-2)

Computer Science cpsc422, Lecture 33

Nov, 30, 2016

Slide source: from David Page (MIT) (which were from From Lise Getoor, Nir Friedman, Daphne Koller, and Avi Pfeffer) and from Lise Getoor

Lecture Overview

- **Recap Motivation and Representation for Probabilistic Relational Models (PRMs)**
  - Full Relational Schema and its Instances
  - Relational Skeleton and its Completion Instances
- Probabilistic Model of PRMs
  - Dependency Structure
  - Parameters

How PRMs extend BNs?

1. PRMs conceptually extend BNs to allow the specification of a probability model for *classes of objects* rather than a fixed set of simple attributes

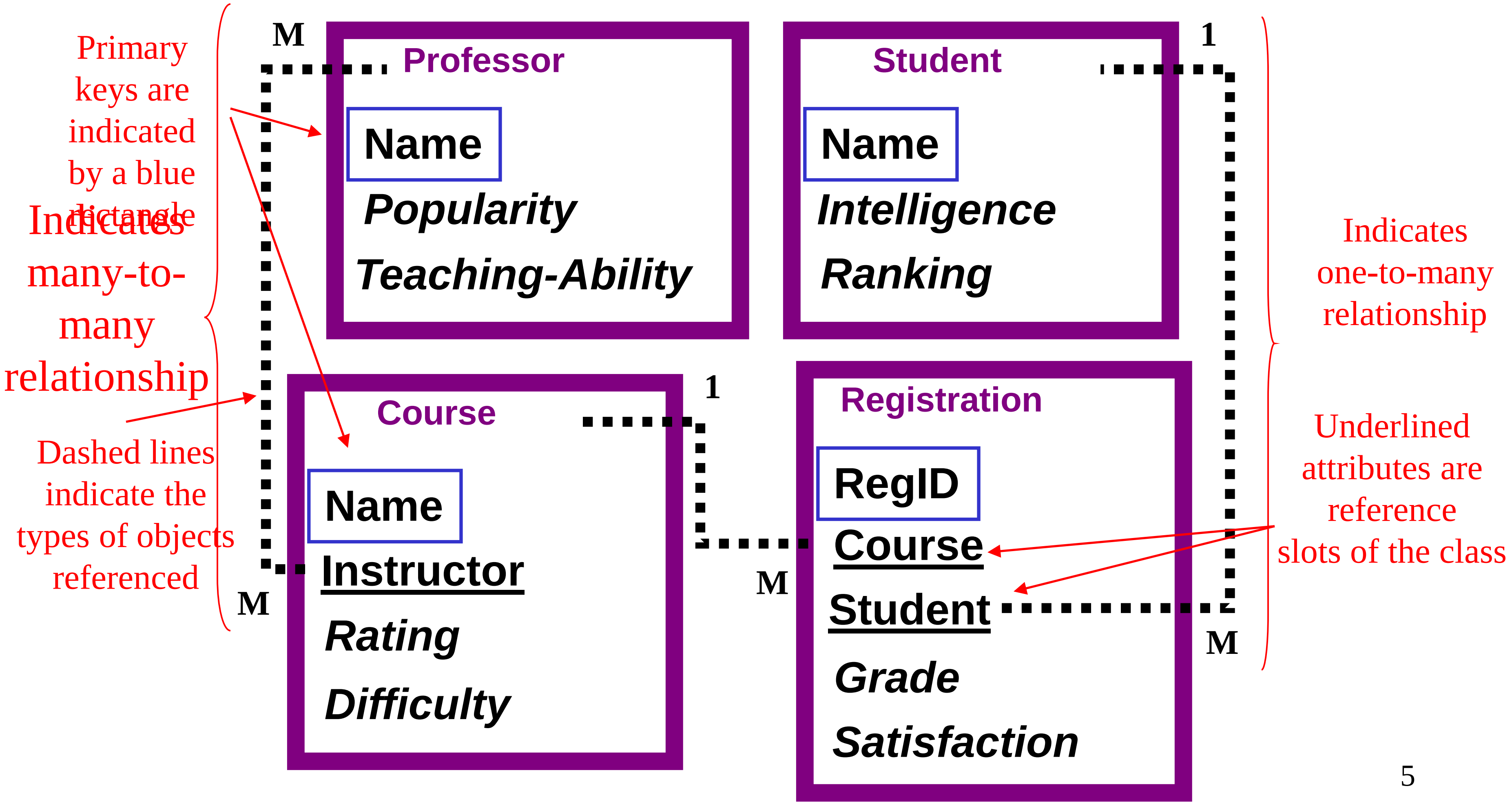
2. PRMs also allow properties of an entity to depend probabilistically on properties of other *related entities*

Mapping PRMs from Relational Models

- The representation of PRMs is a direct mapping from that of relational databases

- *A* relational model consists of a set of *classes* $X_1, \ldots, X_n$ and a set of *relations* $R_1, \ldots, R_m$, where each relation R_i is typed

University Domain Example - Full Relational Schema



Primary keys are indicated by a blue rectangle

Indicates many-to-many relationship

Dashed lines indicate the types of objects referenced

M

Professor

Name

Popularity

Teaching-Ability

1

Student

Name

Intelligence

Ranking

Indicates one-to-many relationship

Course

Name

Instructor

Rating

Difficulty

1

M

M

Registration

RegID

Course

Student

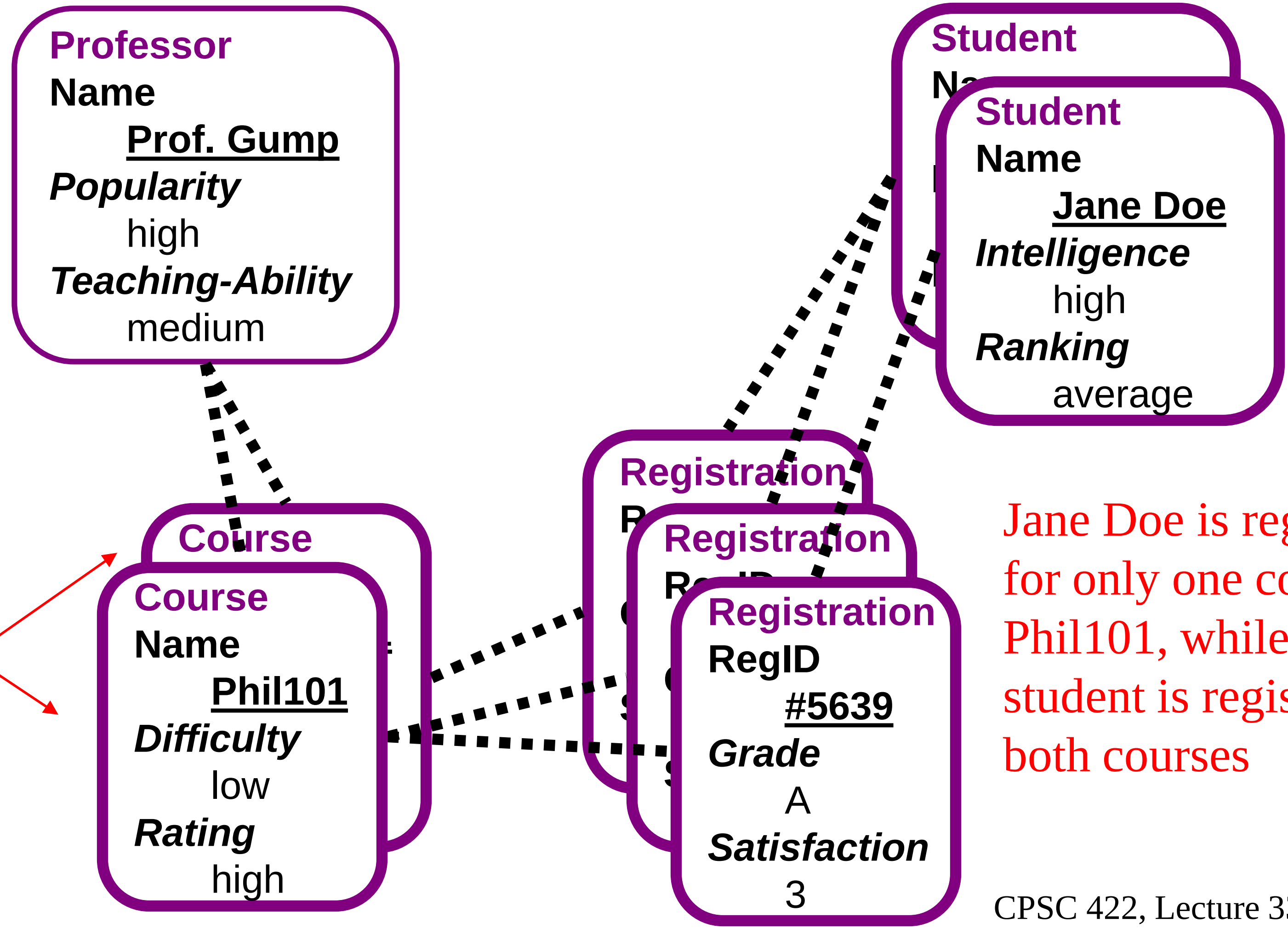
Grade

Satisfaction

M

Underlined attributes are reference slots of the class

University Domain Example - An Instance of the Schema



Professor
Name
Prof. Gump
Popularity
high
Teaching-Ability
medium

Course
Name
Phil101
Difficulty
low
Rating
high

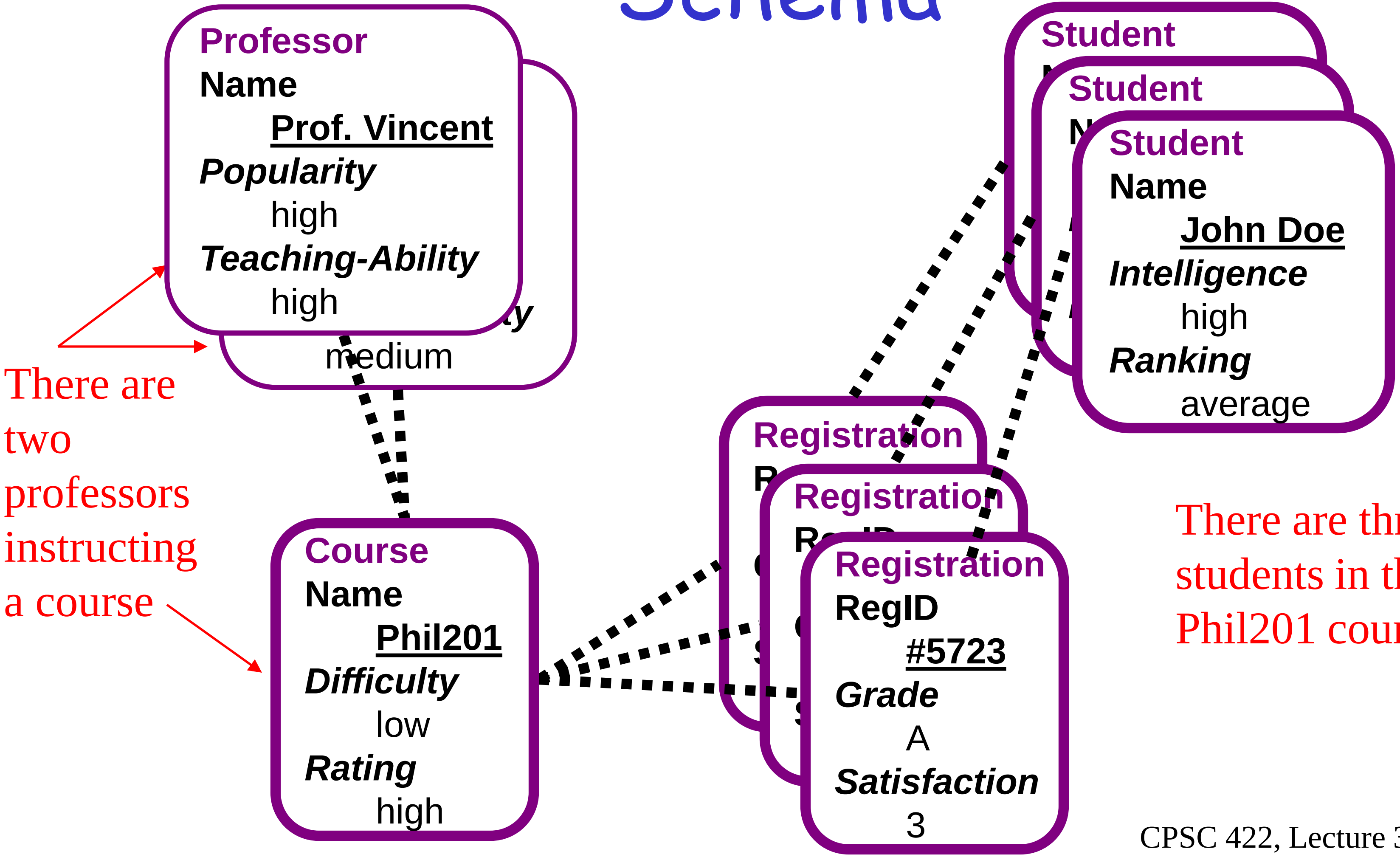
Student
Name
Jane Doe
Intelligence
high
Ranking
average

Registration
RegID
#5639
Grade
A
Satisfaction
3

One professor is the instructor for both courses

Jane Doe is registered for only one course, Phil101, while the other student is registered for both courses

University Domain Example - Another Instance of the Schema



Professor
Name
Prof. Vincent
Popularity
high
Teaching-Ability
high

medium

Course
Name
Phil201
Difficulty
low
Rating
high

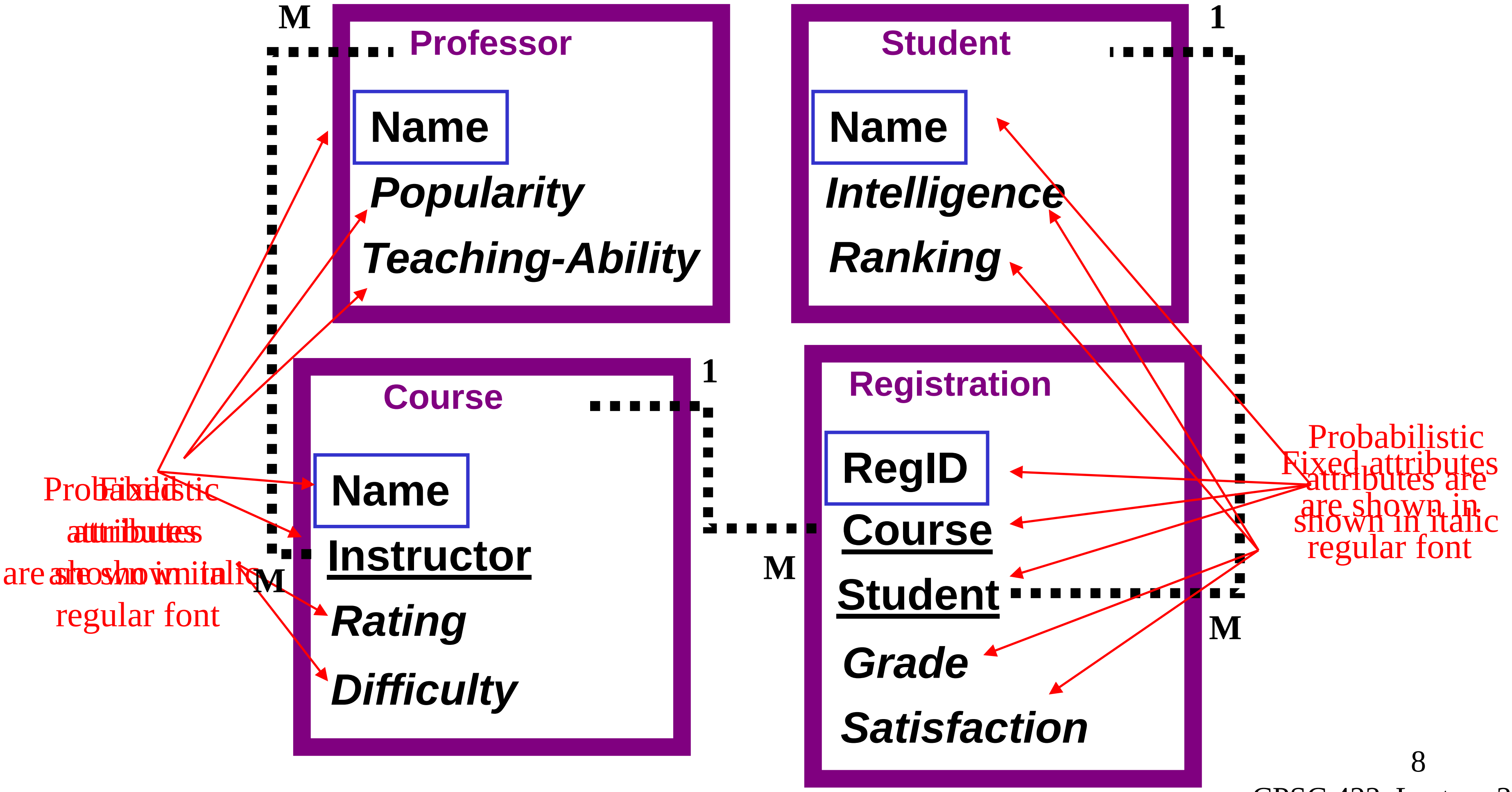
Student
Student
Student
Name
John Doe
Intelligence
high
Ranking
average

Registration
Registration
Registration
RegID
#5723
Grade
A
Satisfaction
3

There are two professors instructing a course

There are three students in the Phil201 course

University Domain Example - fixed vs. probabilistic attributes

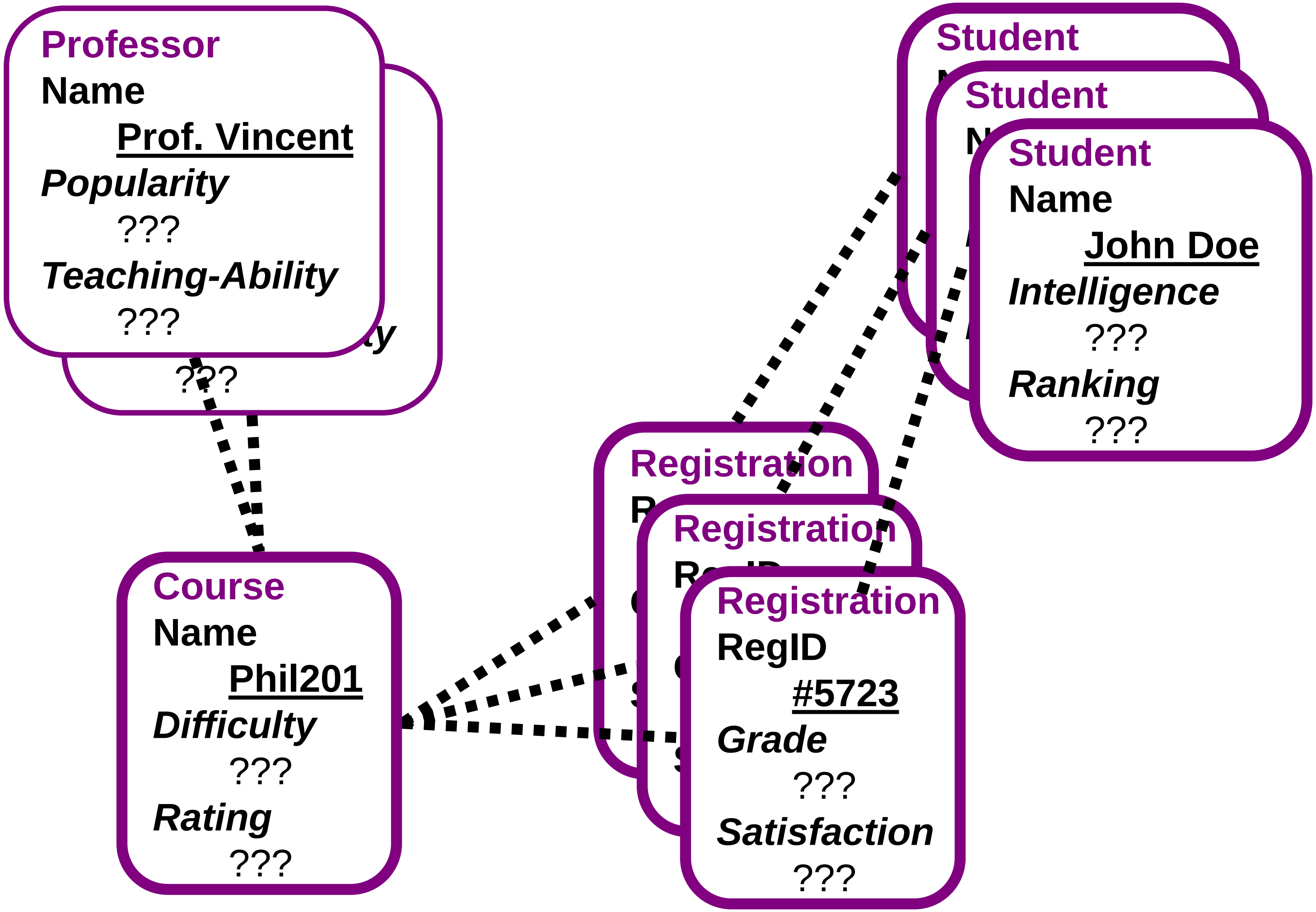


PRM Semantics: Skeleton Structure

- A *skeleton structure σ* of a relational schema is a partial specification of an instance of the schema. It specifies
  - set of objects for each class,
  - values of the fixed attributes of these objects,
  - relations that hold between the objects
- The values of probabilistic attributes are left unspecified

- A *completion I* of the skeleton structure σ extends the skeleton by also specifying the values of the probabilistic attributes

University Domain Example - Relational Skeleton



Professor
Name
Prof. Vincent
Popularity
???
Teaching-Ability
???

ty
???

Student
N

Student
N

Student
Name
John Doe
Intelligence
???
Ranking
???

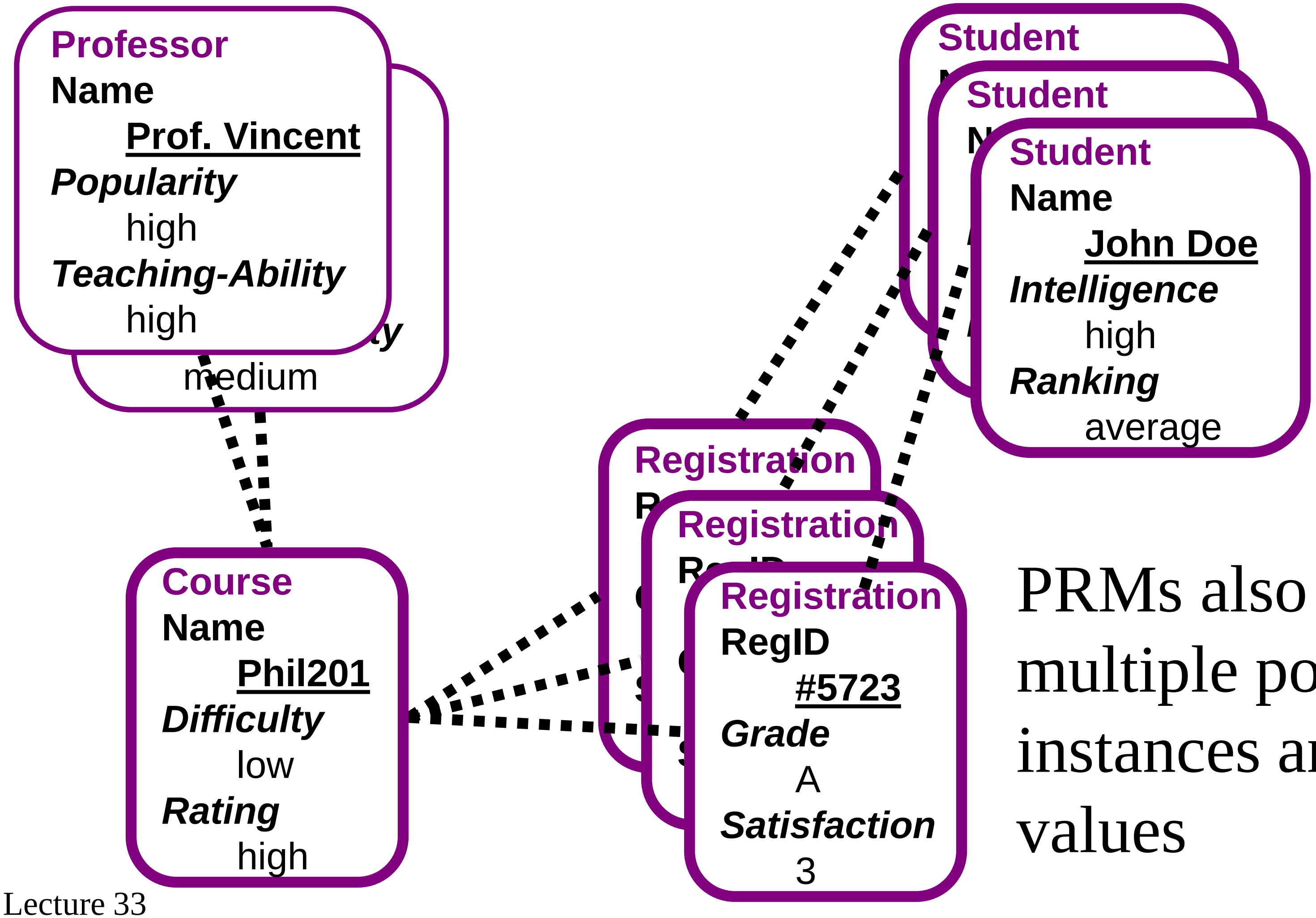
Registration
R

Registration
Re ID

Registration
RegID
#5723
Grade
???
Satisfaction
???

Course
Name
Phil201
Difficulty
???
Rating
???

University Domain Example - The Completion Instance *I*



Professor
Name
Prof. Vincent
Popularity
high
Teaching-Ability
high

medium

Student
Name
John Doe
Intelligence
high
Ranking
average

Course
Name
Phil201
Difficulty
low
Rating
high

Registration
RegID
#5723
Grade
A
Satisfaction
3

PRMs also allow multiple possible instances and values

Lecture Overview

- Recap Motivation and Representation for Probabilistic Relational Models (PRMs)
  - Full Relational Schema and its Instances
  - Relational Skeleton and its Completion Instances
- **Probabilistic Model of PRMs**
  - Dependency Structure
  - Parameters

PRMs: Probabilistic Model

- The probabilistic model consists of two components:
  - the qualitative dependency structure, $\mathcal{S}$
  - the parameters associated with it, $\theta_{\mathcal{S}}$

- The dependency structure is defined by associating with each attribute *X.A* a set of *parents* Pa(X.*A*); parents are attributes that are "direct influences" on *X.A*. This dependency holds for any object of class *X*

Dependencies within a class

The prob. attribute *X.A* can depend on another probabilistic attribute *B* of *X*. This induces a corresponding dependency for individual objects

Registration

RegID

<u>Course</u>

<u>Student</u>

Grade

Satisfaction

Registration

RegID

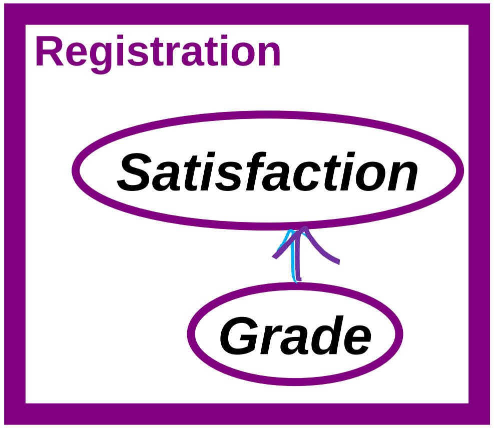
<u>#5723</u>

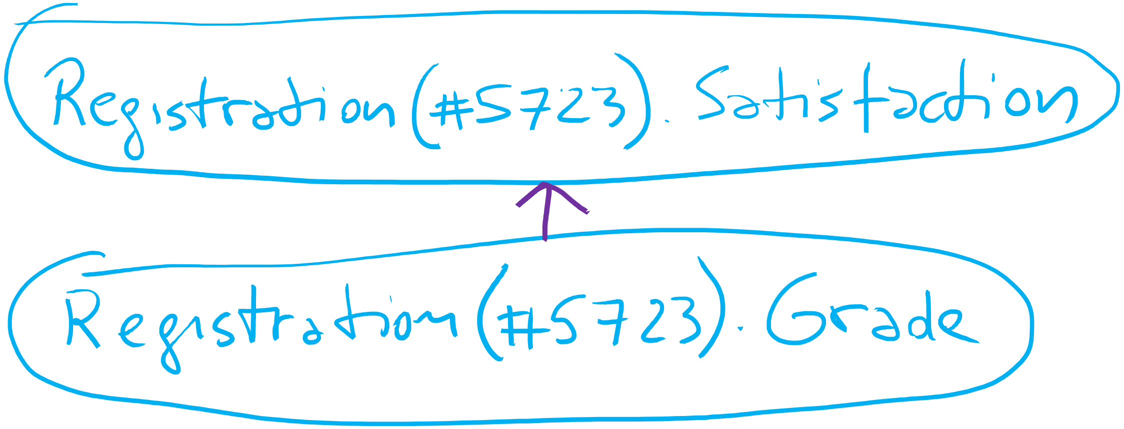
Grade

….

Satisfaction

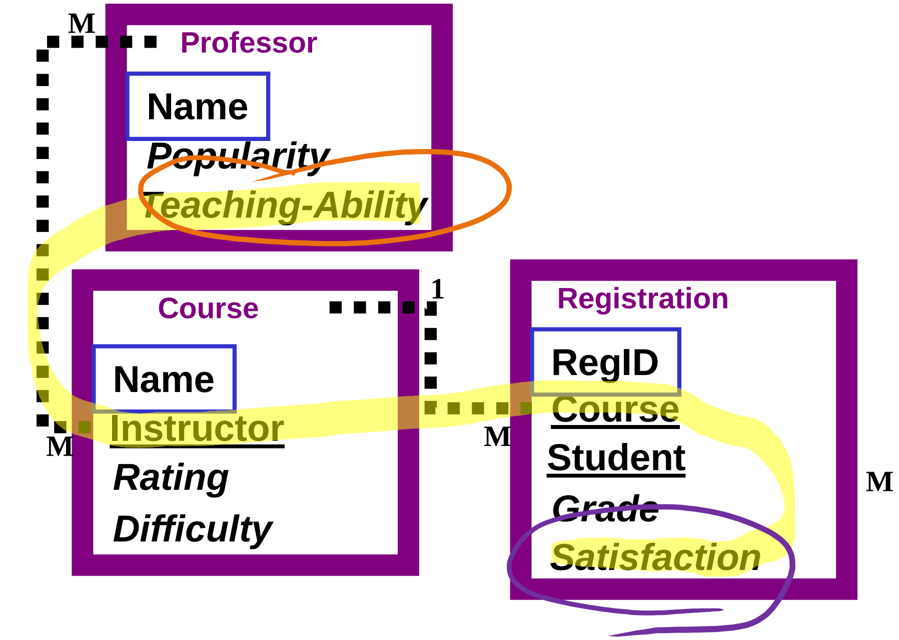
…..

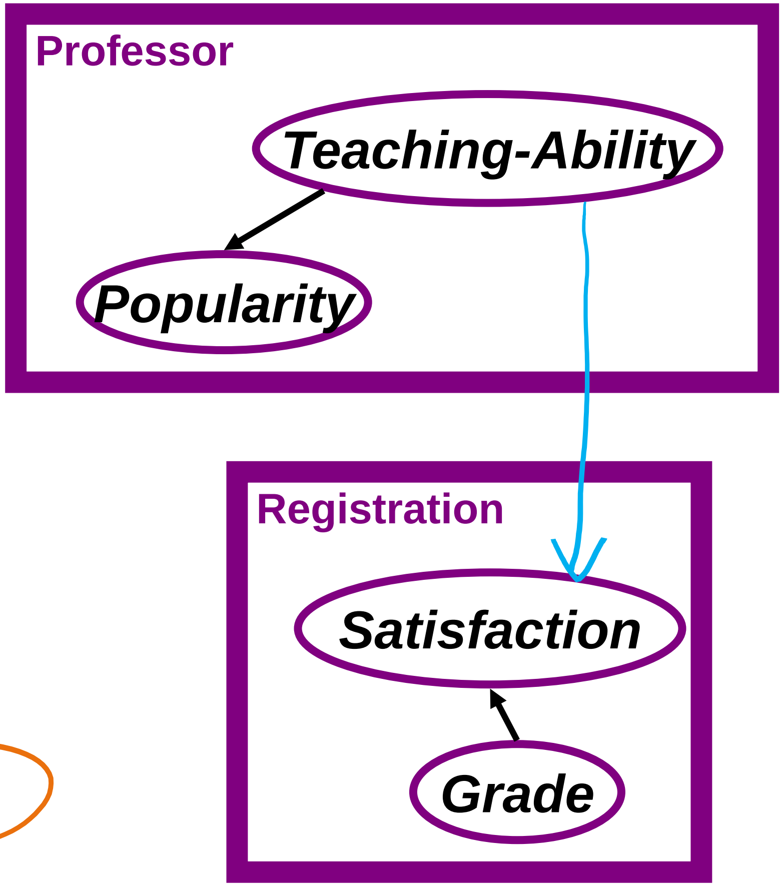




Dependencies across classes

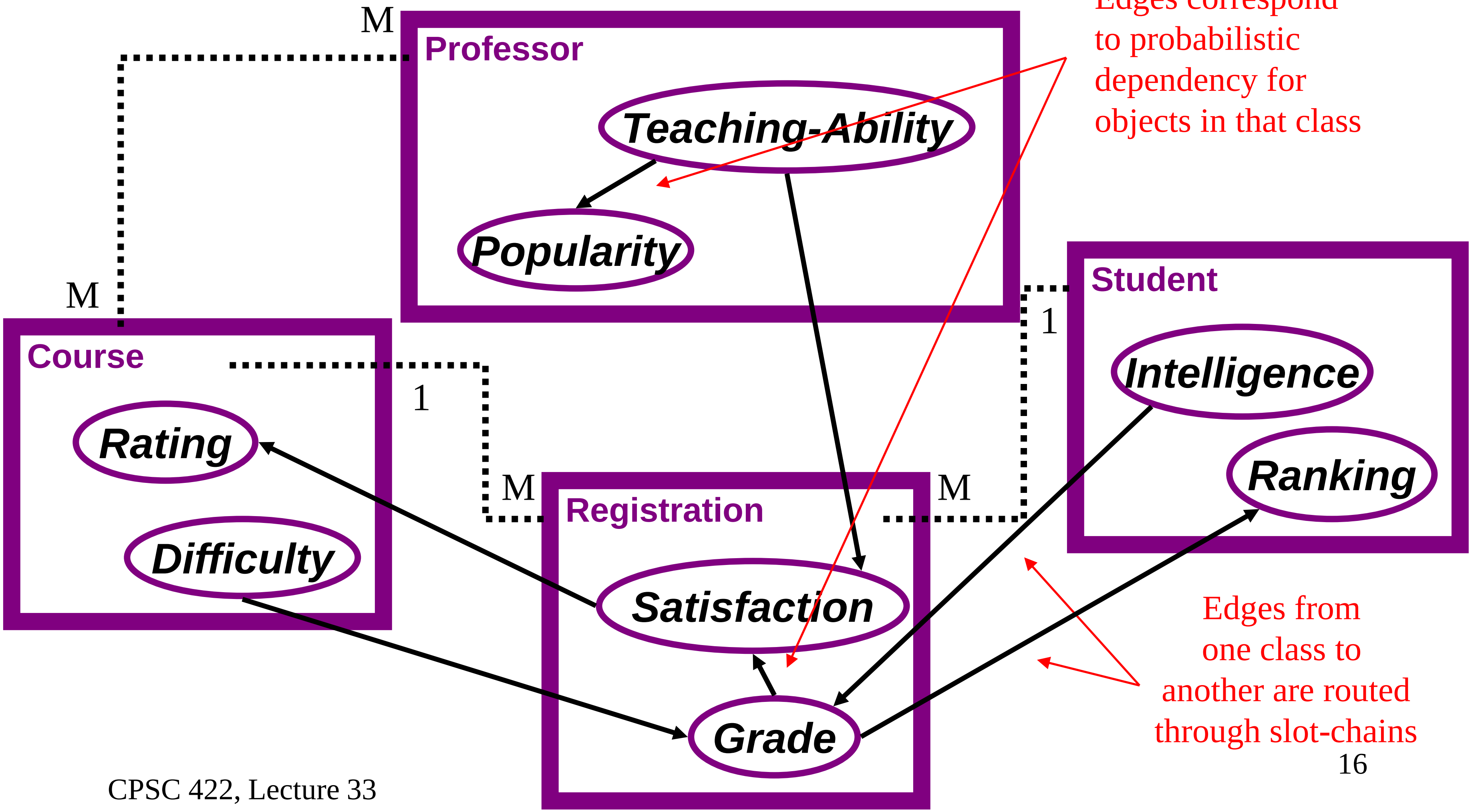
- The attribute *X.A* can also depend on attributes of related objects *X.τ.B*, where τ is a slot chain





.... Course . Instructor... ...

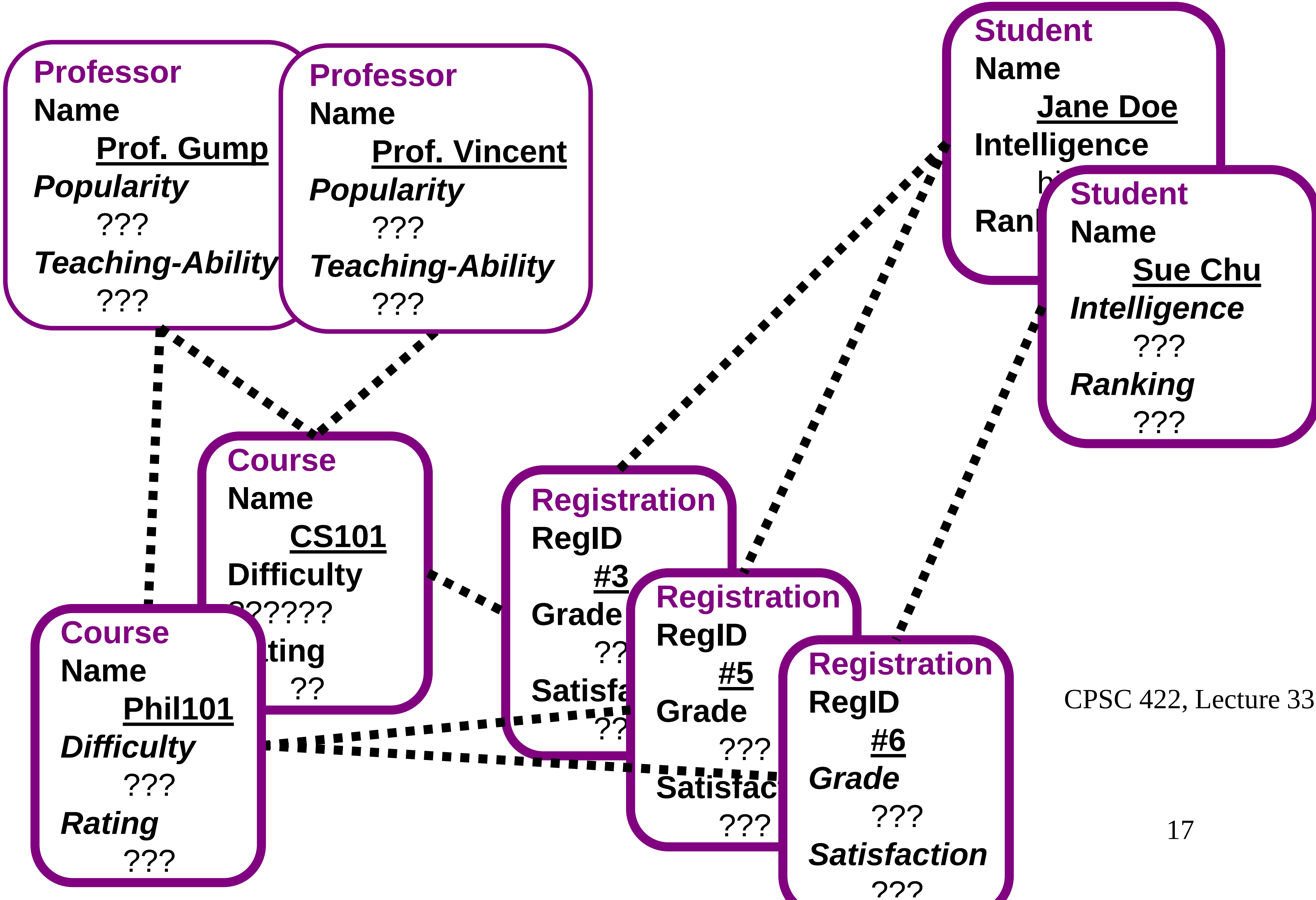
Possible PRM Dependency Structure for the University Domain



Edges correspond to probabilistic dependency for objects in that class

Edges from one class to another are routed through slot-chains

Let's derive the Corresponding "grounded" Dependency Structure for this Skeleton



Professor
Name
Prof. Gump
Popularity
???
Teaching-Ability
???

Professor
Name
Prof. Vincent
Popularity
???
Teaching-Ability
???

Student
Name
Jane Doe
Intelligence
hi
Rank

Student
Name
Sue Chu
Intelligence
???
Ranking
???

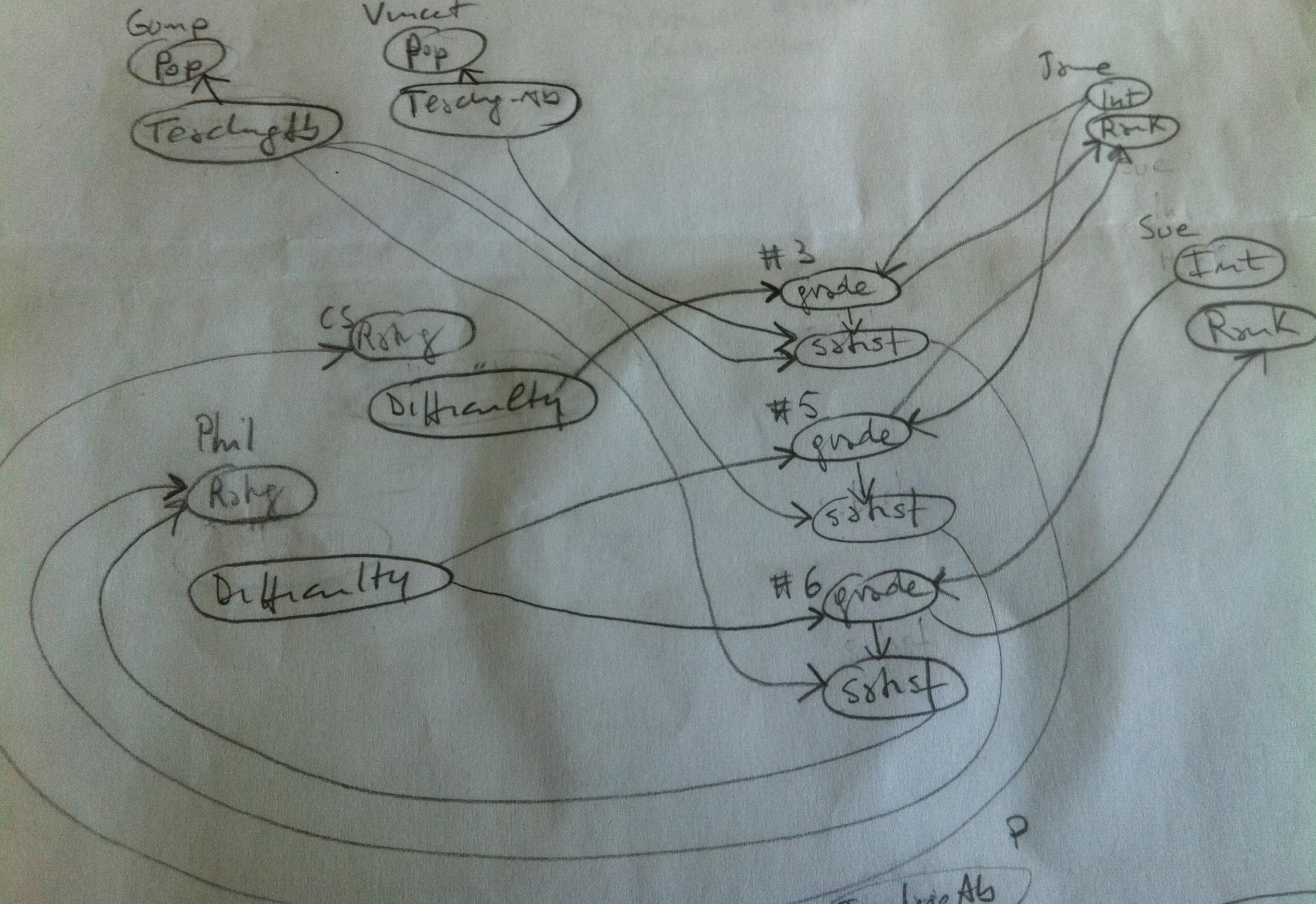
Course
Name
CS101
Difficulty
??????
ting
??

Course
Name
Phil101
Difficulty
???
Rating
???

Registration
RegID
#3
Grade
??
Satisfa
??

Registration
RegID
#5
Grade
???
Satisfac
???

Registration
RegID
#6
Grade
???
Satisfaction
???


Gump
Pop
TeachgAb
Vincet
Pop
Teachg-Ab
Jane
Int
Rank
Sue
Int
Rank
#3
grade
sathst
CS
Rtng
Difficulty
#5
grade
sathst
Phil
Rtng
Difficulty
#6
grade
sathst
P
TeachgAb

Parameters of PRMs

- **A PRM contains a *conditional probability* distribution (CPD) *P*(*X.A*|Pa(*X.A*)) for each attribute *X.A* of each class e.g.,**

P(Registration.Grade | Course.Difficulty, Student.Intelligence)

- **More precisely, let U be the set of parents of X.A. For each tuple of values u ∈ *V*(U), the CPD specifies a distribution *P*(*X.A*|u) over *V*(*X.A*).**

Course.Difficulty = {high, low}

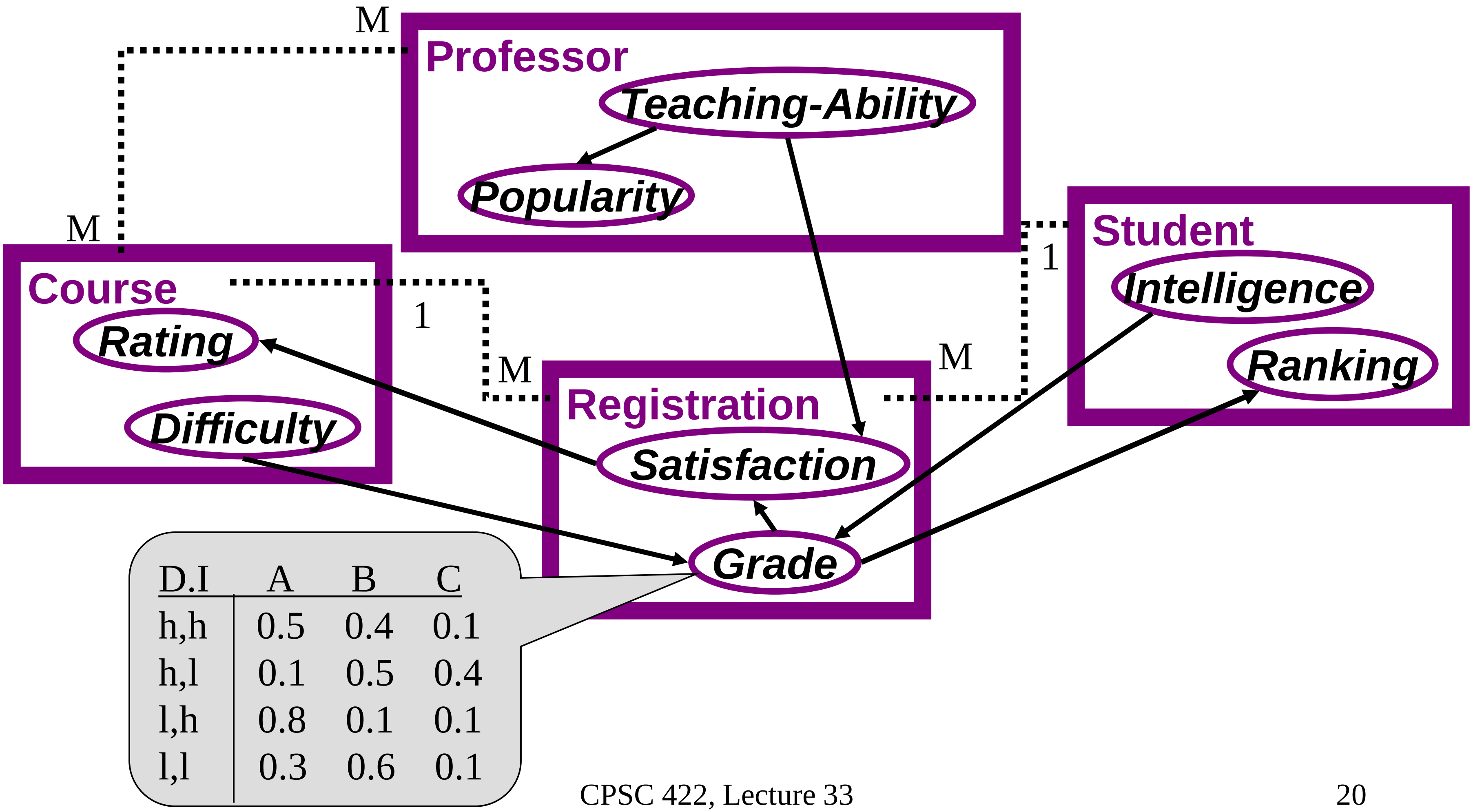
Student.Intelligence = {high, low}

Registration.Grade = {A, B, C}

D.I	A	B	C
h,h	0.5	0.4	0.1
h,l	0.1	0.5	0.4
l,h	0.8	0.1	0.1
l,l	0.3	0.6	0.1

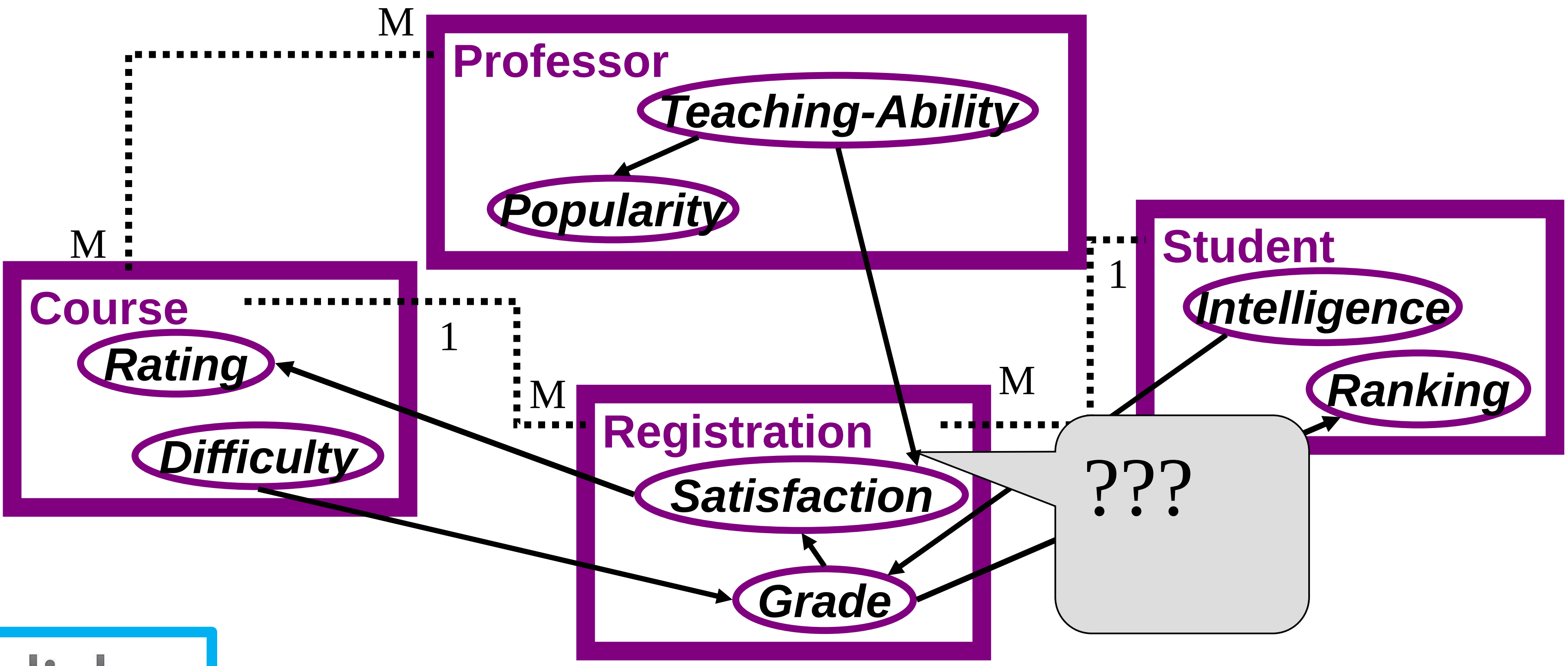
- **The parameters in all of these CPDs comprise θ_S**

Now, what are the parameters θ_S



D.I	A	B	C
h,h	0.5	0.4	0.1
h,l	0.1	0.5	0.4
l,h	0.8	0.1	0.1
l,l	0.3	0.6	0.1

Problem with some parameters θ_S



iclicker.

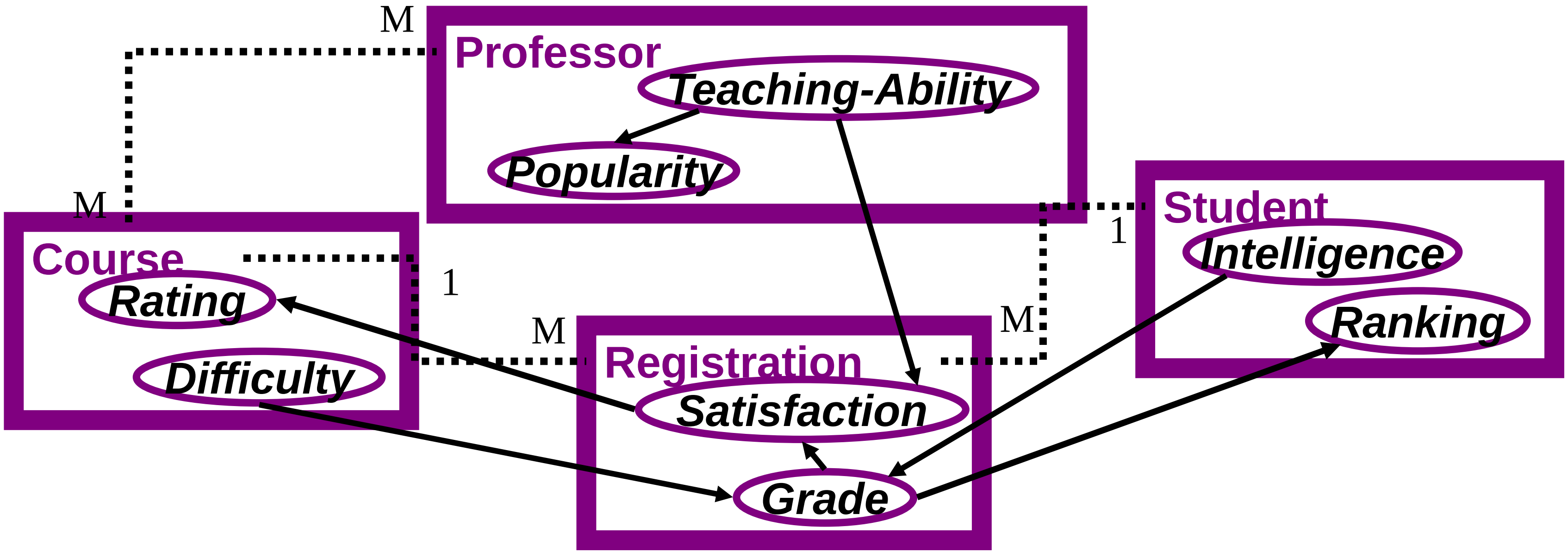
A . too many parents

B . variable # of parents

C. Too few parents

D. another problem

Problem with some parameters θ_S



When the slot chain τ (e.g. Course. Instructor) is not guaranteed to be single-valued, we must specify the probabilistic dependence of

- *x.A* Registration. Satisfaction
- on the set {*y.B*: *y* ∈ *x*.τ}

The Teaching-Ability of all the profs

who are instructors of the Course

How to specify cond. Prob. When # of parents can vary?

- **The notion of *aggregation* from database theory gives us the tool to address this issue; i.e., *x.a* will depend probabilistically on some aggregate property of this set**

Aggregation in PRMs

Examples of aggregation are:

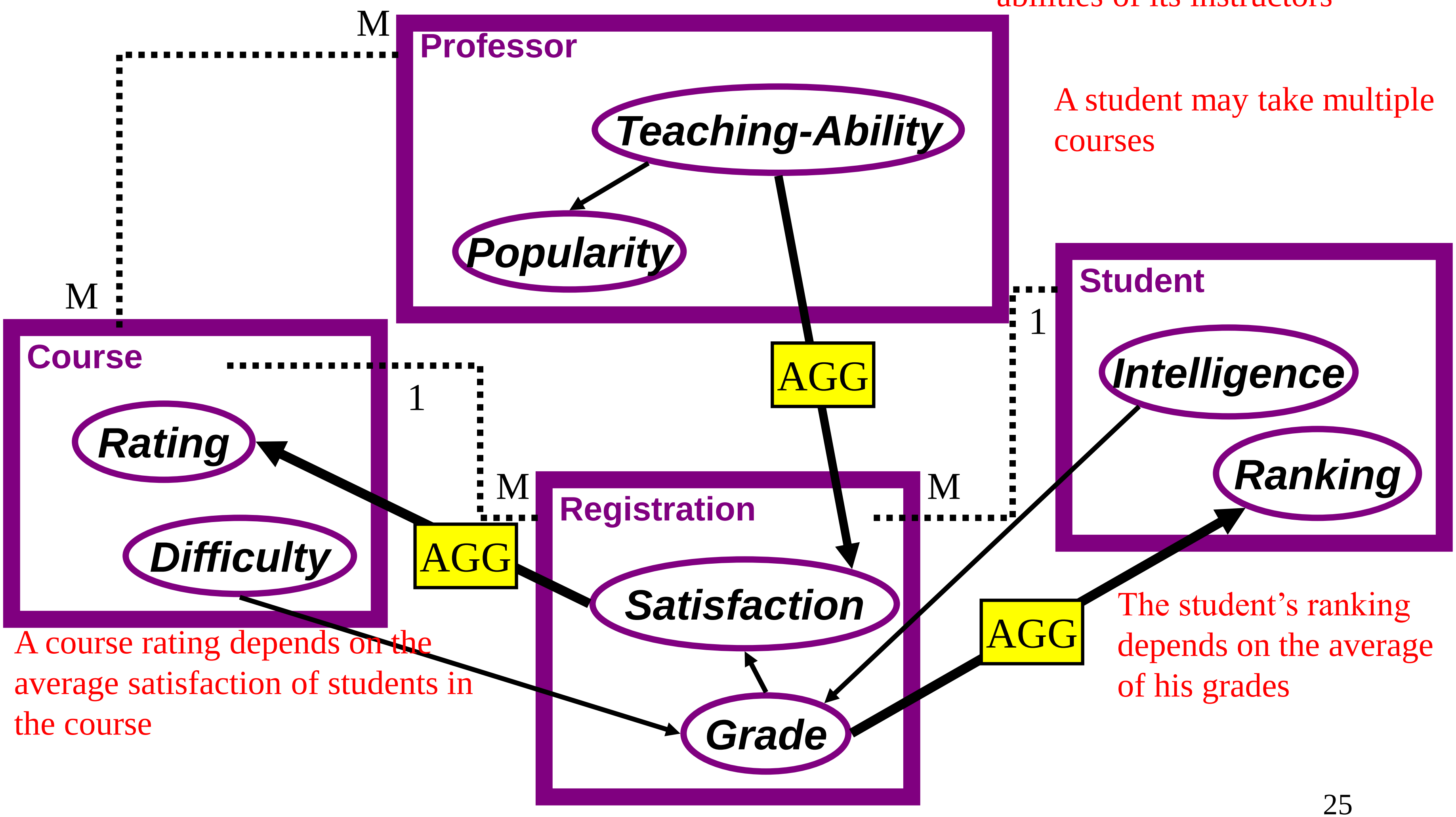
- the **mode** of the set (most frequently occurring value);
- **mean** value of the set (if values are numerical);
- **median**, **maximum**, or **minimum** (if values are ordered);
- **cardinality** of the set; etc.

PRM Dependency Structure with aggregations

The same course can be taught by multiple profs

A course satisfaction depends on the teaching abilities of its instructors

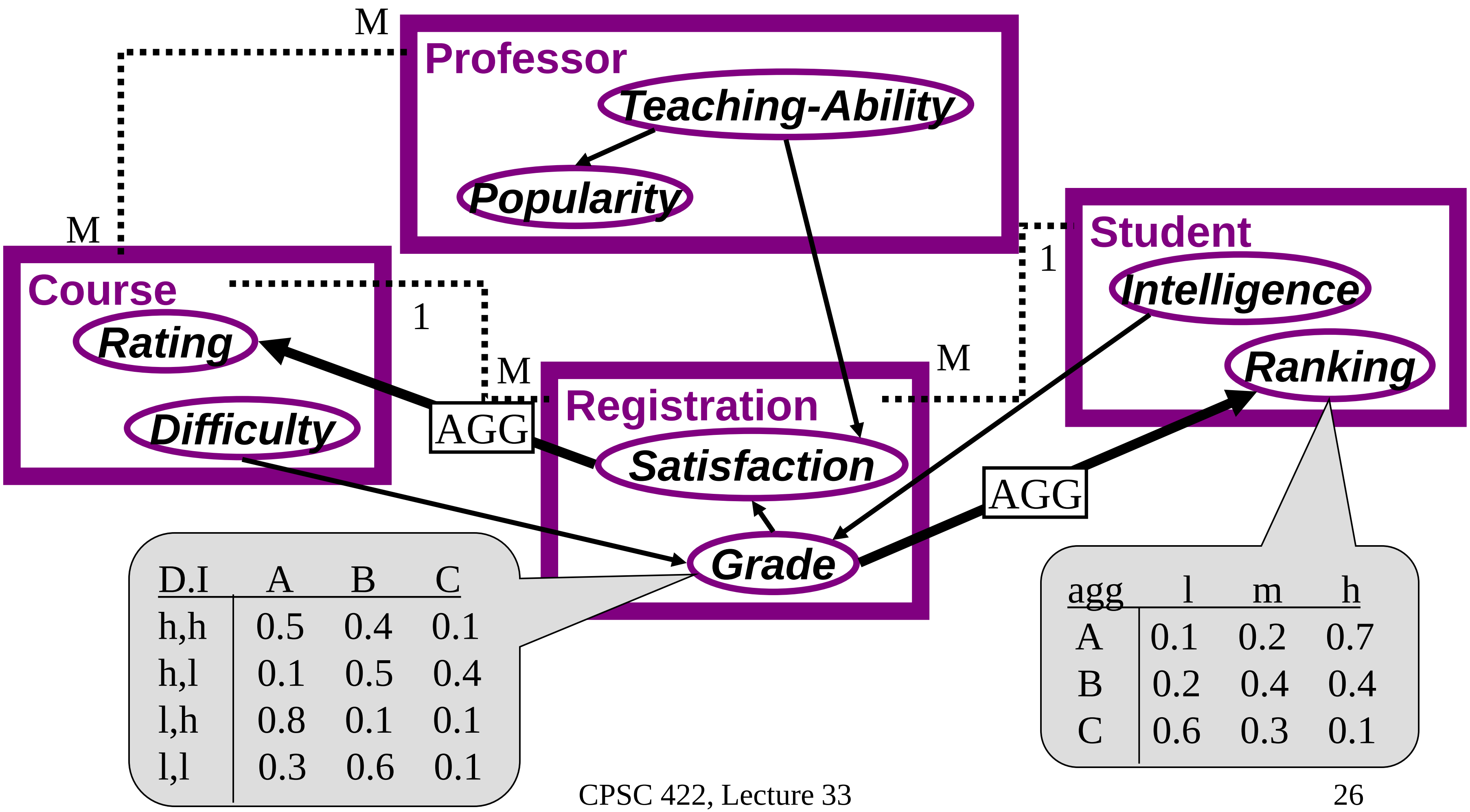
A student may take multiple courses



A course rating depends on the average satisfaction of students in the course

The student's ranking depends on the average of his grades

CPDs in PRMs



D.I	A	B	C
h,h	0.5	0.4	0.1
h,l	0.1	0.5	0.4
l,h	0.8	0.1	0.1
l,l	0.3	0.6	0.1

agg	l	m	h
A	0.1	0.2	0.7
B	0.2	0.4	0.4
C	0.6	0.3	0.1

JPD in PRMs

- Given a **skeleton structure $\boldsymbol{\sigma}$** for our schema, we can apply these **local conditional probabilities** to define a **JPD** (joint probability distribution) **over all completions** of the skeleton
- Note that the objects and relations between objects in a skeleton are always specified by σ, hence **we are disallowing uncertainty over the relational structure of the model**

Parameter Sharing / CPTs reuse, where else?

- **Temporal Models**
- **Because of the stationary assumption!**

Final Issue….

- To define a coherent probabilistic model as a Bayesian network, we **must ensure that our probabilistic dependencies are…..**

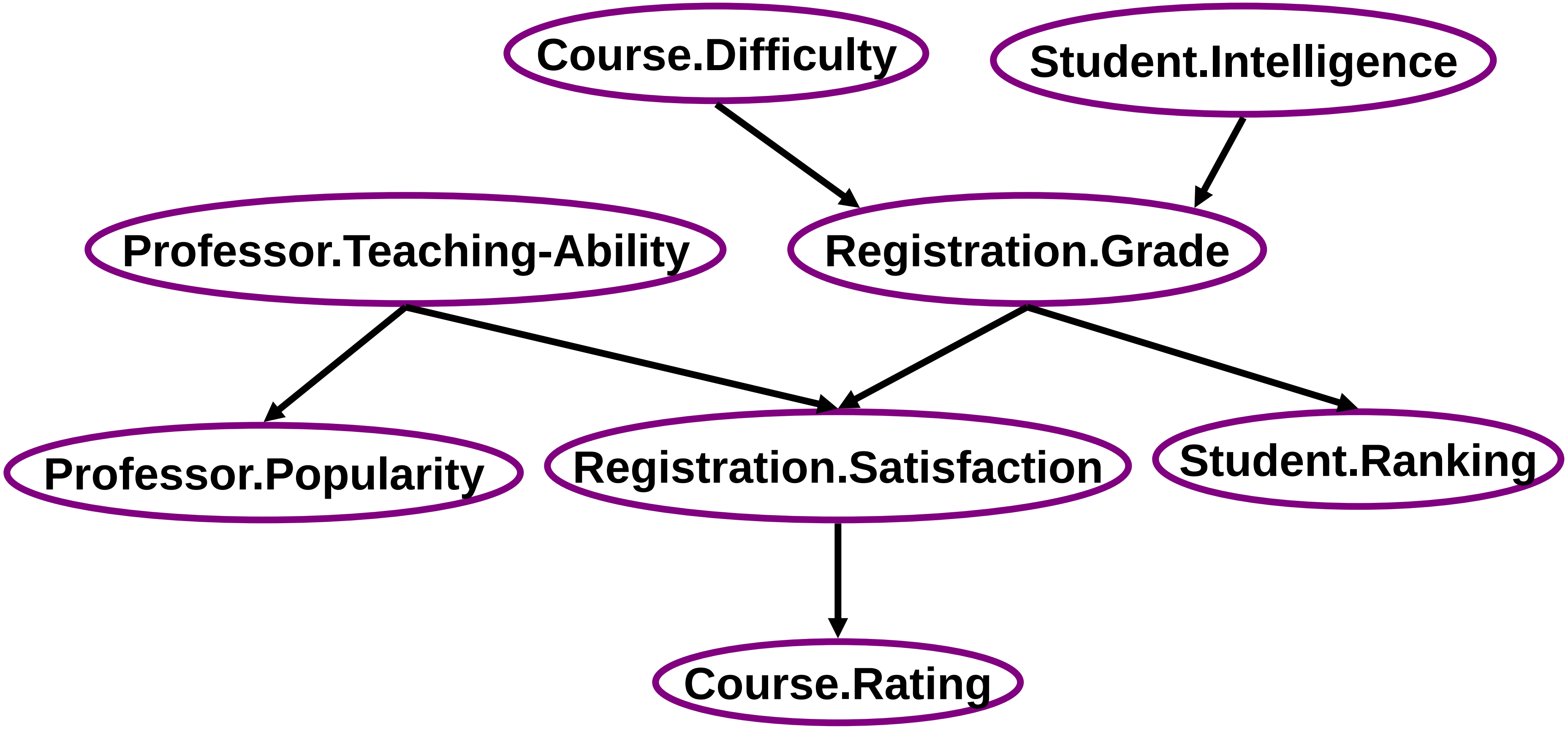
A. Polynomial

[✓ B. Acyclic]

C. Cyclic

D. Recursive

Class Dependency Graph for the University Domain



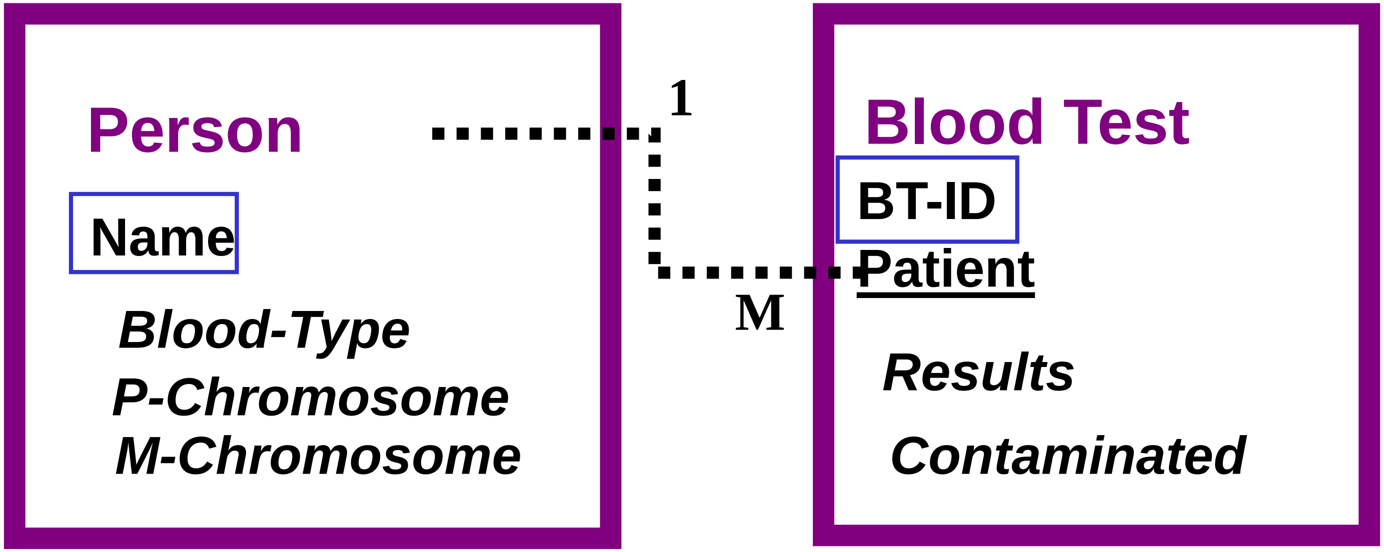
Ensuring Acyclic Dependencies

- In general, however, a cycle in the class dependency graph does not imply that all skeletons induce cyclic dependencies

- A model **may appear to be cyclic at the class level**, however, this cyclicity is always resolved at the level of individual objects

- The ability to guarantee that the cyclicity is resolved relies on **some prior knowledge about the domain**. The user can specify that certain slots are ***guaranteed acyclic***

Relational Schema for the Genetics Domain



Person

Name

Blood-Type

P-Chromosome

M-Chromosome

1

M

Blood Test

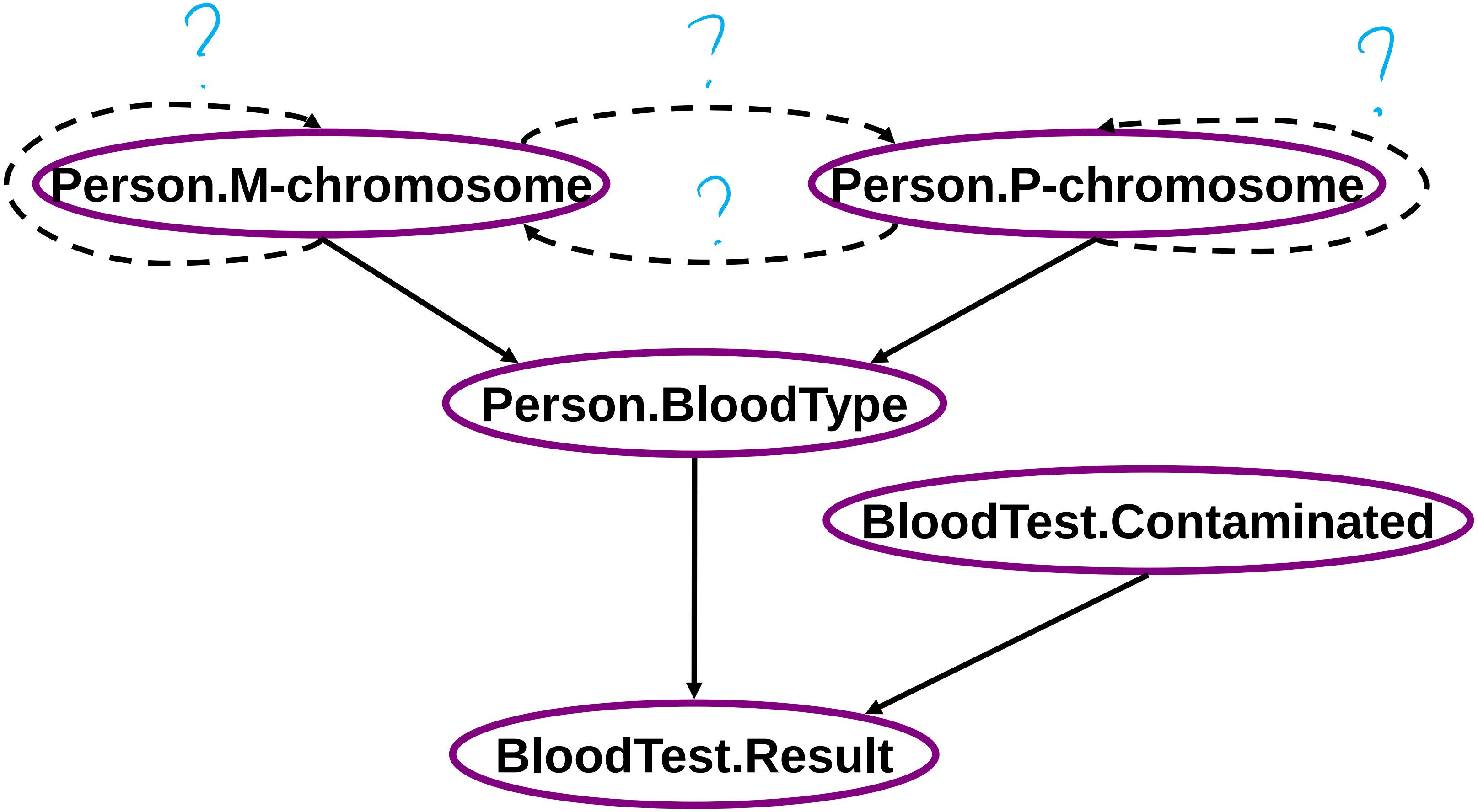
BT-ID

Patient

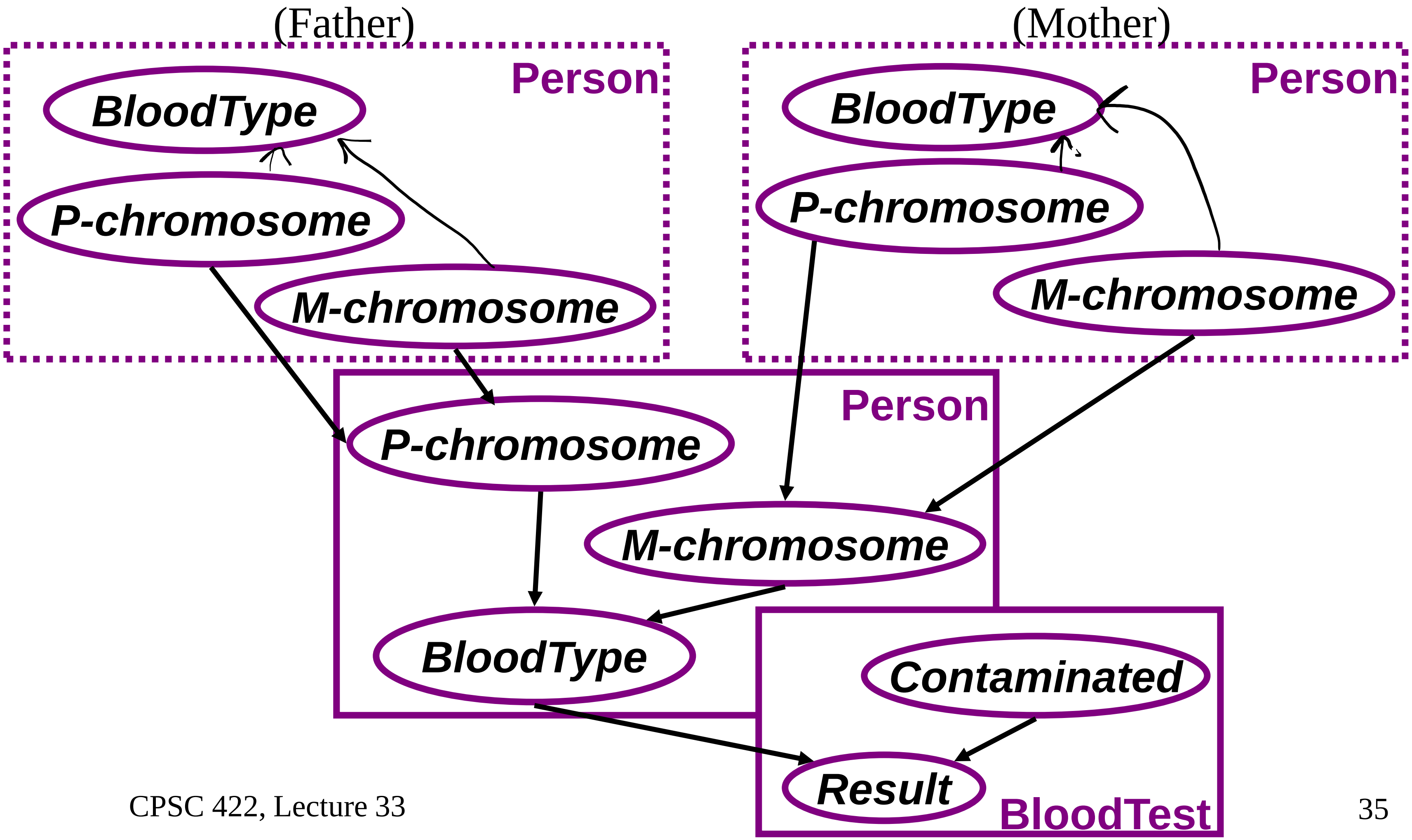
Results

Contaminated

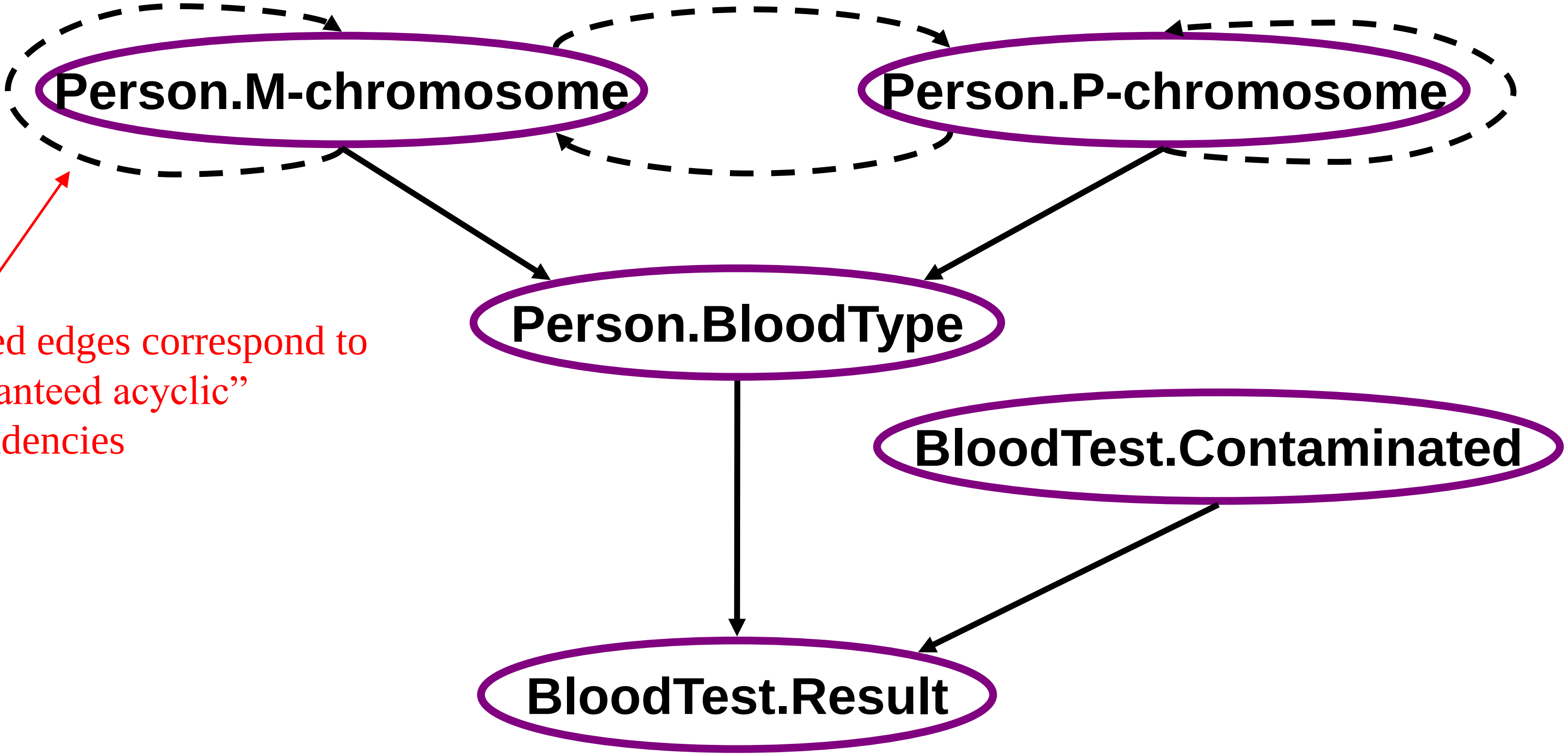
Dependency Graph for Genetics Domain



PRM for the Genetics Domain



Dependency Graph for Genetics Domain



Dashed edges correspond to “guaranteed acyclic” dependencies

Learning Goals for today's class

You can:

- **Build the grounded Bnet, given a Relational Skeleton, a dependency structure, and the corresponding parameters**
- **Define and apply guaranteed acyclicity**

422 big picture: Where are we?

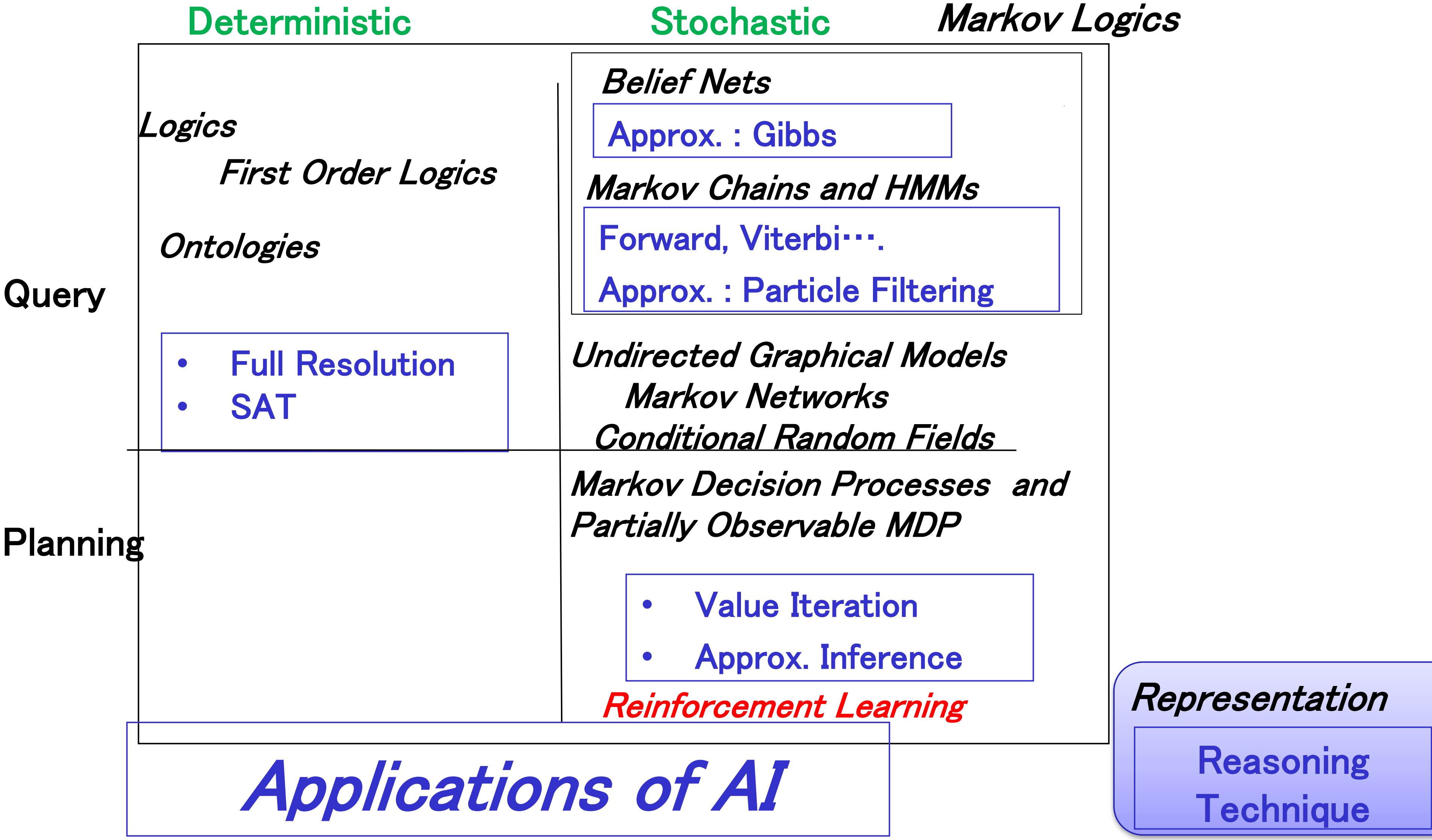
StarAI (statistical relational AI)

Hybrid: Det +Sto

Prob CFG

Prob Relational Models

Markov Logics



Deterministic

Stochastic

Query

Planning

Logics

First Order Logics

Ontologies

- Full Resolution
- SAT

Belief Nets

Approx. : Gibbs

Markov Chains and HMMs

Forward, Viterbi….

Approx. : Particle Filtering

Undirected Graphical Models

Markov Networks

Conditional Random Fields

Markov Decision Processes and Partially Observable MDP

- Value Iteration
- Approx. Inference

Reinforcement Learning

Applications of AI

Representation

Reasoning Technique

Last class on Fri

- Beyond 322/422 (ML + grad courses)
- Watson….
- Final Exam

Fill out on-line Teaching Evaluation