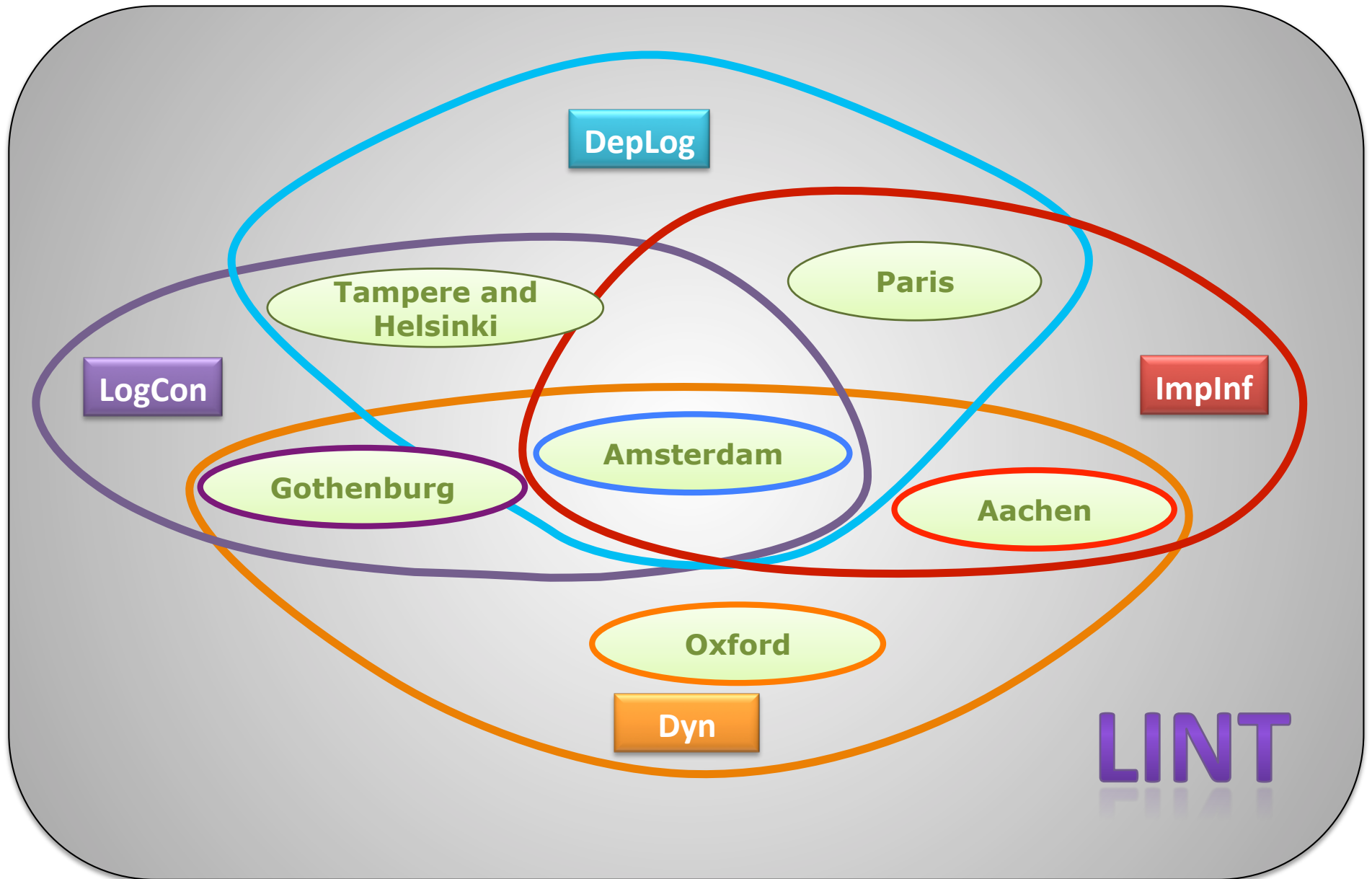


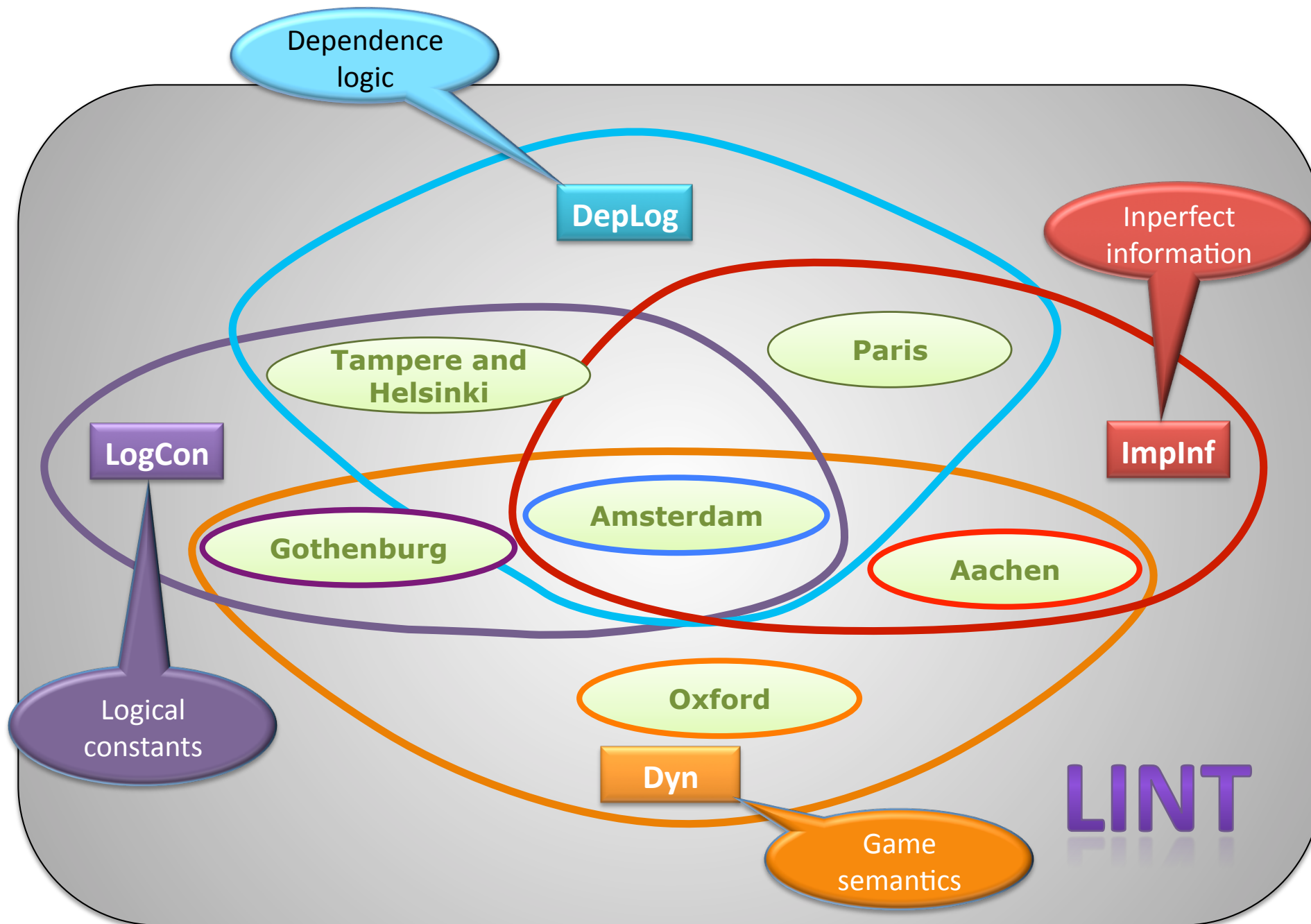


Dependence Logic

Overview of LINT

Jouko Väänänen
University of Amsterdam
University of Helsinki





Classical logic

there is	677
for all	399
for some	399
for every	146

Modal logic

possible	609
probably	313
likely	234
perhaps	201
it is possible that	146
possibly	118
necessarily	85
knows that	38
believes that	30
it is necessary that	23
it is obligatory that	0.1
it is permissible that	0.1

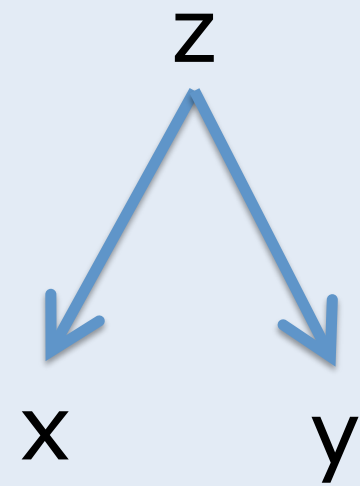
Dependence logic

is part of	720
includes	464
subject to	335
liable to	150
open to	140
dependent	88
determined by	78
given by	70
independence	64
equal to	60
function of	60
dependent on	43
dependence	32
independent of	30
belonging to	26
modified by	20
dependency	19
dependence on	13
vulnerable to	11
independence of	7
computed from	4
totally dependent on	3
uniquely determined by	3
contingent on	3
qualified by	2
totally independent of	1.6
conditioned by	1.5
left open by	1.3
mutually dependent	0.2
totally determined by	0.1
mutual dependency	0.06
mutually dependent on e/o	0.02

Dependencies with (an) other participant(s)	Two participants (legal entities) are dependent on each other where there is a controlling relationship between them: – A legal entity is under the same direct or indirect control as another legal entity (SG); - or – A legal entity directly or indirectly controls another legal entity (CLS); - or – A legal entity is directly or indirectly controlled by another legal entity (CLB). <u>Control:</u> Legal entity A controls legal entity B if: – A, directly or indirectly, holds more than 50% of the nominal value of the issued share capital or a majority of the voting rights of the shareholders or associates of B, - or – A, directly or indirectly, holds in fact or in law the decision-making powers in B. The following relationships between legal entities shall not in themselves be deemed to constitute controlling relationships: (a) the same public investment corporation, institutional investor or venture-capital company has a direct or indirect holding of more than 50 % of the nominal value of the issued share capital or a majority of voting rights of the shareholders or associates; (b) the legal entities concerned are owned or supervised by the same public body.
Character of dependence	According to the explanation above mentioned, please insert the appropriate abbreviation according to the list below to characterise the relation between your organisation and the other participant(s) you are related with: • SG: Same group: if your organisation and the other participant are <u>controlled by the same third party</u>; • CLS: Controls: if your organisation <u>controls the other participant</u>; • CLB: Controlled by: if your organisation is <u>controlled by the other participant</u>.

$x \rightarrow y$

$x \leftarrow y$



Question

Can one add the *dependence* concept to first order logic (or other logics) in a coherent way?

What is the *logic* of dependence?

Solution

- We consider the strongest form of dependence, namely functional determination $z = f(x_1, \dots, x_n)$, where $x_1, \dots, x_n, z$ are individual variables.
- We denote it $= (x_1, \dots, x_n, z)$ and call it a dependence atom. Weaker forms of dependence are derived from this.
- In computer science: $x_1 \dots x_n \Rightarrow z$, where $x_1, \dots, x_n, z$ are database fields. (Armstrong relation)

Solution

- We consider the strongest form of dependence, namely functional determination $z = f(x_1, \dots, x_n)$, where $x_1, \dots, x_n, z$ are individual variables.
- We denote it $=(x_1, \dots, x_n, z)$ and call it a **dependence atom**. Weaker forms of dependence are derived from this.
- In computer science: $x_1 \dots x_n \Rightarrow z$, where $x_1, \dots, x_n, z$ are database fields. (Armstrong relation)

Solution

- We consider the strongest form of dependence, namely functional determination $z = f(x_1, \dots, x_n)$, where $x_1, \dots, x_n, z$ are individual variables.
- We denote it $=(x_1, \dots, x_n, z)$ and call it a **dependence atom**. Weaker forms of dependence are derived from this.
- In computer science: $x_1 \dots x_n \Rightarrow z$, where $x_1, \dots, x_n, z$ are database fields. (Armstrong relation)

Weaker dependence

- Every day after work he is either in the library or in the bar.
- Functional dependence up to 2.

	Name	Job	Gender	Salary group
s_0	Jeff	analyst	M	C
s_1	Paula	assistant	F	A
s_2	Laurie	assistant	M	C

Multitude

- Dependence does not manifest itself in a **single** play, event or observation.
- The underlying concept of dependence logic is a multitude – a **collection** - of such plays, events or observations.
- These collections are called in this talk **teams**.
- They are the basic objects of our approach.

Multitude

- Dependence does not manifest itself in a **single** play, event or observation.
- The underlying concept of dependence logic is a multitude – a **collection** - of such plays, events or observations.
- These collections are called in this talk **teams**.
- They are the basic objects of our approach.

Multitude

- Dependence does not manifest itself in a **single** play, event or observation.
- The underlying concept of dependence logic is a multitude – a **collection** - of such plays, events or observations.
- These collections are called **teams**.
- They are the basic objects of our approach.

Multitude

- Dependence does not manifest itself in a **single** play, event or observation.
- The underlying concept of dependence logic is a multitude – a **collection** - of such plays, events or observations.
- These collections are called **teams**.
- They are the basic objects of our approach.

Teams

- A set of records of stock exchange transactions of a particular dealer.
- A set of possible histories of mankind written as decisions and consequences.
- A set of chess games between Susan and Max, as lists of moves.

Teams

- 1st intuition: A team is a set of plays of a game.

Teams

- 1st intuition: A team is a set of plays of a game.
- 2nd intuition: A team is a database.

	x_0	x_1	x_2
s_0	0	1	0
s_1	0	1	1
s_2	2	5	5

Towards a logic based on teams

- A set of plays satisfies $x_2 > x_0$ if move x_2 is in each play greater than move x_0 .
- A set of plays satisfies $=(x_1, \dots, x_n, y)$ if move y is in each play determined by the moves $x_1, \dots, x_n$.
- A database satisfies $x_2 > x_0$ if field x_2 is always greater than field x_0 .
- A database satisfies $=(x_1, \dots, x_n, y)$ if field y is functionally determined by the fields $x_1, \dots, x_n$.

Towards a logic based on teams

- A set of plays satisfies $x_2 > x_0$ if move x_2 is in each play greater than move x_0 .
- A set of plays satisfies $=(x_1, \dots, x_n, y)$ if move y is in each play determined by the moves $x_1, \dots, x_n$.
- A database satisfies $x_2 > x_0$ if field x_2 is always greater than field x_0 .
- A database satisfies $=(x_1, \dots, x_n, y)$ if field y is functionally determined by the fields $x_1, \dots, x_n$.

Dependence atoms $= (x_1, \dots, x_n, z)$

+

First order logic

=

Dependence logic

Syntax of dependence logic

$=, \neg, \vee, \wedge, \exists, \forall,), (, x_i$

$x_i, c, ft_1 \dots t_n$

$t=t'$

$=(x_1, \dots, x_n, z)$

$Rt_1 \dots t_n$

$t=t'$

$Rt_1 \dots t_n$

$\neg \varphi$

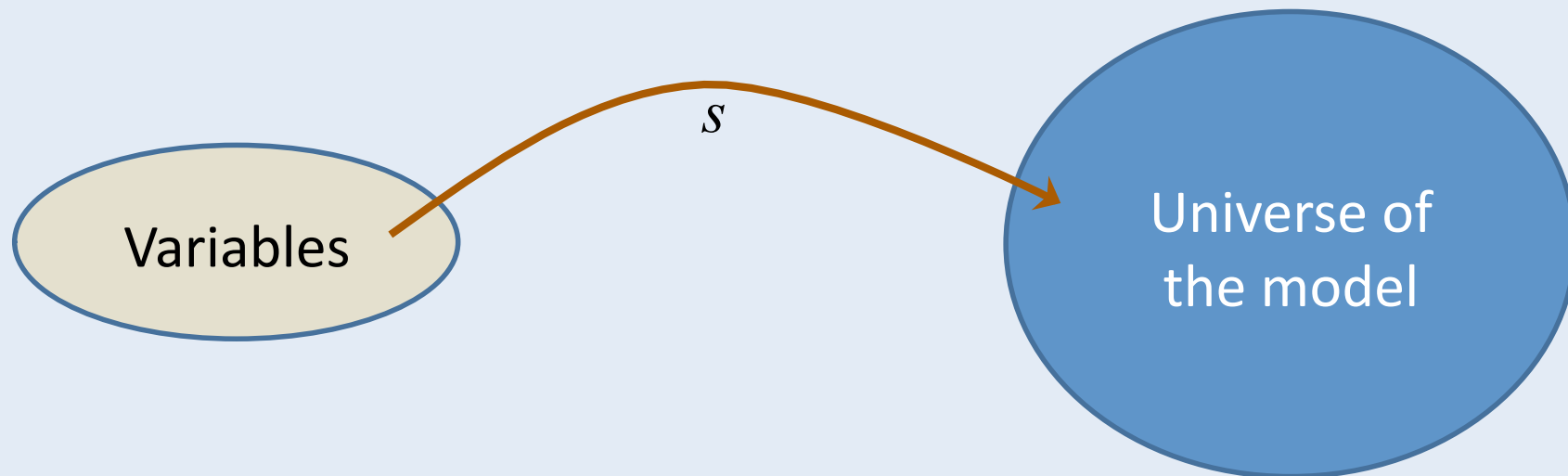
$\varphi \vee \psi$

$\varphi \wedge \psi$

$\exists x_i \varphi$

$\forall x_i \varphi$

Assignment



Teams – exact definition

- A **team** is just a **set** of assignments for a model.

Teams – exact definition

- A **team** is just a **set** of assignments for a model.
- **Empty team** $\emptyset$.
 - Database with no rows.
 - No play was played.

Teams – exact definition

- A **team** is just a **set** of assignments for a model.
- **Empty team $\emptyset$.**
 - Database with no rows.
 - No play was played.
- **The team $\{\emptyset\}$ with the empty assignment.**
 - Database with no columns, and hence with at most one row.
 - Zero moves of the game were played

For the truth definition: Negation Normal Form

We push negations all the way
to atomic formulas using de Morgan laws.
Thus $\neg\neg\varphi$ will have the same meaning as φ .

Truth definition

A team **satisfies a formula** if
every assignment in the team does,
and ...

A team satisfies $Rt_1...t_n$ if every team member does.

	x_0	x_1	x_2
s_0	0	1	0
s_1	0	1	1
s_2	2	5	5

$$x_0 < x_1$$

A team satisfies $\neg R t_1 \dots t_n$ if every team member does.

	x_0	x_1	x_2
s_0	0	1	0
s_1	0	1	1
s_2	2	5	5

$$\neg x_1 < x_0$$

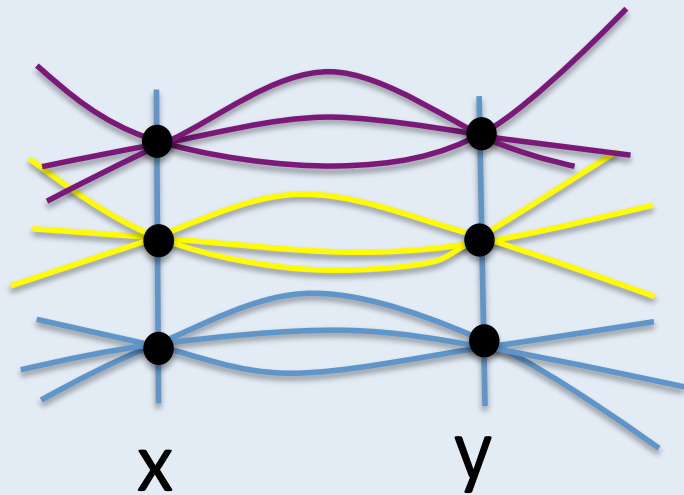
A team satisfies $\neg Rt_1 \dots t_n$ if every team member does.

	x_0	x_1	x_2
s_0	0	1	0
s_1	0	1	1
s_2	2	5	5

$$\neg x_1 < x_0$$

Note: some X satisfy neither $Rt_1 \dots t_n$ nor $\neg Rt_1 \dots t_n$.

A team satisfies $t=t'$ if every team member does.



$$x=y$$

	x_0	x_1	x_2
s_0	1	0	0
s_1	0	1	1
s_2	2	5	5

$$x_1=x_2$$

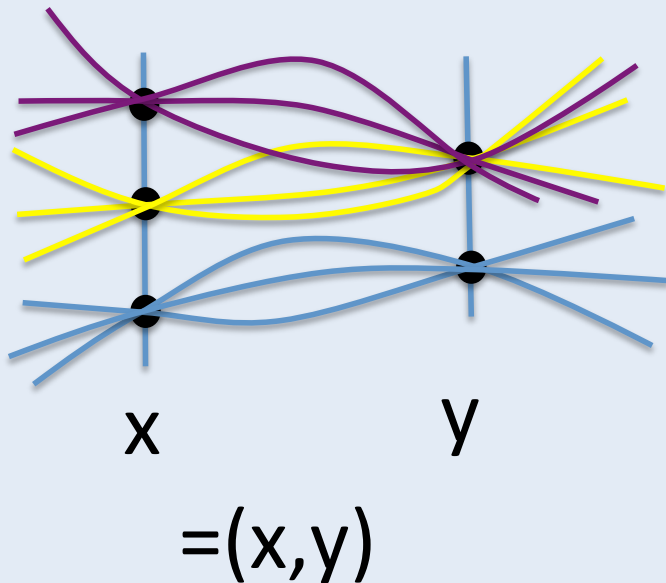
A team satisfies $\neg t = t'$ if every team member does.

	x_0	x_1	x_2
s_0	1	0	0
s_1	0	1	1
s_2	2	5	5

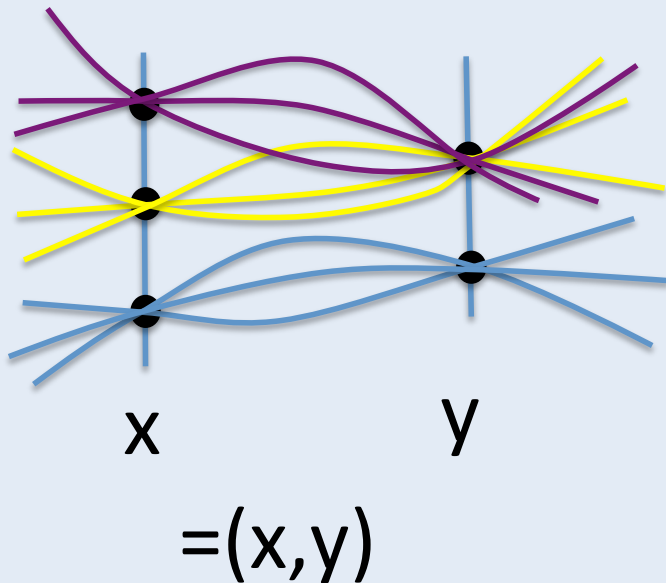
$$\neg x_0 = x_1$$

- A team X satisfies $\varphi(x_1, \dots, x_n, z)$ if in any two assignments in X , in which $x_1, \dots, x_n$ have the same values, also z has the same value.

- A team X satisfies $=(x_1, \dots, x_n, z)$ if in any two assignments in X , in which $x_1, \dots, x_n$ have the same values, also z has the same value.



- A team X satisfies $=(x_1, \dots, x_n, z)$ if in any two assignments in X , in which $x_1, \dots, x_n$ have the same values, also z has the same value.



	x	y	u	z
s_0	0	0	1	0
s_1	0	1	0	2
s_2	2	5	0	5
s_3	0	1	1	2

$=(x, y, z)$

An extreme case

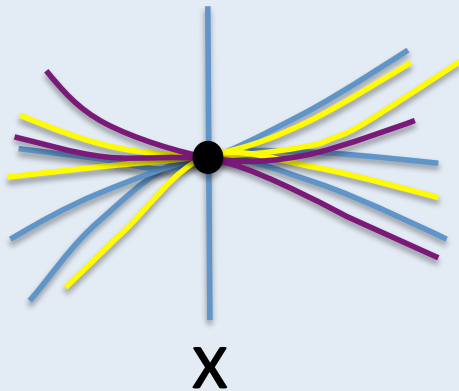
$=(x)$

"x is constant in the team"

An extreme case

$=(x)$

"x is constant in the team"



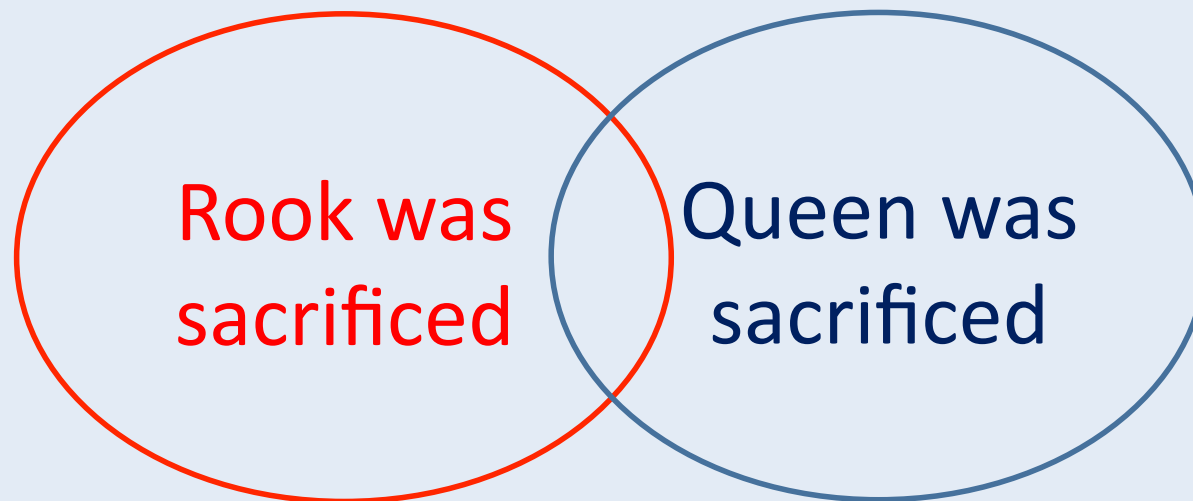
$=(x)$

record	A1	A2	A3	A4	A5	A6
100000	8	6	7	3	0	6
100002	7	5	6	3	0	6
100003	4	8	7	3	0	6
100004	6	5	4	3	0	6
100005	6	12	65	3	0	6
100006	5	56	9	3	0	6
100007	6	23	0	4	0	8
...	...	...	...	...	...	...
408261	77	2	11	1	0	2

- A team X satisfies $\varphi \vee \psi$ if
 $X = Y \cup Z$, where Y satisfies φ and Z
satisfies ψ .

- A team X satisfies $\varphi \vee \psi$ if
 $X = Y \cup Z$, where Y satisfies φ and Z satisfies ψ .

Plays where rook **or** queen was sacrificed:



Rank	Salary
A	2000
A	2100
B	2150
B	2220
C	2340
C	2440
D	2500
D	3100
E	3200
E	3710

=(Rank,Salary) ?

Rank	Salary
A	2000
A	2100
B	2150
B	2220
C	2340
C	2440
D	2500
D	3100
E	3200
E	3710

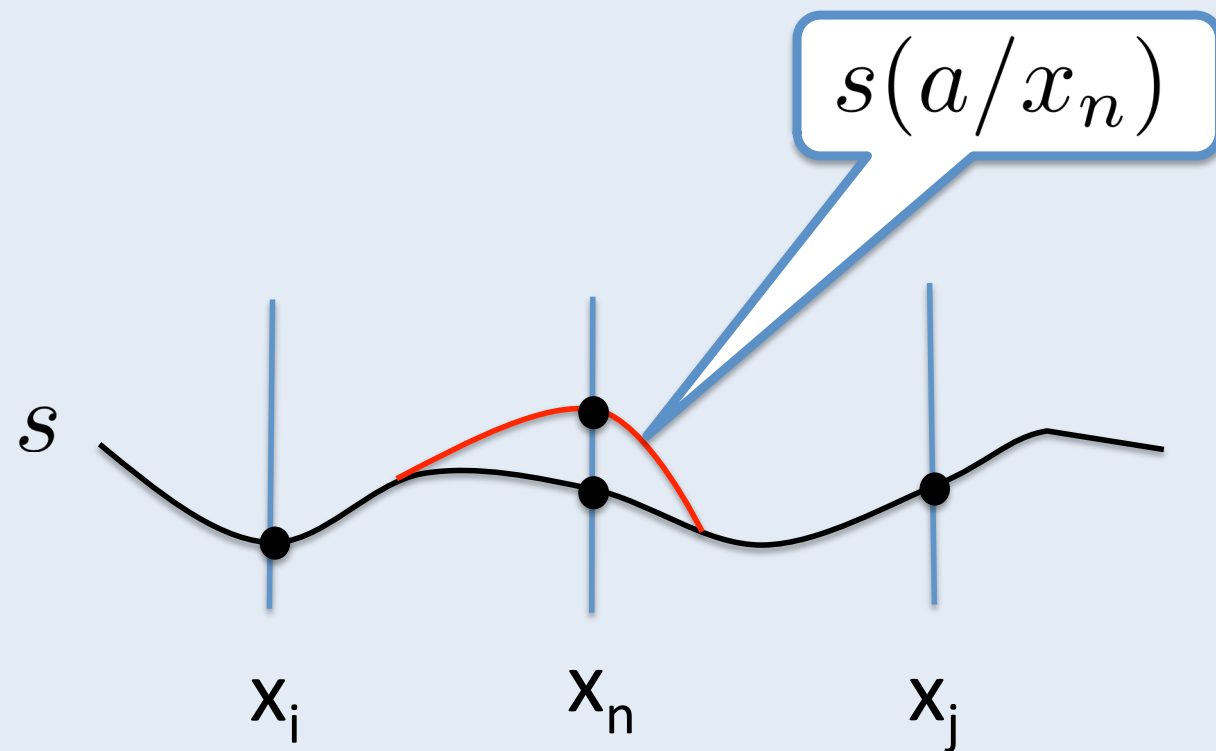
$=(\text{Rank}, \text{Salary}) \vee =(\text{Rank}, \text{Salary})$

Rank	Salary
A	2000
A	2100
B	2150
B	2220
C	2340
C	2440
D	2500
D	3100
E	3200
E	3710

$=(\text{Rank}, \text{Salary}) \vee =(\text{Rank}, \text{Salary})$

- A team X satisfies $\varphi \wedge \psi$ if it satisfies φ and ψ .

Quantifiers - modified assignment



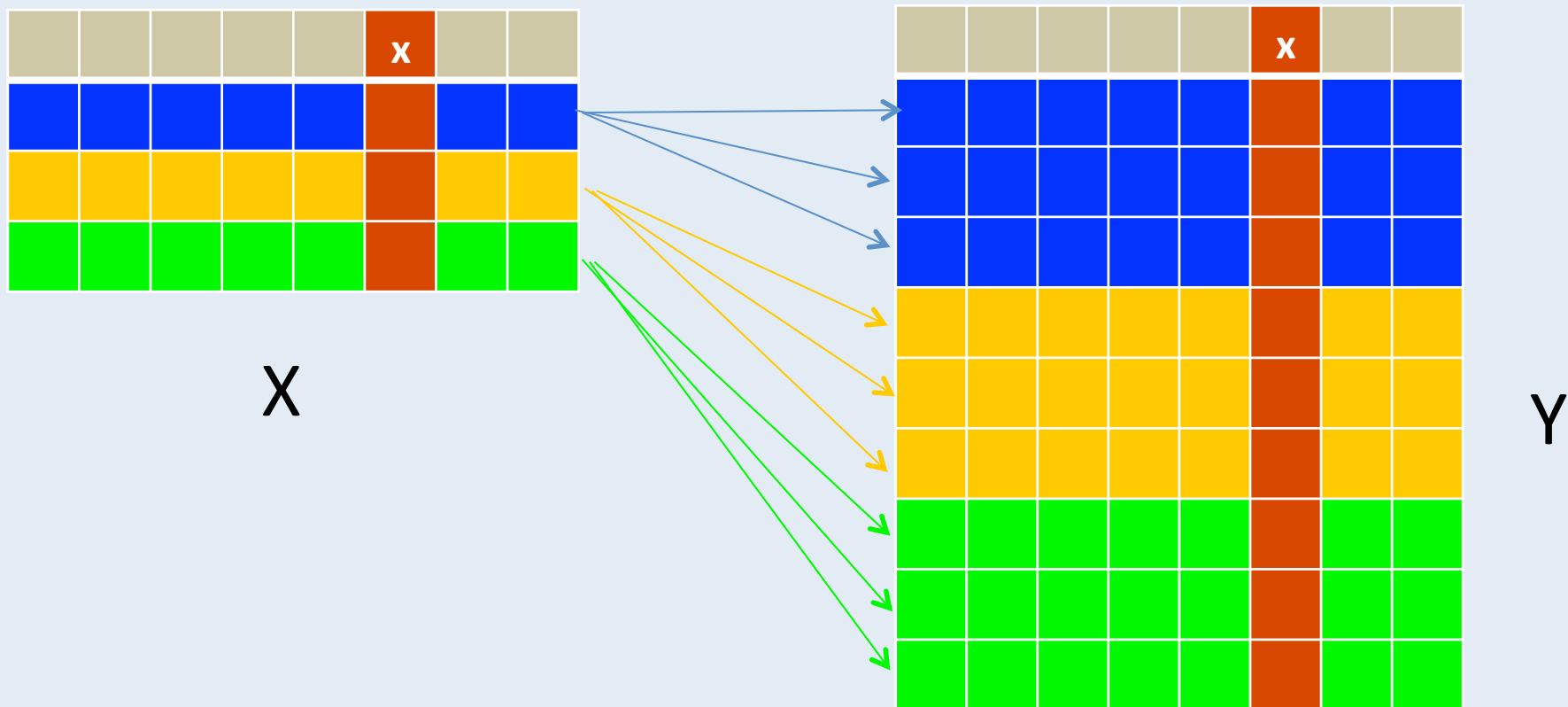
- A team X satisfies $\exists x\varphi$ if
there is a team Y such that Y satisfies φ
and for every s in X we have $s(a/x) \in Y$ for
some a .

Team X can be supplemented with values for x so that φ is satisfied.

					x		

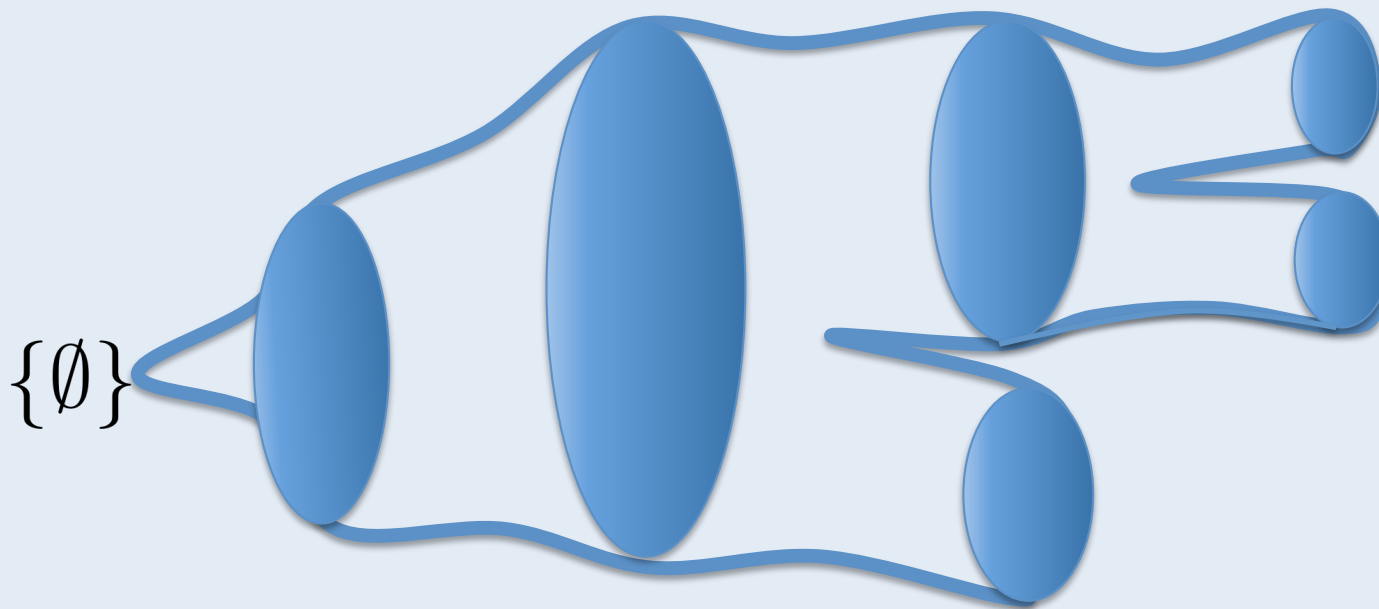
- A team X satisfies $\forall x\varphi$ if
there is a team Y such that Y satisfies φ
and for every s in X we have $s(a/x) \in Y$ for
all a .

Team X can be duplicated along x, by letting x get all possible values, and then φ is satisfied.



Truth

- A sentence is **true** if $\{\emptyset\}$ satisfies it.



Example: even cardinality



$$\begin{aligned} \forall x_0 \exists x_1 \forall x_2 \exists x_3 (& (x_2, x_3) \wedge \neg(x_0 = x_1) \\ & \wedge (x_0 = x_2 \rightarrow x_1 = x_3) \\ & \wedge (x_1 = x_2 \rightarrow x_3 = x_0)) \end{aligned}$$

Like Henkin (partially ordered) quantifiers.

Conservative over FO

A team $\{s\}$ satisfies a **first order formula** φ
iff
 s satisfies φ in the usual sense.

Two important properties

Downward closure: If a team satisfies a formula, every subset does. (Hodges: optimal!)

Empty set property: The empty team satisfies every formula.

Armstrong's Axioms

Always $=(x, x)$

If $=(x, y, z)$, then $=(y, x, z)$.

If $=(x, x, y)$, then $=(x, y)$.

If $=(x, z)$, then $=(x, y, z)$.

If $=(x, y)$ and $=(y, z)$, then $=(x, z)$.

Propositional rules

- From $\varphi \wedge \psi$ follows $\psi \wedge \varphi$.
 - From $\varphi \vee \psi$ follows $\psi \vee \varphi$.
 - From $\varphi \wedge (\psi \wedge \theta)$ follows $(\varphi \wedge \psi) \wedge \theta$.
 - From $\varphi \vee (\psi \vee \theta)$ follows $(\varphi \vee \psi) \vee \theta$.
 - From $(\varphi \vee \eta) \wedge (\psi \vee \theta)$ follows $(\varphi \wedge \psi) \vee (\varphi \wedge \theta) \vee (\eta \wedge \psi) \vee (\eta \wedge \theta)$.
 - From $(\varphi \wedge \eta) \vee (\psi \wedge \theta)$ follows $(\varphi \vee \psi) \wedge (\varphi \vee \theta) \wedge (\eta \vee \psi) \wedge (\eta \vee \theta)$.
 - From φ and ψ follows $\varphi \wedge \psi$.
 - From $\varphi \wedge \psi$ follows φ .
 - From φ follows $\varphi \vee \psi$.
- Commutative
- Associative
- "Almost" distributive

Incorrect rules

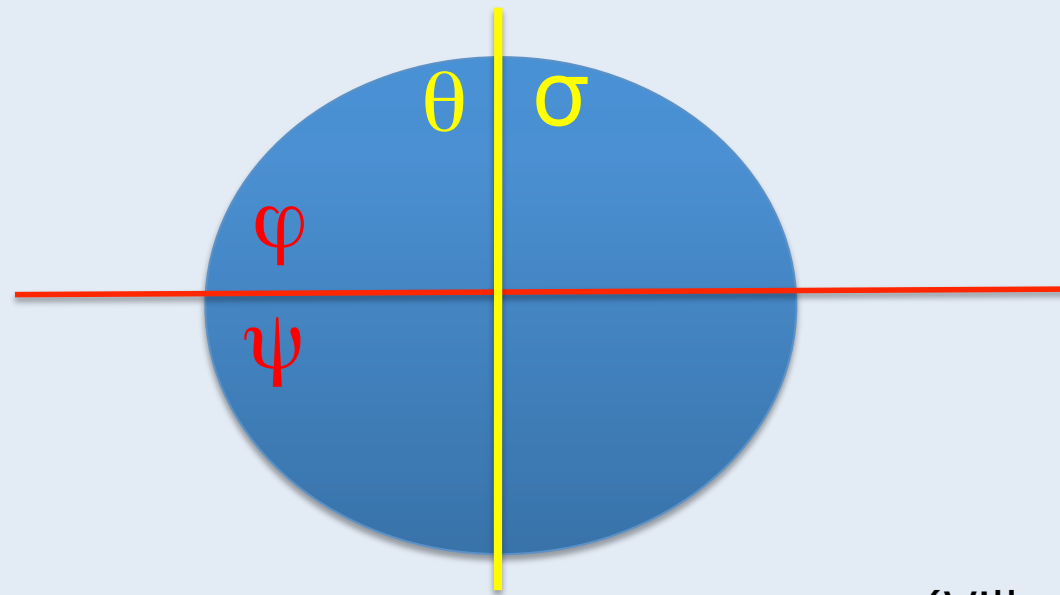
No absorption

- From $\varphi \vee \varphi$ follows φ . **Wrong!**
- From $(\varphi \wedge \psi) \vee (\varphi \wedge \theta)$ follows $\varphi \wedge (\psi \vee \theta)$. **Wrong!**
- From $(\varphi \vee \psi) \wedge (\varphi \vee \theta)$ follows $\varphi \vee (\psi \wedge \theta)$. **Wrong!**

Non-distributive

Surprising rule

- From $(\varphi \vee \psi) \wedge (\theta \vee \sigma)$ follows $\varphi \vee (\psi \wedge \theta) \vee (\psi \wedge \sigma)$.



(Ville Nurmi 2009)

Game theoretic semantics

- Dependence logic has two versions of the following games
 - Semantic (evaluation) game
 - Ehrenfeucht-Fraïssé game

Game theoretic semantics

- Dependence logic has two versions of the following games
 - Semantic (evaluation) game
 - Ehrenfeucht-Fraïssé game
- Version 1: Players move assignments.
 - Non-determined, imperfect information.

Game theoretic semantics

- Dependence logic has two versions of the following games
 - Semantic (evaluation) game
 - Ehrenfeucht-Fraisse game
- Version 1: Players move assignments.
 - Non-determined, imperfect information.
- Version 2: Players move teams.
 - Determined, perfect information.

Model theory of dependence logic

Hodges 1997: For every formula $\varphi(x_1, \dots, x_n)$ there is an **existential second order** sentence $\Phi(P)$ with P **negative** such that a team X satisfies φ iff $\Phi(X)$ is true.

Model theory of dependence logic

Hodges 1997: For every formula $\varphi(x_1, \dots, x_n)$ there is an **existential second order** sentence $\Phi(P)$ with P **negative** such that a team X satisfies φ iff $\Phi(X)$ is true.

Theorem (Kontinen-V. 2008): The converse is also true.

Answers a question of Hodges.

Consequences

- A language for NP on finite models.
- Compactness.
- Löwenheim-Skolem.
- Separation (Interpolation).

Classical negation

- The closure of dependence logic under classical negation has the exact strength of second order logic (Ville Nurmi, 2008).
- But we need negation to express Arrow's Theorem?

How about intuitionistic negation?

Joint work with S. Abramsky.

- **Definition:** X satisfies $\varphi \rightarrow \psi$ iff every subteam of X which satisfies φ also satisfies ψ .
- **Definition:** X satisfies $\perp$ iff X is the empty team.
- $\neg \varphi$ is now equivalent to $\varphi \rightarrow \perp$ for atomic φ .
- Intuitionistic negation ($\varphi \rightarrow \perp$) is an alternative way to extend negation from atomic to non-atomic formulas.

How about intuitionistic negation?

Joint work with S. Abramsky.

- **Definition:** X satisfies $\varphi \rightarrow \psi$ iff every subteam of X which satisfies φ also satisfies ψ .
- **Definition:** X satisfies $\perp$ iff X is the empty team.
- $\neg \varphi$ is now equivalent to $\varphi \rightarrow \perp$ for **atomic** φ .
- Intuitionistic negation ($\varphi \rightarrow \perp$) is an alternative way to extend negation from atomic to non-atomic formulas.

How about intuitionistic negation?

Joint work with S. Abramsky.

- **Definition:** X satisfies $\varphi \rightarrow \psi$ iff every subteam of X which satisfies φ also satisfies ψ .
- **Definition:** X satisfies $\perp$ iff X is the empty team.
- $\neg \varphi$ is now equivalent to $\varphi \rightarrow \perp$ for **atomic** φ .
- Intuitionistic negation ($\varphi \rightarrow \perp$) is an **alternative** way to extend negation from atomic to non-atomic formulas.

How about intuitionistic negation?

- Dependence atoms can now be defined in terms of constancy:

$$=(x_1, \dots, x_n, z) \equiv (=(x_1) \wedge \dots \wedge =(x_n)) \rightarrow =(z).$$

- Downward closure and the empty set property are preserved.
- Compactness fails.
- Goes beyond NP, unless NP=co-NP.

How about intuitionistic negation?

- Dependence atoms can now be defined in terms of constancy:

$$=(x_1, \dots, x_n, z) \equiv (=(x_1) \wedge \dots \wedge =(x_n)) \rightarrow =(z)$$

- Downward closure and the empty set property are preserved.
- Compactness fails.
- Goes beyond NP, unless NP=co-NP

How about intuitionistic negation?

- Dependence atoms can now be defined in terms of constancy:

$$=(x_1, \dots, x_n, z) \equiv (=(x_1) \wedge \dots \wedge =(x_n)) \rightarrow =(z)$$

- Downward closure and the empty set property are preserved.
- Compactness fails.
- Goes beyond NP, unless NP=co-NP.

How about intuitionistic negation?

- Dependence atoms can now be defined in terms of constancy:

$$=(x_1, \dots, x_n, z) \equiv (=(x_1) \wedge \dots \wedge =(x_n)) \rightarrow =(z)$$

- Downward closure and the empty set property are preserved.
- Compactness fails.
- Goes beyond NP, unless NP=co-NP.

We can **prove** Armstrong's Axioms

Dependence logic	Heyting's intuitionistic logic
$=(\mathbf{x}, \mathbf{x})$	$=(\mathbf{x}) \rightarrow =(\mathbf{x})$
If $=(\mathbf{x}, \mathbf{y}, \mathbf{z})$, then $=(\mathbf{y}, \mathbf{x}, \mathbf{z})$.	If $(=(\mathbf{x}) \wedge =(\mathbf{y})) \rightarrow =(\mathbf{z})$, then $(=(\mathbf{y}) \wedge =(\mathbf{x})) \rightarrow =(\mathbf{z})$
If $=(\mathbf{x}, \mathbf{x}, \mathbf{y})$, then $=(\mathbf{x}, \mathbf{y})$.	If $(=(\mathbf{x}) \wedge =(\mathbf{x})) \rightarrow =(\mathbf{y})$, then $=(\mathbf{x}) \rightarrow =(\mathbf{y})$
If $=(\mathbf{x}, \mathbf{z})$, then $=(\mathbf{x}, \mathbf{y}, \mathbf{z})$.	If $=(\mathbf{x}) \rightarrow =(\mathbf{z})$, then $(=(\mathbf{x}) \wedge =(\mathbf{y})) \rightarrow =(\mathbf{z})$
If $=(\mathbf{x}, \mathbf{y})$ and $=(\mathbf{y}, \mathbf{z})$, then $=(\mathbf{x}, \mathbf{z})$.	If $=(\mathbf{x}) \rightarrow =(\mathbf{y})$, and $=(\mathbf{y}) \rightarrow =(\mathbf{z})$ then $=(\mathbf{x}) \rightarrow =(\mathbf{z})$

Galois connection

- Intuitionistic implication is the adjoint of conjunction:

$$(\phi \wedge \psi) \models \theta \iff \phi \models \psi \rightarrow \theta$$

The moral of the story

- One can add intuitionistic implication to dependence logic without losing the downward closure.
- Intuitionistic negation agrees with the original negation on the atomic level, and basic axioms of dependence become provable.
- Good (?) for proof theory, but bad (?) for model theory. Is there a reason for this?

What is dependence logic good for?

- A logic for a variety of dependence concepts.
- Language for NP.
- A vehicle for uncovering the mathematics of dependence in a variety of contexts
 - Data mining
 - Social choice theory
 - Logic for Interaction

- J. Väänänen, *Dependence Logic*, Cambridge University Press, 2007.

Thank you!