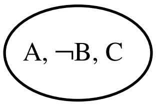
Interpretationen in Aussagenlogik/Prädikatenlogik/Modallogik

In der AL fragt man:
Was gilt in **einer** (ganzen) Welt oder **Situation**?

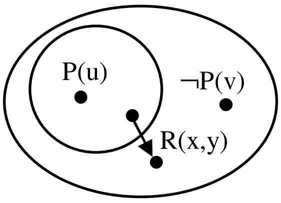
In der PL1 fragt man:
Was gilt für **welche Objekte** in **einer** Welt oder Situation?

In der AL-ML fragt man:
Was gilt für **welche** (ganzen) Welten oder **Situationen** – auch in Bezug auf **Übergänge** dazwischen?

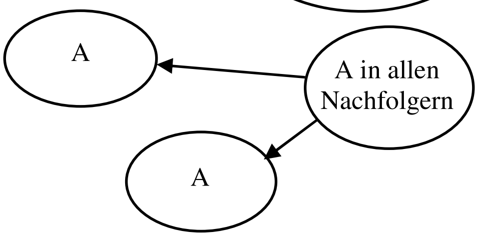
AL:



PL1:



AL-ML:



Modale Logik/Modallogik

Klassische Modallogiken über der Aussagenlogik verwenden neben den Sprachmitteln der AL zwei zusätzliche einstellige Operatoren, $\Box$ (Box) und $\Diamond$ (Karo, Diamond). Es gibt auch Modallogik(en) über PL1; diese behandeln wir hier nicht.

Durch unterschiedliche praktische Interpretationen und daraus resultierend unterschiedliche Axiome und evtl. unterschiedliche Ableitungsregeln ergeben sich **verschiedene Modallogiken**.

Klassische Modallogik(en) sprechen von ...

$\Box$ – Es ist **notwendig**, dass ...

$\Diamond$ – Es ist **möglich**, dass ...

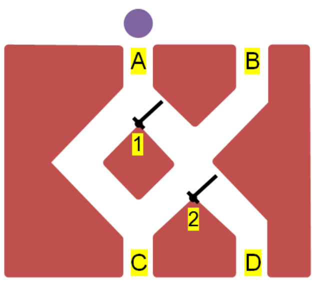
Klammerung/Nichtklammerung erfolgt wie bei ¬. So ergibt sich induktiv die Formelmenge ***MLForm*** der ML-Formeln.

Dazu werden generell noch 2 Äquivalenzen (der **Dualität** zwischen $\Box$ und $\Diamond$) gefordert:

$$\Diamond A \equiv \neg\Box\neg A$$

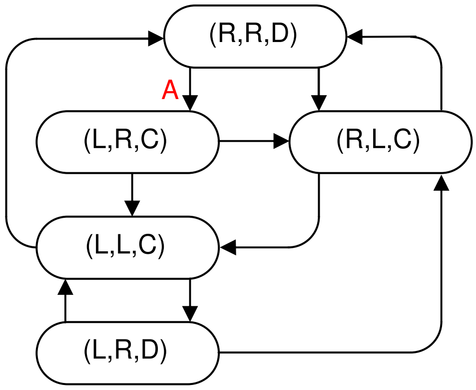
$$\Box A \equiv \neg\Diamond\neg A$$

Je nach (philosophischem oder mathematischem) **Notwendigkeitsbegriff** kommen unterschiedliche weitere Axiome hinzu, und so ergeben sich unterschiedliche klassische modale „Logik<u>en</u>".



Betrachten wir z.B. ein mechanisches Brettspiel, bei dem jeweils eine Kugel durch Gänge mit Klappen rollt. Über kleine Seitenflügel legt eine passierende Kugel die Klappe jeweils um.

Für die Anfangssituation, d.h. 1. und 2. Klappe rechts, letzte Kugel war durch D herausgekommen (wieso?) gilt: Die in A eingeworfene Kugel verändert die Situation: Sie legt Klappe 1 um – von R nach L – und kommt durch C heraus, schematisch siehe Pfeil mit „A" im Diagramm. Auch die anderen möglichen Übergänge sind eingezeichnet, wobei auf „A" und „B" verzichtet wird.



Hier bedeutet ...

- notwendig: nach der nächsten Kugel muss …
- möglich: nach der nächsten Kugel kann …

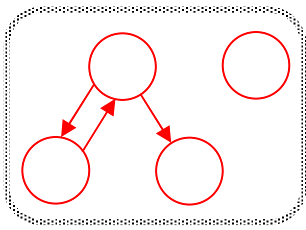
Einige Fragen lassen sich bereits anhand des Diagramms beantworten:

- „Wann" muss die nächste Kugel notwendig nach C? – z.B. RRD $\models \Box$C.
- „Wann" ist es möglich, eine Kugel so einzuwerfen, dass die folgende Kugel notwendig nach D muss? – z.B. RLC $\models \Diamond\Box$D.
- Können in der Ausgangssituation zwei Kugeln so eingeworfen werden, dass die folgende Kugel notwendig nach D muss? – Ja, RRD $\models \Diamond\Diamond\Box$D.

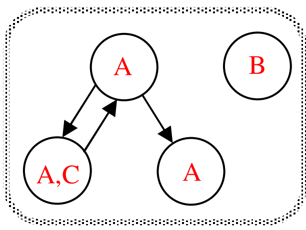
Formale Semantik modaler Logiken

Eine **Kripke-Interpretation** $M = (Rah, Bel, Anf)$ (über einer Aussagevariablenmenge AV): besteht aus:

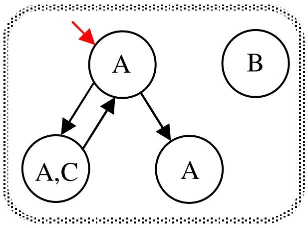
a) Kripke-**Rahmen** $Rah = (K, R)$ – ein (gerichteter) **Graph** mit der Knotenmenge $K \neq \varnothing$ (**„Welten"** oder **„Situationen"**) und der **Übergangsrelation** $R \subseteq K \times K$, hier abgebildet als Kanten des Graphen:



b) **Belegung** $Bel : K \to \mathbf{P}(AV)$ – **Knoteninschrift-**Abbildung, die jedem Knoten die die mit W belegten Aussagevariablen zuordnet. Rahmen und Belegung nennt man eine **Kripke-Struktur** (leider auch **„Kripke-Modell"**).



c) meist auch **Anfangssituation**, ein ausgezeichneter **Knoten** $Anf \in K$.



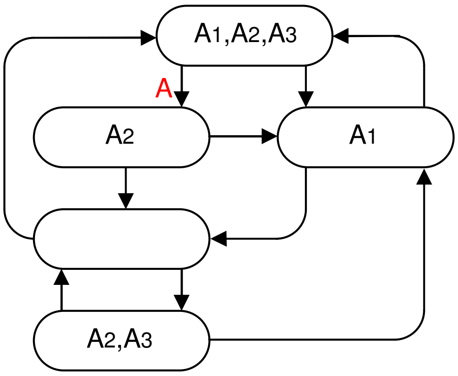
Jetzt können wir auch das Brettspiel als Kripke-Interpretation darstellen, indem wir z.B. wie folgt bezeichnen: Aussagevariablen

A_1: Klappe 1 rechts

A_2: Klappe 2 rechts

A_3: letzte Kugel kam aus D

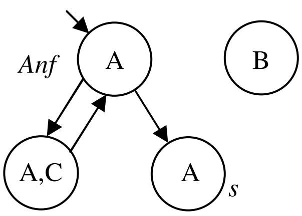
L,R,D beispielsweise entspricht dann A_2, A_3.



Mit einer Kripke-Interpretation erhalten alle ML-Formeln einen **Wahrheitswert**, nämlich so:

- Von allen **Aussagevariablen** sollen „in der Situation s" genau die dem Knoten s zugeordneten A_i aus $Bel(s)$ wahr sein.
- **Ohne Modaloperatoren** „geht es nur um die Anfangssituation" *Anf.*
- Jetzt □ (notwendig) und ◇ (möglich) zusätzlich zur AL:
  Notwendig sind die Formeln, die in allen von *Anf* aus unmittelbar über 1 Kante erreichbaren Situationen wahr sind:
  Möglich sind die Formeln, die in mindestens einer von *Anf* aus unmittelbar über 1 Kante erreichbaren Situation wahr sind.

Im folgenden Beispiel erkennen wir bereits ohne formale Definition und Methode …



Intuitiv wahr sind in $M = (Rah, Bel, Anf)$:

$A,\ \neg B,\ \neg C,\ \Box A,\ \Diamond C$

Weniger einfach sind vielleicht Fragen wie:

Gilt $(\Box(A \wedge C) \to \Diamond\Diamond\neg C) \to B$?

Für das Folgende eine Def.: **Übergang** zu einer anderen Anfangssituation:

Ist $M = (Rah, Bel, Anf)$ mit $Rah = (K, R)$ eine Kripke-Interpretation und $s \in K$ eine Situation (ein Knoten) des Rahmens, so ist $M^s := (Rah, Bel, s)$. Natürlich ist damit $M^{Anf} = M$ und $(M^{s1})^{s2} = M^{s2}$. Wir können so ähnlich auch einer Kripke-Struktur (Rah, Bel) eine Kripke-Interpretation $M^s := (Rah, Bel, s)$ zuordnen.

Im Beispiel wird bei M^s der Knoten rechts unten neuer Anfangsknoten.

Die **Kripke-Semantik** weist einer ML-Formel unter einer Kripke-Interpretation einen Wahrheitswert zu:

Sei $M = (Rah, Bel, Anf)$.

$wert_M(\varphi)$ ist wie folgt rekursiv definiert:

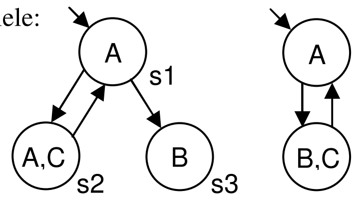
$\varphi = A_i \in AV \quad \Rightarrow$
$\quad wert_M(A_i) :=$ if $A_i \in Bel(Anf)$ then W else F

$\varphi, \psi \in MLForm \quad \Rightarrow$
$\quad wert_M(\neg\varphi) :=$ if $wert_M(\varphi) =$ W then F else W
$\quad wert_M(\varphi \wedge \psi) :=$ if
$\quad\quad wert_M(\varphi) = wert_M(\psi) =$ W then W else F
$\quad wert_M(\Box\varphi) :=$ if
$\quad\quad \forall s \in K : (Anf\ R\, s \Rightarrow wert_{M^s}(\varphi) = \mathrm{W})$ then W else F*
$\quad wert_M(\Diamond\varphi) :=$ if
$\quad\quad \exists s \in K : (Anf\ R\, s \wedge wert_{M^s}(\varphi) = \mathrm{W})$ then W else F*

*) ∀: für alle, ∃: für mindestens ein(e)

$M \vDash \varphi$:⇔ φ gilt in M
:⇔ M erfüllt/ ist Modell für φ
:⇔ $wert_M(\varphi)$ =W

Auswertungsbeispiele:



□◇A	F	W
□(B→¬◇A)	W	F
□(A→◇C)	F	W

Systematische Auswertung: z.B. analog AL-**Wahrheitstafel** bottom up – aber für alle Situationen an Stelle von allen Belegungsmöglichkeiten. **Beispiel (links)**:

Formel	s1 >s2,s3	s2 >s1	s3
A	W	W	F
B	F	F	W
◇A	W	W	F
¬◇A	F	F	W
B→¬◇A	W	W	W
□(B→¬◇A)	**W**	W	W

Manchmal interessiert, ob eine Formel in einer Kripke-Struktur (Rah, Bel) in **jeder möglichen Situation** wahr ist:

$(Rah, Bel) \vDash \varphi$

$:\Leftrightarrow$ (Rah, Bel) erfüllt/ ist Modell für φ,
φ gilt in (Rah, Bel)
$:\Leftrightarrow$ $\forall s \in K : (Rah, Bel, s) \vDash \varphi$

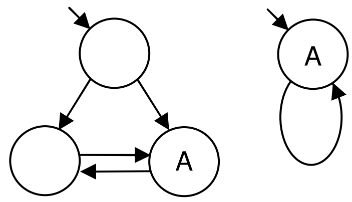
Manchmal interessiert, ob eine Formel in einem Rahmen **sogar bei jeder möglichen Belegung** in jeder möglichen Situation wahr ist:

$Rah \vDash \varphi$
$:\Leftrightarrow$ Rah erfüllt / ist Modell für φ, φ gilt in Rah
$:\Leftrightarrow$ $\forall$ Abb. $Bel : K \to P(AV) : (Rah, Bel) \vDash \varphi$
$:\Leftrightarrow$ $\forall$ Abb. $Bel : K \to P(AV) : \forall s \in K : (Rah, Bel, s) \vDash \varphi$

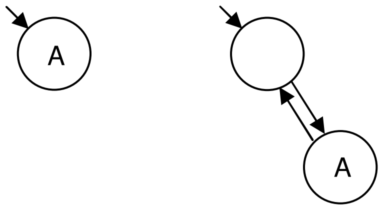
Dann geht es nur noch um Eigenschaften der Relation/ des Graphen.

Einige **Modelle und Gegenbeispiele** auf den verschiedenen Ebenen für die Formel $\Diamond\Box$**A**:

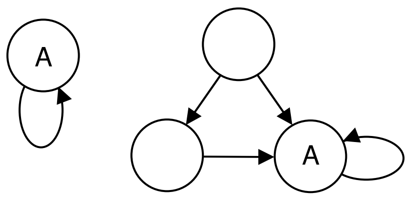
2 Modelle **mit** Anfangssituation



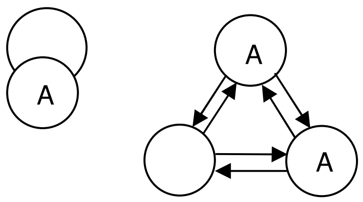
2 Gegenbeispiele **mit** Anfangssituation



2 Kripke-Struktur-Modelle (**bel.** Anf.)



2 Kripke-Struktur-Gegenbeispiele (**bel.** Anf.)



Rahmen-Modelle (bel. Anf., bel. Bel.) gibt es keine – alle Rahmen sind Gegenbeispiele! – Warum?

Annahme: (#) Es existiert ein Rahmen Rah mit $\Diamond\Box A$ für alle Belegungen und jede Wahl d. Anfangssituation.
Belege (per Bel) alle Knoten K leer, so dass überall $\neg A$ gilt. $\Rightarrow$ (∗) Jeder Knoten mit $\Box A$ hat keine Nachfolger (*sonst gälte im Nachfolger A und (s.o.)* $\neg A$).
Sei K_A Anfangsknoten in Rah mit Belegung Bel,
$\Rightarrow$ In K_A gilt $\Diamond\Box A$ – wegen (#)
$\Rightarrow$ K_A hat Nachfolgerknoten L, in dem $\Box$A gilt.
$\Rightarrow$ (○) L hat keine Nachfolger – wegen (∗)
Sei nun L Anfangsknoten in Rah mit Belegung Bel
$\Rightarrow$ In L gilt $\Diamond\Box A$ – wegen (#)
$\Rightarrow$ (¬○) L hat Nachfolger, Widerspruch!

Weitere semantischen Begriffe und Zusammenhänge:

Äquivalent ($\equiv$) sind ML-Formeln, die genau **dieselben** (erfüllenden) **Kripke-Modelle** (Rah, Bel, Anf) haben.

Beispiele: $\Box(A\wedge B) \equiv (\Box A\wedge\Box B)$, $\Diamond\neg A \equiv \neg\Box A$

Allgemeingültig (ML-Tautologien, MLTs) sind ML-Formeln, die in allen Kripke-Interpretationen (Rah, Bel, Anf) gelten.

Beispiele: $\Box(A\to B) \to (\Box A\to\Box B)$ (**K** wie Kripke),
$\Diamond(A\vee B) \leftrightarrow (\Diamond A\vee\Diamond B)$

Erfüllbar sind ML-Formeln, die **mindestens ein** (erfüllendes) **Kripke-Modell** (Rah, Bel, Anf) haben.

Man kann

- in einer tautologischen AL-Formel alle Vorkommen einer Aussagevariable durch dieselbe ML-Formel ersetzen, und man erhält eine ML-Tautologie (**Substitutionslemma**);
- in einer ML-Formel φ ein Vorkommen einer Teilformel durch eine dazu äquivalente ML-Formel ersetzen, und man erhält eine zu φ äquivalente ML-Formel (**Ersetzungslemma**);
  Substitutions- und Ersetzungslemma gelten bezüglich aller drei Ebenen.
- eine zu einer gegebenen ML-Formel äquivalente ML-Formel konstruieren, bei der alle Negationen unmittelbar vor Aussagevariablen stehen – eine **Normalform**.

Ein korrekter und vollständiger **Kalkül** zur Ableitung von ML-Tautologien (MLTs) umfasst

- AL-Axiome, AL-Ableitungsregeln (also alle **AL-Tautologien**)
- Substitution und Ersetzung in MLTs
- die 2 **ML-Axiome**: (K) und $\Box A \leftrightarrow \neg\Diamond\neg A$
- **Notwendigkeit** der ML-Tautologien:
  φ MLT $\Rightarrow$ $\Box\varphi$ MLT
- **ML-Modus-Ponens**:
  φ und $\varphi\to\psi$ MLT'en $\Rightarrow$ ψ MLT

Man kann auch den Werkzeugkasten entsprechend erweitern.

Zusammenhänge Rahmen – Axiome

Besondere Graphen/Relationen-Eigenschaften der Rahmen entsprechen der Einführung besonderer Axiome der zugehörigen modalen Logik.

- Der Graph R wird festhalten, Belegung und Anfangssituation dürfen variieren.
- Welche Eigenschaft muss der Graph haben, damit stets das Axiom gilt?
- Welche Axiome gelten bei gegebener Grapheneigenschaft?

Beispiele:

Übergangsrelation R		Axiom
reflexiv	$\Leftrightarrow$	$\Box A\to A$
transitiv	$\Leftrightarrow$	$\Box A\to\Box\Box A$
symmetrisch	$\Leftrightarrow$	$A\to\Box\Diamond A$
linkstotal	$\Leftrightarrow$	$\Box A\to\Diamond A$

Man überlegt jeweils leicht, dass aus der R-Eigenschaft und der Prämisse im Axiom die Konklusion folgt, und findet bei fehlender R-Eigenschaft ein Gegenbeispiel.

Spezialfall: Gültigkeit in einer Menge von Situationen

- $Rah = (K,R)$,
- K = gegebene Menge MB von Belegungen von $A_1, \ldots, A_n$
- $bel = \mathrm{id}_{MB}$ (Situationen = Belegungen)
- $R = K \times K$ (1 Schritt von allen nach allen)
- $Anf \in MB$ (beliebig)

Dann bedeutet für AL-Formeln …

- notwendig – in allen Situationen aus MB
- möglich – in mindestens einer MB-Situation

„Spezieller Spezialfall“:
Gültigkeit in beliebigen Belegungen – wie oben aber MB = Menge aller totalen Belegungen (aller A_i).

Dann bedeutet für AL-Formeln …

- notwendig – allgemeingültig
- möglich – erfüllbar
- Semantische Begriffe der AL syntaktisch **innerhalb** dieser Logik ausdrückbar!

Anwendungen der Modallogik

Deontische Logik

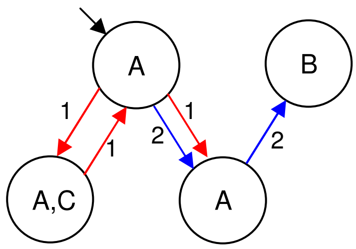
- $\Box$: Es ist geboten (Pflicht), dass ...
- $\Diamond$: Es ist erlaubt, dass ...

Nachvollziehbares Axiom: $\Diamond\varphi \coloneqq \neg\Box\neg\varphi$

→ Zusammenhänge mit juristischem Denken.

weitere Anwendungen: normative, alethische, epistemische, doxastische, intensionale, dynamische, temporale Logik

Verallgemeinerung der (monomodalen) Modallogik



In manchen Fragen ist es sinnvoll mehrere unterschiedliche Übergangsrelationen $R_1, R_2, R_3, \ldots$ zwischen den Welten nebeneinander zu betrachten: Entsprechend gibt es dann unterschiedliche Modaloperatoren $\Box_1$, $\Diamond_1$, $\Box_2$, $\Diamond_2$ usw. → **Multimodale** Logiken

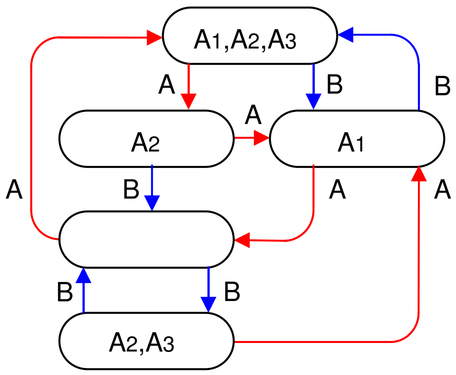
(Altes) Beispiel: Wir schreiben eingangs im Diagramm der Zustandsübergänge des Brettspiels an **jede** Kante A oder B (Relationen R_A, R_B,) entsprechend der gewählten Einwurf-Öffnung, siehe rechts oben.

Temporallogik

Bei Spezifikation und Test realer Systeme möchte man oft Anforderungen an zeitliche Abläufe darstellen und behandeln. Manchmal hat man **quantitative** Anforderungen an Zeiträume und Zeitpunkte,

- Die Seite muss spätestens 4 Sekunden nach dem Aufruf vollständig auf dem Bildschirm aufgebaut sein.

Multimodalbeispiel:



- Die Uhr soll zu jeder vollen Stunde schlagen.
- Die Angestellten sollen jeden Werktag 8 Stunden arbeiten.

Dazu eignen sich z.B.

- Zeitnetze (z.B. Timer Nets[1])
- Zeitautomaten (Timed Automata[2])

Manchmal genügt es, dass („qualitativ“) in der Abfolge beispielsweise

- etwas Erwünschtes
  - im nächsten Schritt oder zumindest
  - irgendwann oder
  - solange notwendig, bzw.
- etwas Unerwünschtes
  - nie oder
  - höchstens alle n-mal

der Fall ist.

Beispiele:

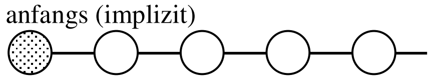
- Wenn ich die Nachricht immer wieder sende, muss sie irgendwann ankommen.
- Solange ich noch keine Empfangsbestätigung habe, sende ich die Nachricht immer wieder.
- Das Kommunikationsmedium verfälscht keine Nachricht.

Dazu eignet sich Lineare Temporale Aussagenlogik, **PLTL** (Propositional Linear Temporal Logic) mit folgenden Ausdrucksmitteln: siehe Abb. nächste Seite

Ausdrucksmittel PLTL:

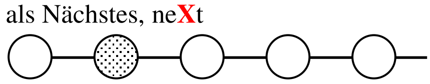
– Aussagenlogik

- φ anfangs (implizit)

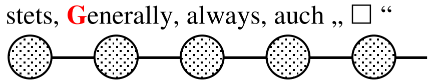


– temporale Operatoren …

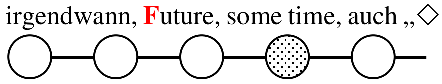
- $X\varphi$ als Nächstes, ne**X**t



- $G\varphi$ stets, **G**enerally, always, auch „ $\Box$ “

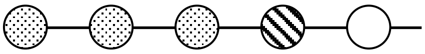


- $F\varphi$ irgendwann, **F**uture, some time, auch „$\Diamond$“



[1] vgl. B. Baumgarten: Druskininkai Lectures
[2] R. Alur, Dill: Theoretical Comp. Sci. 126

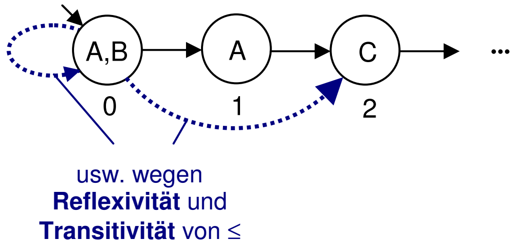
- $(\varphi \mathrm{U} \psi)$ mindestens bis, Until



„Abschneide-Semantik"

PLTL ist eine spezielle Modallogik mit dem Rahmen $(I\!N_0, \leq)$. Die potentiellen Modelle (Interpretationen) sind unendliche Folgen $\pi = (\pi_0, \pi_1, \ldots)$ von „Gegebenheiten" (Zuständen, Situationen), in denen jeweils alle Aussagevariablen in einer Menge $\pi_i \subseteq AV$ wahr sind (und die anderen – in $AV \setminus \pi_i$ – nicht).

Eine typische Kripke-Interpretation ist z.B.



Syntax: Die Formeln der PLTL sind induktiv definiert:

- Alle Aussagevariablen in AV sind PLTL-Formeln.
- Für alle PLTL-Formeln φ, ψ sind ebenfalls PLTL-Formeln ...
  $(\neg\varphi)$ und $(\varphi \wedge \psi)$ [3] (Aussagenlogik)
  $(\mathrm{X}\varphi)$, $(\mathrm{G}\varphi)$, $(\mathrm{F}\varphi)$, $(\varphi \mathrm{U} \psi)$ [4] (TemporaleLogik)

Achtung! In der Literatur werden teils auch andere Operatoren bzw. Schreibweisen verwendet.

Semantik: ... legt fest, wann eine PLTL-Formel auf einer Folge $\pi = (\pi_0, \pi_1, \ldots)$, $\pi_i \subseteq AV$, wahr ist.

Aussagenlogischer Anteil:

- Für $\varphi \in AV$: $\pi \models \varphi \;:\Leftrightarrow\; \varphi \in \pi_0$, d.h. gilt im ersten Folgenglied.
  ansonsten für PLTL-Formeln φ, ψ:
- $\pi \models \neg\varphi \quad :\Leftrightarrow \quad$ nicht $\pi \models \varphi$
- $\pi \models \varphi \wedge \psi \quad :\Leftrightarrow \quad \pi \models \varphi$ und $\pi \models \psi$
- $\vee$, $\rightarrow$ und $\leftrightarrow$ kann man mit auflisten oder als expandierbare Abkürzungen behandeln.

Temporaler Anteil:

- $\pi^i :=$ Endstück ab Glied i Abschneide-Semantik)
- $\pi \models \mathrm{X}\varphi \quad :\Leftrightarrow \quad \pi^1 \models \varphi \quad \varphi$ gilt im nächsten Glied.
- $\pi \models \mathrm{G}\varphi \quad :\Leftrightarrow \quad \forall i \geq 0 : \pi^i \models \varphi$
  φ gilt in allen Folgengliedern.
- $\pi \models \mathrm{F}\varphi \quad :\Leftrightarrow \quad \exists i \geq 0 : \pi^i \models \varphi$
  φ gilt in einem Folgenglied.
- $\pi \models \varphi \mathrm{U} \psi :\Leftrightarrow$
  $(\exists i \geq 0 : \pi^i \models \psi) \wedge (\forall 0 \leq j < i : \pi^j \models \varphi)$
  ψ gilt in einem Folgenglied, und mindestens bis unmittelbar davor gilt φ.

Zusammenhänge und weitere Temporaloperatoren

Die Operatoren G und F können durch U (und AL) ausgedrückt werden:

- $\mathrm{F}\varphi \;\equiv\; \mathrm{T}\,\mathrm{U}\,\varphi \;\equiv\; (A \vee \neg A)\,\mathrm{U}\,\varphi$
- $\mathrm{G}\varphi \;\equiv\; \neg \mathrm{F} \neg\varphi \;\equiv\; \neg(\mathrm{T}\,\mathrm{U}\,\neg\varphi)$

[3] minimale Junktorenbasis

[4] PLTL-Operatoren minimal? Kommt gleich rechts.

Weitere Temporaloperatoren:

$\varphi \mathbf{R} \psi$: ψ gilt bis zu und einschließlich dem ersten Folgenglied, an dem φ gilt, sofern ein solches existiert; andernfalls gilt ψ für immer.

$\varphi \mathbf{W} \psi$: Wenn φ nicht für immer gilt, dann mindestens bis unmittelbar vor das – dann sicher kommende – erste Folgenglied, an dem ψ gilt.

$\mathrm{Op}_1 \varphi$: φ gilt unendlich oft.

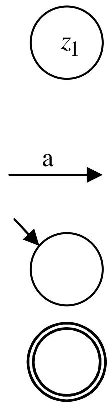
$\mathrm{Op}_2 \varphi$: φ gilt unendlich oft nicht.

$\mathrm{Op}_3 \varphi$: φ gilt nie.

Nichtdeterministische Automaten

Ein (ND-) Automat („über Σ") ist ein Quintupel $A = (Z, \Sigma, T, Z_{Anf}, Z_{Ziel})$ mit

- einer endlichen Menge Z, deren Elemente **Zustände** genannt werden;
- einer endlichen Menge Σ (**Alphabet**), deren Elemente **Symbole** genannt werden;
- einer dreistelligen **Transition**s-(Übergangs-) Relation $T \subseteq Z \times \Sigma \times Z$;
- einer Menge $Z_{Anf} \subseteq Z$, deren Elemente **Anfangszustände** genannt werden;
- einer Menge $Z_{Ziel} \subseteq Z$, deren Elemente **Zielzustände** genannt werden.

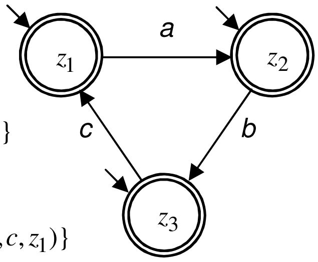


Beispiel:

$Z = Z_{Anf} = Z_{Ziel} = \{z_1, z_2, z_3\}$

$\Sigma = \{a, b, c\}$

$T = \{(z_1, a, z_2), (z_2, b, z_3), (z_3, c, z_1)\}$



[In der → herkömmlichen Automatentheorie ist die ND-Eigenschaft gewissermaßen gleichgültig, hier aber nicht.]

Ein Automat A definiert („akzeptiert") eine **Sprache** L(A) (endlicher Wörter) nach dem „Prinzip der möglichen Reisetagebücher":

Wir lassen uns in einem Anfangszustand absetzen, reisen entlang der Pfeile von Zustand zu Zustand und schreiben dabei die Pfeilanschriften mit. In jedem Endzustand dürfen wir aufhören, müssen aber nicht. Was wir bis dahin mitgeschrieben haben, ist ein Wort der Sprache.

(Formale Definition?)

Im Beispiel ist $\mathrm{L}(A) = \{\varepsilon, a, b, c, ab, bc, ca, abc, bca, \ldots\}$ $= \mathrm{Pref}(\mathrm{Suff}((abc)^*)) = (\varepsilon \mid c \mid bc)(abc)^*(\varepsilon \mid a \mid ab) \mid b$.

Die erste Angabe ist informell, die letzte ist ein **regulärer Ausdruck**.

Büchi-Automaten

Ein Automat A definiert **auch** eine sog. ω-Sprache $\mathrm{L}_\omega(A)$ **unendlicher Folgen**; A – obwohl unverändert – wird dann als **Büchi-Automat** bezeichnet:

Die Reisen werden unendlich lange, und zur ω-Sprache gehören („vom Büchi-Automaten akzeptiert" werden) alle Folgen („unendlichen Reisetagebücher"), bei denen

mindestens ein Zielzustand **unendlich oft** besucht wurde. (Formale Definition?)

Welche ω-Sprache definiert der obenstehende Automat als Büchi-Automat?

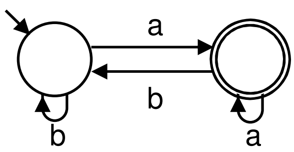
– $L_\omega(A) = (\varepsilon \mid c \mid bc)(abc)^\omega$, d.h. $(\varepsilon \mid c \mid bc)\, abcabc\ldots$

Wie kann ich ihn verändern, ohne seine ω-Sprache zu verändern?

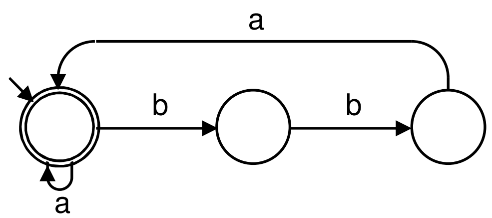
– Z.B. einen oder zwei Zustände aus Z_{Ziel} streichen.

Ordnen Sie den Büchi-Automaten A–D die richtigen „**Büchi-akzeptierten**" ω-Sprachen 1–4 über {a,b} zu.

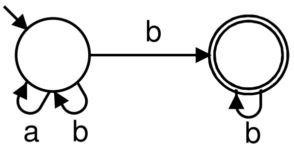
A.



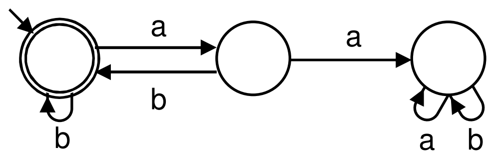
B.



C.



D.



ω-Sprachen :

1. unendlich viele a's drin
2. nur endlich viele a's drin
3. nie zwei a's hintereinander (kein ...aa...)
4. b's nur zu zweit hintereinander (kein ...aba/ bbb...)

Lösung: A1 B4 C2 D3

Sei AV_φ := Menge der in φ vorkommenden Aussagevariablen. Jedes (die Formel **wahr** machende) **Modell** einer PLTL-Formel φ ist eine Folge $\pi = (\pi_0, \pi_1, \ldots)$, $\pi_i \subseteq AV_\varphi \subseteq AV$. Wegen der Endlichkeit von AV_φ können wir auch die endliche Menge $\Sigma := \mathbf{P}(AV)$ als ein **Alphabet** (mit etwas ungewöhnlichen Symbolen) betrachten!

Beispiel $AV_\varphi = \{A, B, C\}$ →

$\Sigma = \{ \varnothing, \{A\}, \{B\}, \{C\}, \{A,B\}, \{A,C\}, \{B,C\}, \{A,B,C\}\}$.

Mod(φ), die Menge aller Modelle von φ, ist dann eine ω-Sprache über Σ.

Satz: PLTL-Formel-Modelle werden von Büchi-Automaten akzeptiert.

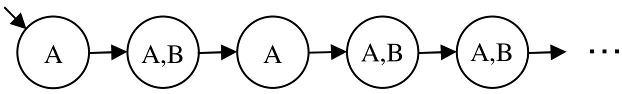
Zu jeder PLTL-Formel φ über der Aussagevariablenmenge AV existiert ein Büchi-Automat A mit Alphabet $\Sigma = \mathbf{P}(AV)$, so dass $\mathrm{Mod}(\varphi) = L_\omega(A)$.

Beweis → Literatur[5]

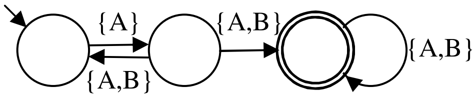
Beispiel mit AV={A,B}

PLTL-Formel	Wirkung auf Modellsituationen
$A \wedge \neg B \wedge$	Die erste Situation ist {A} &
$\mathsf{G}(\neg B \to \mathsf{X}B) \wedge$	keine 2 hintereinander ohne B &
$\mathsf{G}(B \to A) \wedge$	stets wo {B,…}, dort {A,B} &
$\mathsf{G}(B \to \mathsf{X}$	stets folgt auf {B,…} sofort
$((A \wedge \neg B) \vee \mathsf{G}B) \wedge$	{A} oder für immer {B,…} &
$\mathsf{FG}B$	irgendwann für immer {B,…}.

Modellbeispiel:



Büchi-Automat für die Modelle der Formel:



Satz: **Büchi-Entscheidbarkeit**

Entscheidbar sind:

die Frage, ob für einen Büchi-Automaten A die Sprache der akzeptierten Wörter **nicht leer** ist, d.h. $L_\omega(A) \neq \varnothing$[6], also auch die **Erfüllbarkeit** und **Allgemeingültigkeit** jeder **PLTL-Formel**.

Beweis 2. Hälfte: Man konstruiert die Büchi-Automaten A und B mit $\mathrm{Mod}(\varphi) = L_\omega(A)$ und $\mathrm{Mod}(\neg\varphi) = L_\omega(B)$. Nun ist φ erfüllbar ⇔ $L_\omega(A) \neq \varnothing$ und φ allgemeingültig ⇔ $L_\omega(B) = \varnothing$.

ω-reguläre Ausdrücke

Die Menge der **regulären** Ausdrücke Reg(Σ) über einem Alphabet Σ ist induktiv definiert:

- ∅, ε ∈ Reg(Σ)
- Σ ⊆ Reg(Σ) (Zeichen "=" Wort der Länge 1)
- $p, q \in$ Reg(Σ) ⇒ p^*, $(p|q)$, $(pq) \in$ Reg(Σ).

Jeder reguläre Ausdruck *reg* definiert rekursiv eine **Sprache** L(*reg*):

- $L(\varnothing) := \varnothing$
- $L(a) := \{a\}$ für alle $a \in \Sigma$
- $L(p^*) := (L(p))^*$
- $L((p|q)) := L(p) \cup L(q)$
- $L((pq)) := \{vw \mid v \in L(p) \wedge w \in L(q)\}$

Beispiel: L(a*bb*(aa*bb*)*) = L((a|b)*b) = Wörter über {a, b}, letztes Zeichen b.

An **Klammern** werden oft weggelassen:

- die äußersten, sowie die die sich erübrigen wegen ...
- Einführung von Bindungsprioritäten (z.B. $* > \circ > |$)
- oder Assoziativität.

Die Menge der **ω-regulären Ausdrücke** ω-Reg(Σ) über einem Alphabet Σ ist induktiv definiert:

- ∅ ∈ω-Reg(Σ)
- $p \in \mathrm{Reg}(\Sigma) \wedge \varepsilon \notin L(p) \Rightarrow p^\omega \in \omega\text{-Reg}(\Sigma)$

[5] vgl. z.B. http://i12www.ira.uka.de/~pschmitt/FormSys/FormSys1314/skriptum.pdf

[6] s.o.

- $p, q \in \omega\text{-Reg}(\Sigma) \quad \Rightarrow \quad (p+q) \in \omega\text{-Reg}(\Sigma)$
- $p \in \text{Reg}(\Sigma) \wedge q \in \omega\text{-Reg}(\Sigma)$
  $\Rightarrow \quad (pq) \in \omega\text{-Reg}(\Sigma)$

Jeder ω-reguläre Ausdruck q definiert rekursiv eine **Sprache $L^{\omega}(q)$** unendlicher Wörter:

- $L\omega(\varnothing) \quad := \quad \varnothing$
- $L\omega((p+q)) \quad := \quad L\omega(p) \cup L\omega(q)$
- $L\omega(p\omega) \quad := \quad \{v_1 v_2 \cdots \mid \forall i \geq 1 : v_i \in L(p)\}$
- $L\omega((pq)) \quad := \quad \{vw \mid v \in L(p) \wedge w \in L\omega(q)\}$

Büchi-Automaten und ω-reguläre Ausdrücke „definieren **dieselben Sprachen** über $\Sigma = \mathbf{P}(AV)$," und daher sind auch PLTL-Formel-Modellmengen durch ω-reguläre Ausdrücke darstellbar.

So können wir den ω-Sprachen auf der vorigen Seite rechts die folgenden ω-regulären Ausdrücke zuordnen:

A. unendlich viele a drin: $(b^*a)^{\omega}$
B. nur endlich viele a drin: $(b^*a)^*b^{\omega}$
C. nie zwei a's hintereinander (kein ...aa...):
$(b^*ab)^* \; (b^{\omega} \mid (b^*ab)^{\omega})$
D. b's nur zu zweit hintereinander (kein ...aba/ bbb...):
$(a^*bba)^* \; (a^{\omega} \mid (a^*bba)^{\omega})$

Es gibt PLTL-Anwendungsbereiche, bei denen jede Situation mit **genau einer Aussagevariable** belegt ist.

Mit $AV = \{A_1, \cdots, A_n\}$ entspräche das dem Axiom „$\Box\text{XOR}(A_1, \cdots, A_n)$", d.h. $\Box\, [\, (A_1 \vee \cdots \vee A_n) \wedge$ ...
... $(\neg(A_1 \wedge A_2)) \wedge (\neg(A_1 \wedge A_3)) \wedge \cdots \wedge (\neg(A_{n-1} \wedge A_n))\,]$

Man beschreibt oft Systeme mittels möglicher **Aktionen** $Act = \{a_1, \cdots, a_n\}$, die sequentiell beobachtet werden, z.B. die Syst.-Beobachtung = Aktionenfolge $a_1 a_2 a_3 a_1$ ...
$A_i \;\hat{=}\;$ „a_i ist als letztes geschehen" (anfangs nil $\hat{=}$ noch keine Aktion) → in jeder Situation ist dann genau eine Aussagevariable wahr, z.B.

$$nil\; A_1\; A_2\; A_3\; A_1 \cdots \text{ für } a_1\, a_2\, a_3\, a_1 \cdots$$

Dann brauchen wir auch keine Potenzmenge mehr für das Alphabet des Büchi-Automaten oder des ω-regulären Ausdrucks. ☺ Zu unserer Beispielreihe passen dann die „einelementigen PLTL-Formeln über {a,b}"

A. unendlich viele a drin: GFa und ¬FGb
B. nur endlich viele a drin: ¬GFa und FGb
C. nie zwei a's hintereinander (kein ...aa...):
G(a→¬Xa), ¬F(a+Xa)
D. b's nur zu zweit hintereinander (kein ...aba/ bbb...):
¬F(a∧Xb∧XXa)∧¬F(b∧Xb∧XXb)

PLTL-Kritik: das **Nicht-Falsifizierbarkeits-Problem**

Soll man Systemeigenschaften spezifizieren, deren Verletzung nie festzustellen ist – weder direkt noch indirekt? (→ K. Popper)

Beispiel: Black Box, zeigt ab jetzt jede Sekunde eine Zahl, und zwar 0 oder 1.
Beobachtbar: lediglich **Anfangsstücke** (beliebig lange aber **endlich**)!

• Wunscheigenschaft **1**: abwechselnd 0 und 1!
PLTL: $\Box[\, (0 \rightarrow X1) \wedge (1 \rightarrow X0)\,]$

Beobachtung dass **einwandfrei**: **unmöglich**
(„meta-☺" erst im ∞)
Beobachtung dass **fehlerhaft**: **möglich** z.B. 00 ☹,
aber zeitlich **nicht einzugrenzen**: 10101010101010 ☺,

• Wunscheigenschaft **2**: irgendwann 1! PLTL: $\Diamond 1$
Beobachtung dass **einwandfrei**: **möglich** z.B. 00001 ☺,
aber zeitlich **nicht einzugrenzen**: 000000000000000 ☹,
Beobachtung dass **fehlerhaft**: **unmöglich**
(„meta-☹" erst im ∞),
→ http://www.bernd-baumgarten.de/Dru/PostDruA.pdf, S.1-11

Beschreibungslogik(en)

(BL, Description Logics, DL)

Wie können wir ...

- **Zusammenhänge** zwischen **Eigenschaften** (Prädikaten, **Konzepten**) und (meist **2-stelligen**) **Beziehungen** (Relationen, **Rollen**) ausdrücken?
- **aus** atomaren **Konzepten** und **Rollen** komplexere **Konzepte** und **Rollen** konstruieren?
- aus einem Wissensfundus (Wissens(daten)bank, knowledge base) mit gegebenen Konzepten und Rollen **Schlussfolgerungen** ziehen?

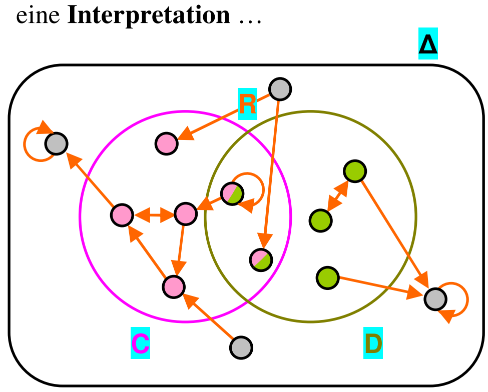
vgl. auch: Wissensdarstellung, Semantische Netze, Ontologien, Begriffssprachen, Datenbanken → Wikipedia, Google, The Description Logic Handbook

Im Bereich der **Menschen** gilt z.B.:

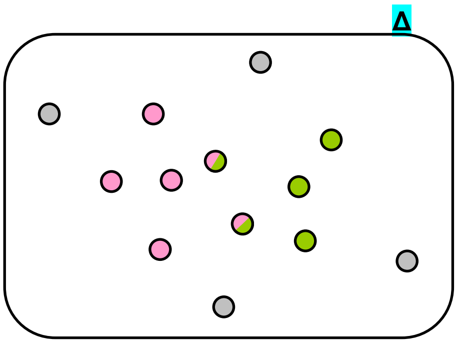
1. **Mutter** ist man genau dann, wenn man Elternteil und Frau ist.
   in PL1: $\forall x : (Mutter(x) \leftrightarrow (Elternteil(x) \wedge Frau(x)))$
   in BL: Mutter := Elternteil ⊓ Frau
2. **Elternteil** ist man genau dann, wenn ein Mensch existiert, den man als Kind hat.
   in PL1: $\forall x : (Elternteil(x) \leftrightarrow \exists y (HatAlsKind(x, y)))$
   in BL: Elternteil := ∃ HatAlsKind.⊤ (⊤= true/top)
   d.h. steht in Beziehung **HatAlsKind** zu mindestens einem Menschen (der Art **beliebig** (top))
3. Ursula von der Leyen **ist Mutter.**
   in PL1: Mutter(UrsulaVonDerLeyen)
   in BL: Mutter := Elternteil ⊓ Frau

Die Merkmale **einer** BL (**ALC – Attributive Language with Complements)** lassen sich so illustrieren:

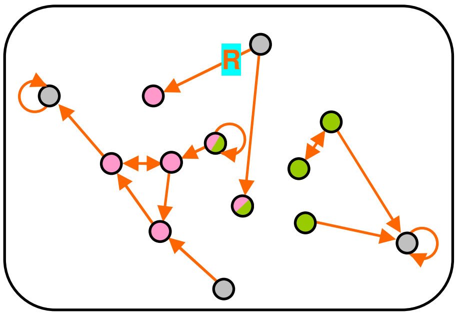
- eine **Interpretation** …



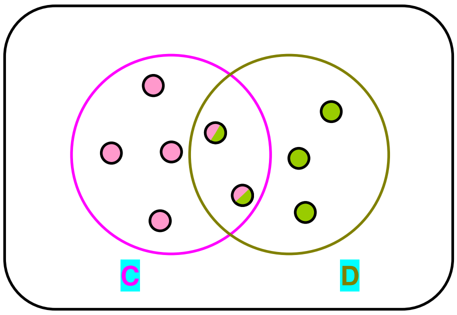
- … mit ihren Bestandteilen **Domäne** Δ



- … **Rolle** R (↔ entspricht → & ←)

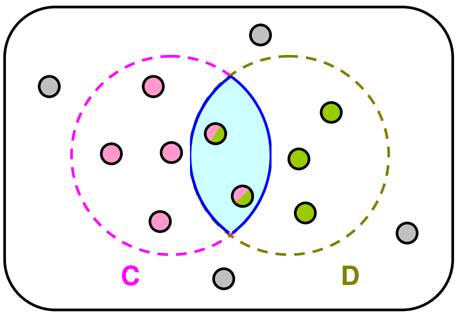


- … **Konzepten** C und D

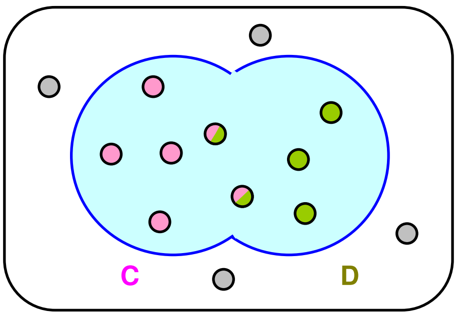


ALC kennt folgende Möglichkeiten der Definition neuer Konzepte aus alten:

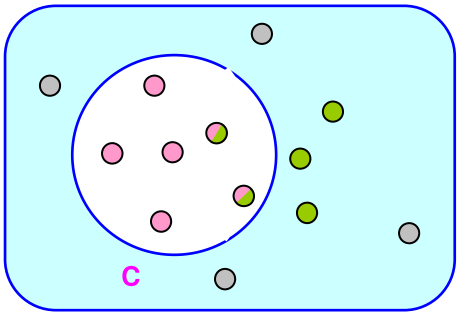
- **Durchschnitt** $C \sqcap D$



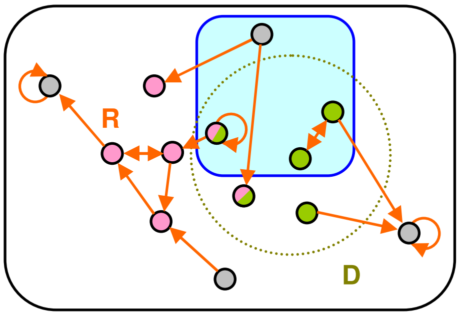
- **Vereinigung** $C \sqcup D$



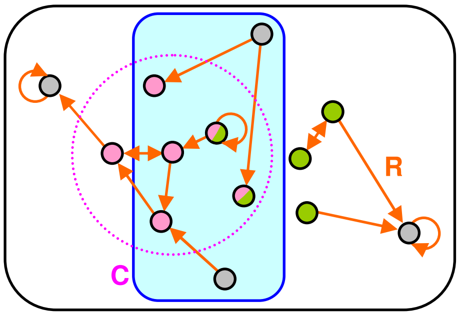
- **Komplement** $\neg C$



- **Existenz** von R-bezogenen Elementen in gegebenem Konzept $\exists R.D$



- R-bezogene Elemente **alle** in gegebenem Konzept $\forall R.C$



ALC (<u>**A**</u>ttributive <u>**L**</u>anguage with <u>**C**</u>omplements) ist <u>eine</u> elementare Beschreibungslogik.

(Es gibt **zahlreiche Beschreibungslogiken** mit weiteren Fragestellungen, Begriffen und Symbolen)

ALC-Syntax über einer Menge N_C von **Konzeptnamen** und einer Menge N_R von **Rollennamen**

- $C \in \{\top, \bot\} \cup N_C \Rightarrow C$ ist Konzeptterm
- C, D Konzeptterme und $R \in N_R \Rightarrow C \sqcap D$, $C \sqcup D$, $\neg C$, $\forall R.C$ und $\exists R.C$ sind auch Konzeptterme.

Formale **Semantik** von ALC

Eine **ALC-Interpretation** $I = (\Delta^I, \bullet^I, \circ^I)$ besteht aus

- einer nichtleeren Menge Δ^I **Domäne**, **Grundmenge**
- einer Abbildung $\bullet^I : N_C \to \mathbf{P}\Delta^I$
  Konzeptname $C \mapsto$ Konzept $C^I \subseteq \Delta^I$
- einer Abbildung $\circ^I : N_R \to \mathbf{P}(\Delta^I \times \Delta^I)$
  Rollenname $R \mapsto$ Rolle $R^I \subseteq \Delta^I \times \Delta^I$

I wird wird von den Konzeptnamen aus rekursiv auf alle Konzeptterme fortgesetzt:

Sind *C*, *D* Konzeptterme, so ist ...

$\bot^I := \varnothing$ *bottom*

$\top^I := \Delta^I$ *top*

$(\neg C)^I := \Delta^I \setminus C^I \setminus$ *Gegenteil*

$(C \sqcap D)^I := C^I \cap D^I$ *beides*

$(C \sqcup D)^I := C^I \cup D^I$ *mind. eines von beiden*

$(\exists R.C)^I := \{x \in \Delta^I \mid \exists y : (x,y) \in R^I \wedge y \in C^I\}$

in Beziehung R zu mindestens einer ***C-Instanz***

$(\forall R.C)^I := \{x \in \Delta^I \mid \forall y : (x,y) \in R^I \rightarrow y \in C^I\}$

in Beziehung R zu ausschließlich ***C-Instanzen***

Semantik-Übung in einer Gruppe:

Wir wählen als **Interpretation** die suggestiv naheliegenden **Domäne, Rollen** und **Konzepte** und bestimmen die **abgeleitete Interpretation** der folgenden **abgeleiteten Konzepte**.

- HierImRaum ⊓ (WohntDA ⊓ Männlich)
- HierImRaum ⊓ (Weiblich ⊔ SpieltMusikinstr)
- HierImRaum ⊓ (FährtSchi ⊔ SpieltMusikinstr)
- HierImRaum ⊓ (∃BefreundetMit.SchwimmtImVerein)
- HierImRaum ⊓ (∀IstKindVon.(∀IstKindVon.WohntDA))

Häufige Fragestellung: **Inklusion/Subsumtion**:
Ist ein Konzept in einem anderen enthalten, C ⊑ D?

Es gibt zwei Möglichkeiten:

1. C ⊑ D explizit **spezifizierbar** (in DL-Variante mit **Konzept-Inklusion**): Nur Interpretationen mit $C^I \subseteq D^I$ sind dann zulässig.
2. C ⊑ D (z.B. aus Axiomen korrekt) **ableitbar** (**Konzept-Subsumtion**): In allen Interpretationen ist automatisch $C^I \subseteq D^I$.

Logisch gültige (ableitbare) Subsumtionen sind z.B.

- Erwachsen ⊓ Männlich ⊑ Erwachsen
- Männlich ⊑ $\cancel{\forall}$HatAlsKind.Männlich
- ∀HatAlsKind.(Erwachsen ⊓ Männlich) ⊑ ∀HatAlsKind.Erwachsen
- ∀HatAlsKind.(Erwachsen ⊓ Männlich) $\not\sqsubseteq$ ∃HatAlsKind.Erwachsen
- (∀HatAlsKind.Erwachsen) ⊓ ∃HatAlsKind.⊤ ⊑ ∃HatAlsKind.Erwachsen

Wissensbanken in ALC

Eine **T-Box** (*terminological box*) ist eine endliche Menge von Konzeptdefinitionen (z.B. *C* := ∃R.D) und

Konzeptinklusionen (z.B. C ⊑ D). Eine **T-Box** enthält das allgemeine Wissen über die Domäne, Konzepte, Rollen, das strukturelle, terminologische Wissen.

Eine **Konzeptzuordnung** hat die Form *C*(*a*) , wobei *a* ein Individuenname und *C* ein Konzeptname ist.

Eine **Rollenzuordnung** hat die Form *R*(*a,b*), wobei *a,b* Individuennamen sind und *R* ein Rollenname ist.

Eine **A-Box** (*assertional box*) ist eine endliche Menge von Konzept- und Rollenzuordnungen. Eine **A-Box** enthält das Wissen über Konzepte und Rollenbeziehungen von Individuen und repräsentiert den Zustand der modellierten Welt.

Eine **Wissensbank** ist ein Paar (*T*,*A*), bestehend aus einer A-Box *A* und einer T-Box *T* .

Wissensbank

- T-Box
  - Konzeptdefinitionen
  - Konzeptinklusionen
- A-Box
  - Konzeptzuordnungen
  - Rollenzuordnungen

Entscheidungsprobleme

- Konzeptzugehörigkeit *C*(*a*)
  Gehört a zum Konzept C?
- Rollenüberprüfung *R*(*a,b*)
  Gehört (a,b) zur Rolle R?
- Subsumtion C ⊑ D
  Gehören alle zum Konzept C gehörenden Individuen zum Konzept D?
- Erfüllbarkeit
  Ist C nichtleer interpretierbar?

All dieses ist **entscheidbar**! (aber: Komplexität …)