Automatentheorie und Formale Sprachen

Teil 3 - Deterministische Endliche Automaten

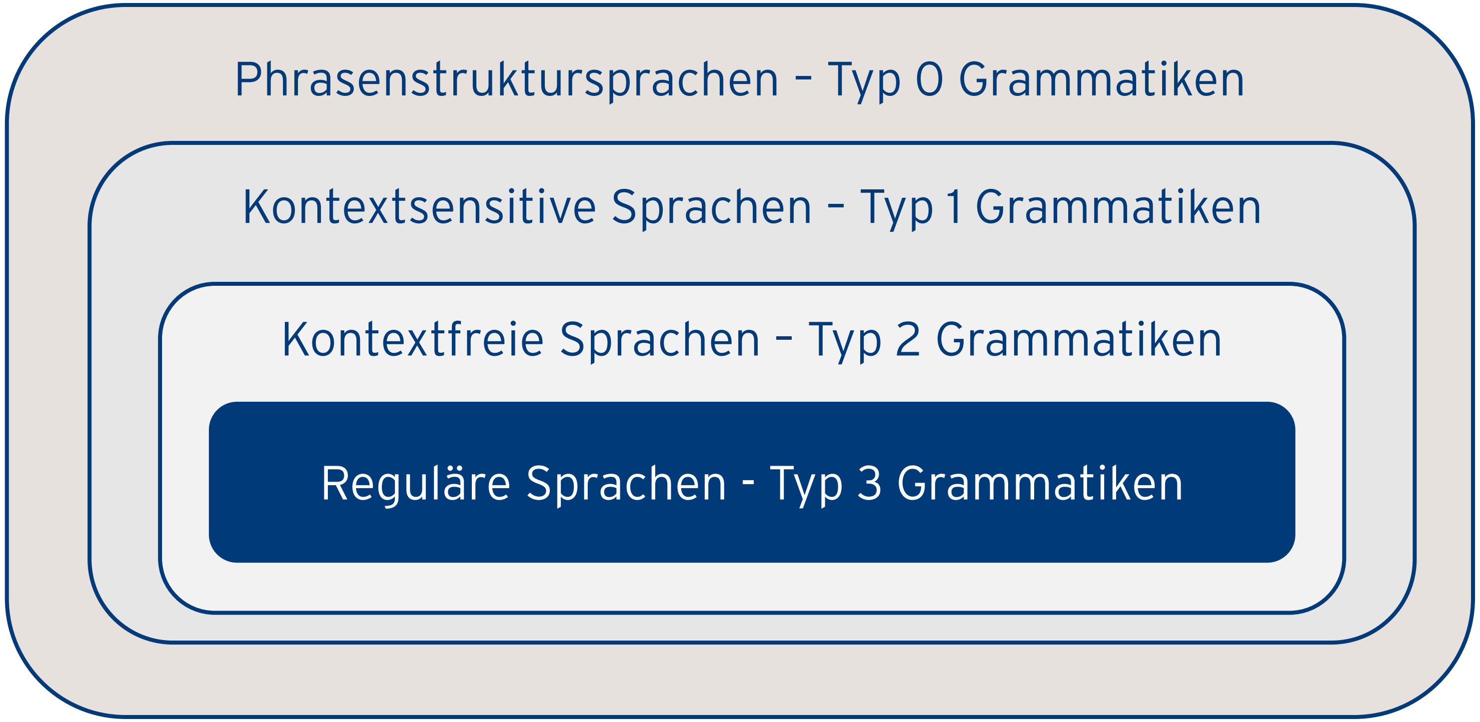
Prof. Dr. Hans-Werner Sehring

NORDAKADEMIE
HOCHSCHULE DER WIRTSCHAFT

Chomsky-Hierarchie formaler Sprachen

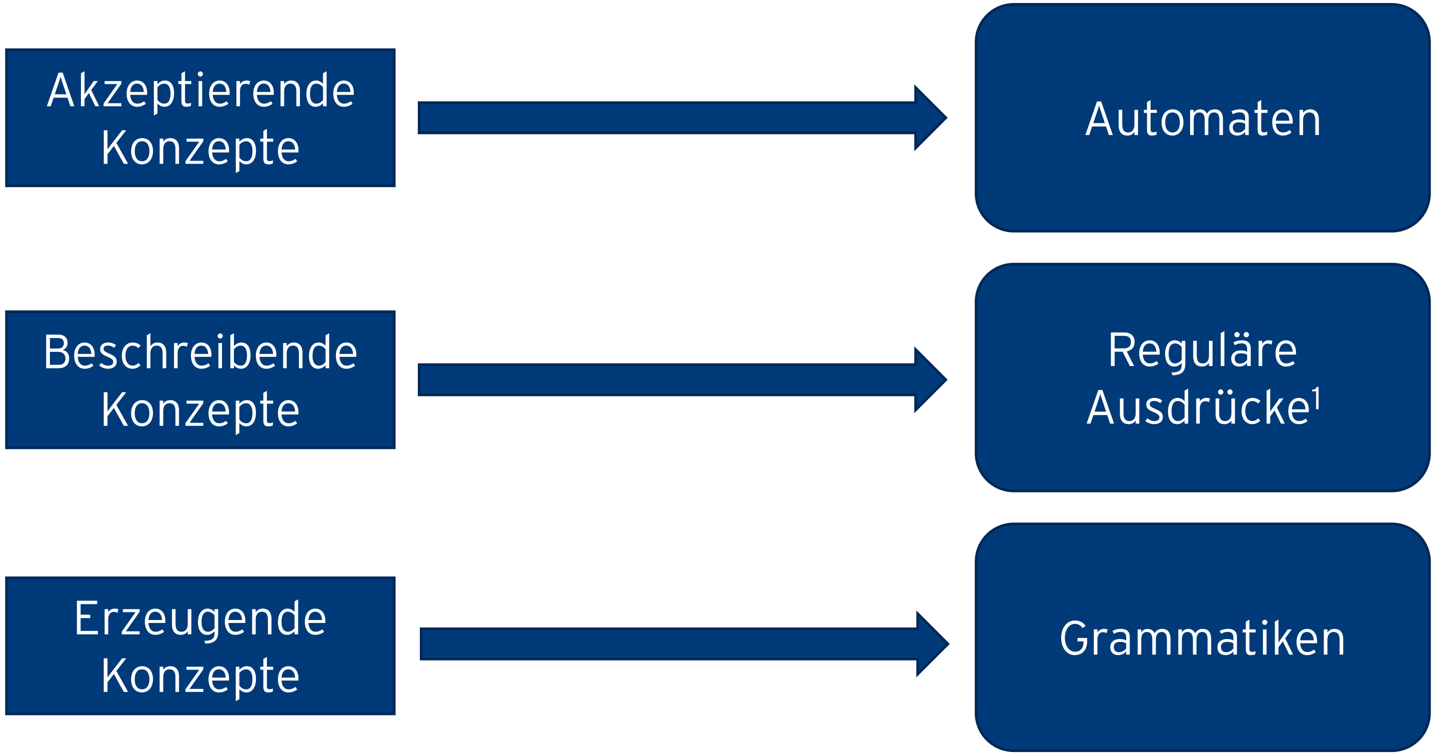
Formale Sprachen werden gemäß der Chomsky-Hierarchie klassifiziert.

Wir beginnen ganz „unten“/„innen“ bei den Regulären Sprachen.



Sprachdefinitionen

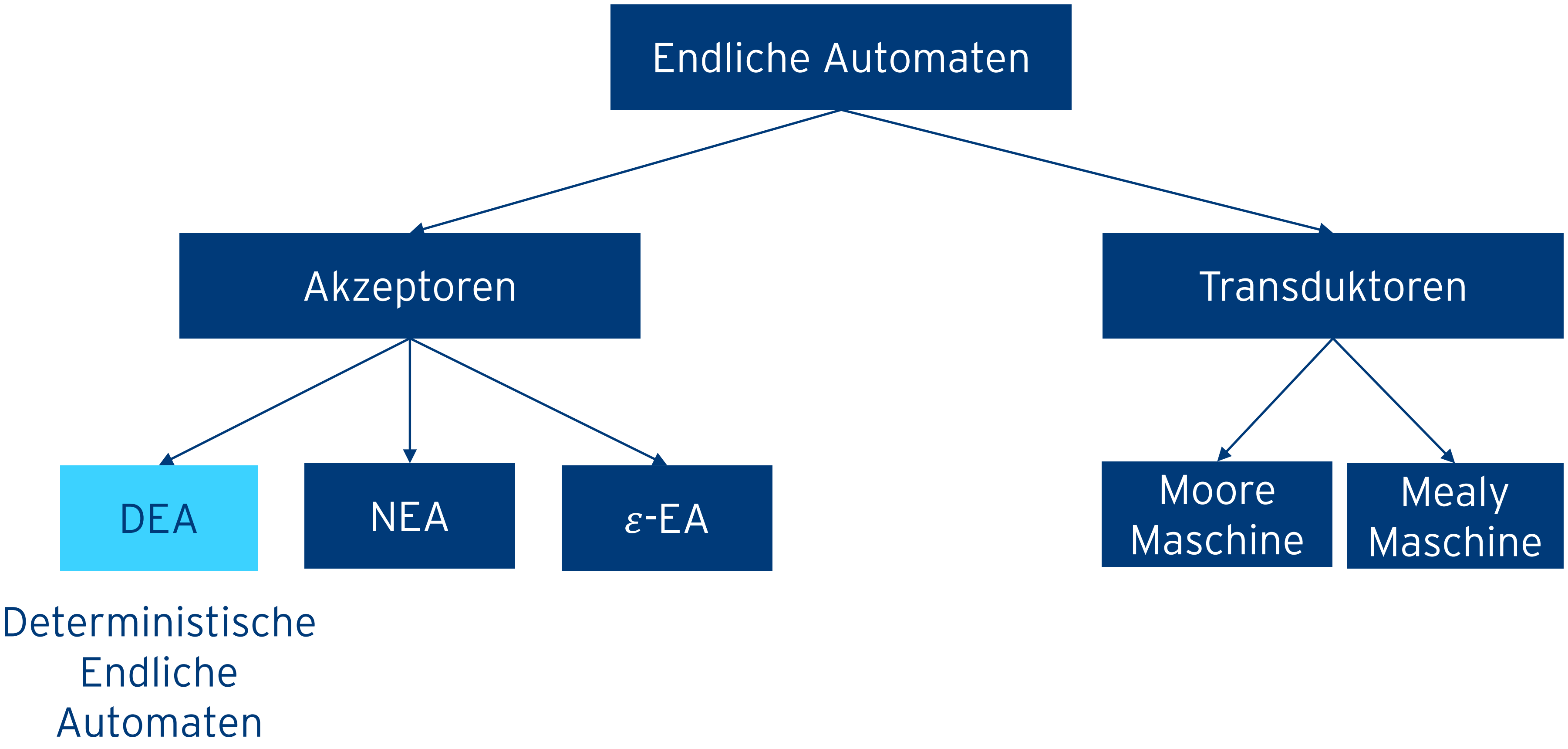
(Klassen) formale(r) Sprachen lassen sich auf verschiedene Arten definieren.



[1] Nur reguläre Sprachen

Endliche Automaten – Übersicht

Folgende Typen endlicher Automaten werden im Rahmen der Vorlesung behandelt:



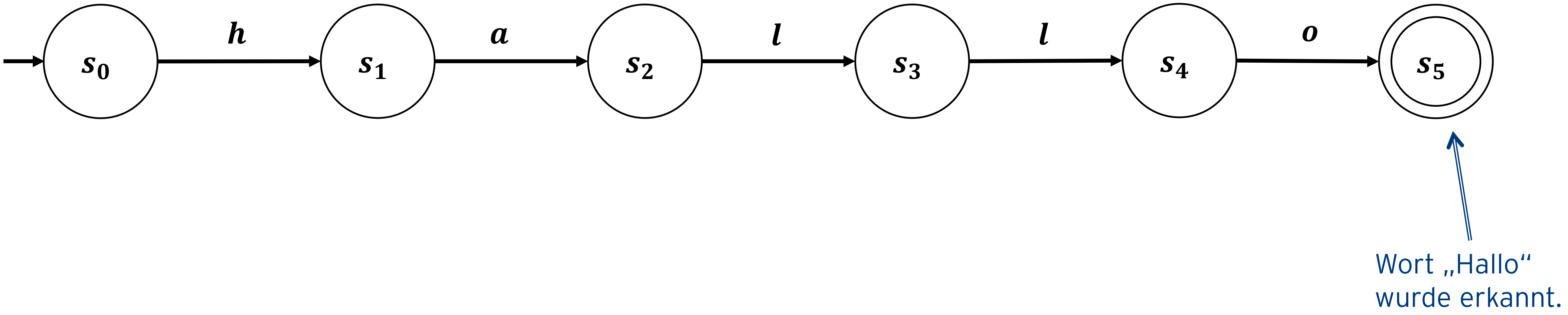
Lernziele

Welche Sprachen sind durch deterministische endliche Automaten beschrieben (akzeptiert)?

Was zeichnet einen deterministischen endlichen Automaten aus?

Initiales Beispiel

Einführendes Beispiel: Erkennung eines Wortes $\boldsymbol{w} = \boldsymbol{hallo}$

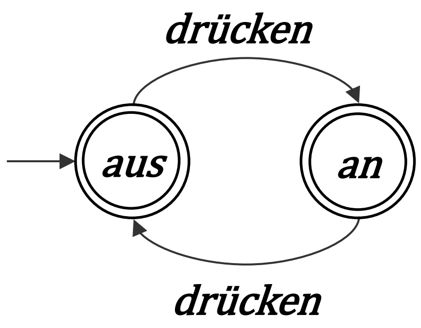


Die Abarbeitung eines Worts $\boldsymbol{w}$ lässt sich als deterministischer Automat abbilden.
Wird der **akzeptierende Zustand** $\boldsymbol{s_5}$ erreicht, gilt das Wort bzw. die Eingabe als erkannt.

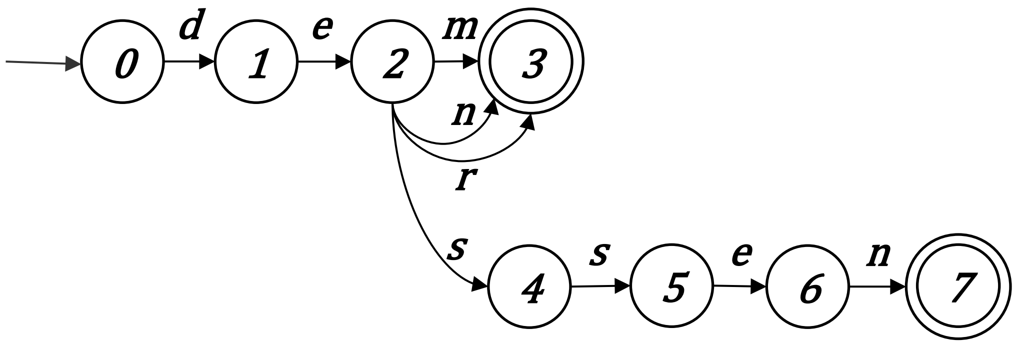
Der Automat fungiert als akzeptierendes Konzept.

Weitere Beispiele

Weitere Beispiele (Schalter und Wortsuche)

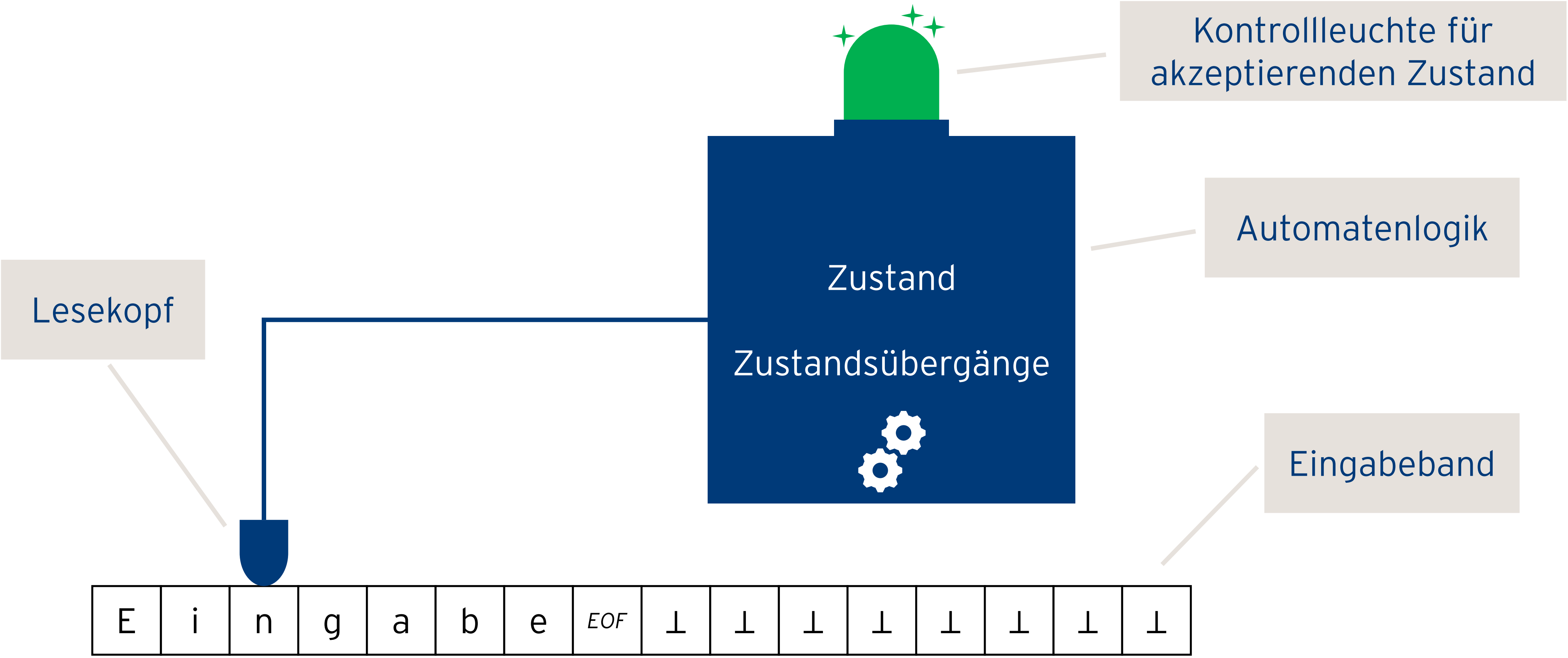


	drücken
aus	an
an	aus



	d	*e*	*m*	*n*	*r*	*s*
0	1					
1		2				
2			3	3	3	4
3						
4						5
5		6				
6				7		
7						

Versinnbildlichung eines endlichen Automaten



Deterministische endliche Automaten – DEA (1)

Ein deterministischer endlicher Automat $\boldsymbol{A_{DEA}}$ wird über ein Quintupel

$$\boldsymbol{A_{DEA}} = (\boldsymbol{\Sigma}, \boldsymbol{S}, \boldsymbol{\delta}, \boldsymbol{s_0}, \boldsymbol{F})$$

beschrieben. Die Menge der Zustände wird als

$$\boldsymbol{S} = \{\boldsymbol{s_0}, \dots, \boldsymbol{s_n}\}$$

notiert, wobei der **Startzustand** $\mathbf{s_0}$ ist und die **Endzustände** (akzeptierende Zustände) Elemente der Menge $\boldsymbol{S}$ sind. Endzustände (oder finale Elemente) werden über die Menge $\boldsymbol{F}$ angezeigt.

$$\mathbf{s_0} \in \boldsymbol{S}, \boldsymbol{F} \subseteq \boldsymbol{S}$$

Man spricht in diesem Fall von **endlichen** Automaten, da

$$|\boldsymbol{S}| + |\boldsymbol{\Sigma}| \neq \infty$$

Ferner spricht man in diesem Fall auch von **deterministischen** Automaten, da es bei jeder Eingabe $\boldsymbol{a} \in \boldsymbol{\Sigma}$ in einem bestimmten Zustand $\boldsymbol{s} \in \boldsymbol{S}$ **maximal einen Folgezustand** geben kann. $\boldsymbol{\delta}$ ist eine **Funktion**.

Deterministische endliche Automaten – DEA (2)

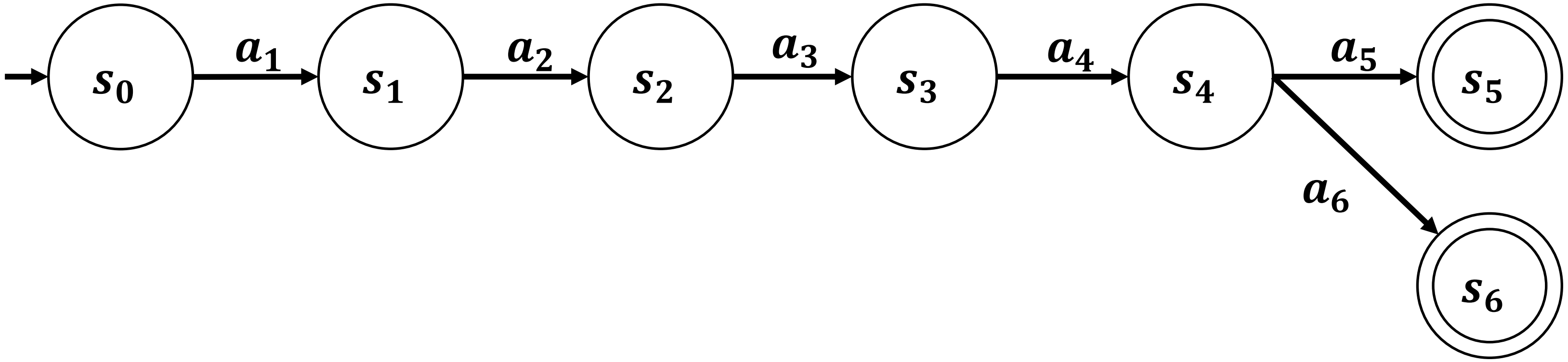
Die **Zustandsübergangsfunktion** $\boldsymbol{\delta}$ wird wie folgt ausgedrückt:

$$\boldsymbol{\delta}: \boldsymbol{S} \times \boldsymbol{\Sigma} \rightarrow \boldsymbol{S}$$

bzw.

$$\boldsymbol{\delta}(\boldsymbol{s}, \boldsymbol{a}) = \boldsymbol{s}' \qquad \boldsymbol{s}, \boldsymbol{s}' \in \boldsymbol{S}, \boldsymbol{a} \in \boldsymbol{\Sigma}$$

Die Zustandsübergangsfunktion lässt sich auch graphisch oder tabellarisch darstellen:

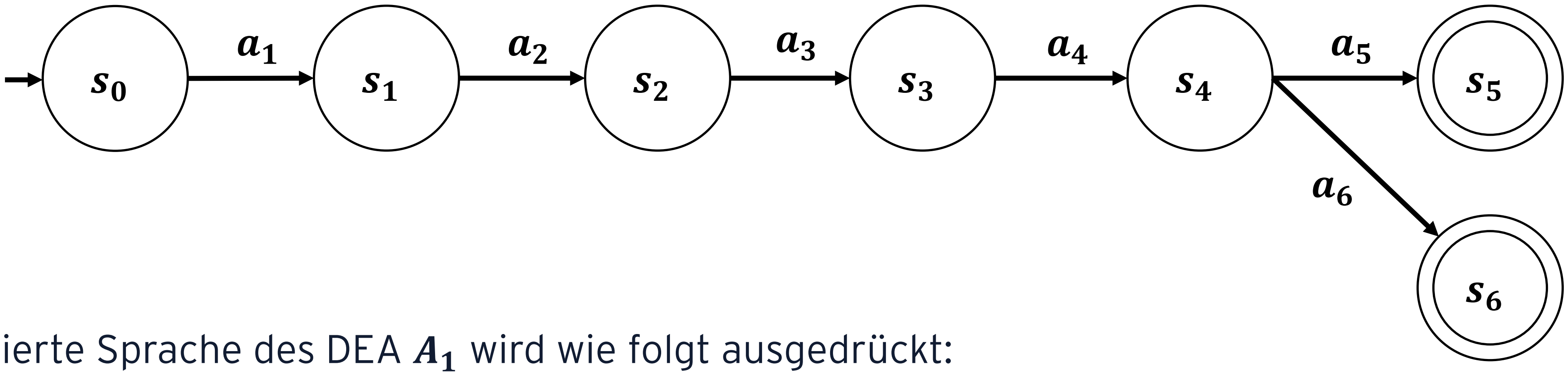


	a_1	a_2	a_3	a_4	a_5	a_6
s_0	s_1					
s_1		s_2				
s_2			s_3			
s_3				s_4		
s_4					s_5	s_6
s_5						
s_6						

Von einem DEA akzeptierte Sprache – Beispiel

Beispiel: $A_1 = (\Sigma, S, \delta, s_0, F)$

mit $\Sigma = \{a_i\}_{i \in I}, S = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6\}, F = \{s_5, s_6\}$



Die akzeptierte Sprache des DEA A_1 wird wie folgt ausgedrückt:

$$L_1 = \{a_1 a_2 a_3 a_4 a_5, a_1 a_2 a_3 a_4 a_6\}$$

Anmerkung:

Diese Sprachbeschreibung gibt lediglich die **Morphologie** des Automaten wieder, nicht die akzeptierten Worte als solche, da $a_1, \dots, a_6$ noch einer Instanziierung bedürfen.

Erweiterung von DEAen auf Wörter (1)

Erweitert man die Zustandsübergangsfunktion $\boldsymbol{\delta}$ auf die Eingabe von Wörtern $\boldsymbol{w}$, also mit

$$\boldsymbol{w} \in \boldsymbol{\Sigma}^*$$

gilt folgende Verknüpfung als Grundlage für die Worterkennung:

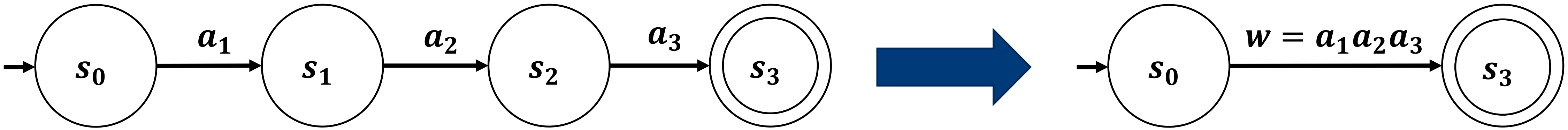
$$\boldsymbol{\delta}^*: \boldsymbol{S} \times \boldsymbol{\Sigma}^* \rightarrow \boldsymbol{S}$$

Dabei gilt:

$$\forall \boldsymbol{a} \in \boldsymbol{\Sigma}, \boldsymbol{w} \in \boldsymbol{\Sigma}^*\colon\ \boldsymbol{\delta}^*(\mathbf{s}, \mathbf{aw}) = \boldsymbol{\delta}^*(\boldsymbol{\delta}(\mathbf{s}, \mathbf{a}), \mathbf{w}),$$

und unter Berücksichtigung von $\boldsymbol{\Sigma}^*$ gilt

$$\forall \boldsymbol{s} \in \boldsymbol{S}\colon\ \boldsymbol{\delta}^*(\mathbf{s}, \boldsymbol{\varepsilon}) = \boldsymbol{s}$$



(In Kürze werden wir uns vergewissern, dass der verallgemeinerte Automat auch ein DEA ist.)

Erweiterung von DEAen auf Wörter (2)

Beispiel:

$$\boldsymbol{w = \ aab}$$

$$\boldsymbol{\delta^*(\mathrm{s}, \mathrm{aab}) = \delta^*(\delta(\mathrm{s}, a), \mathrm{ab}) = \delta^*(\delta(\delta(\mathrm{s}, a), a), b) = \delta^*(\delta(\delta(\delta(\mathrm{s}, a), a), b), \varepsilon) = \delta(\delta(\delta(\mathrm{s}, a), a), b)}$$

Das heißt, die einzelnen Zeichen $\boldsymbol{a} \in \boldsymbol{\Sigma}$ des Wortes $\boldsymbol{w}$ werden schrittweise verarbeitet.

Anmerkung:

$\boldsymbol{\delta^*}$ ist die **reflexiv-transitive Hülle** (auch **transitiver Abschluss**) der Zustandsübergangsfunktion $\boldsymbol{\delta}$.

Die **transitive Hülle** $\boldsymbol{R^+}$ einer Relation $\boldsymbol{R}$ auf einer Menge $\boldsymbol{M}$ ist wie folgt definiert:

$$\boldsymbol{xR^+y \Leftrightarrow \exists n \geq 0\colon \exists x_1, \ldots, x_n \in M\colon \ xRx_1, \ldots, x_nRy}$$

Somit ergibt sich die reflexiv-transitive Hülle $\boldsymbol{R^*}$ über:

$$\boldsymbol{xR^*y \Leftrightarrow x = y \ \vee xR^+y}$$

Anwendung: Definition von Ableitungen über dem reflexiv-transitiven Abschluss der Transitionsfunktion (hier $\boldsymbol{\delta}$).

DEA-Konfigurationen (1)

Konfiguration eines Automaten: Alternativ lassen sich Verarbeitungsfolgen durch Zustandsfolgen über Paare

$$k \in S \times \Sigma^*$$
$$k = (s, v), s \in S, substr(w, v) = \mathbf{ja}$$

unter Berücksichtigung des Eingabewortes w beschreiben.

Beispiel:

$$w = aabaa$$
$$k = (s_2, aa)$$

Das heißt, der Präfix aab wurde verarbeitet, wobei Zustandswechsel von s_0 nach s_2 erfolgten und das Suffix aa noch zu verarbeiten ist.

Eine Vorgänger-Nachfolgerbeziehung $\vdash$ zwischen dem Paar k und k' wird über die Relation

$$\vdash \subseteq (S \times \Sigma^*) \times (S \times \Sigma^*)$$

ausgedrückt. Allgemein:

$$k \vdash k' = (s, w) \vdash (s', w')$$

bzw.

$$k \vdash k' = (s, aw) \vdash (s', w)$$

DEA-Konfigurationen (2)

Die reflexiv-transitive Hülle $\vdash^*$ für die Relation $\vdash$ ist wie folgt rekursiv definiert:

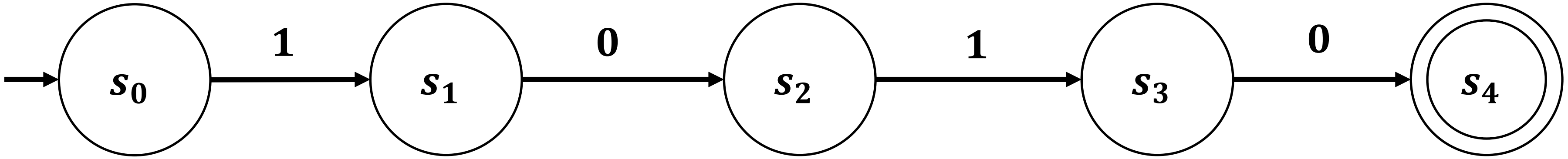
$$\boldsymbol{K} = \boldsymbol{S} \times \boldsymbol{\Sigma}^*$$

$$\forall \boldsymbol{k} \in \boldsymbol{K}: \boldsymbol{k} \vdash^* \boldsymbol{k}' \text{gdw. } \exists \boldsymbol{k}'' \in \boldsymbol{K}: \boldsymbol{k} \vdash^* \boldsymbol{k}'' \wedge \boldsymbol{k}'' \vdash \boldsymbol{k}'$$

d.h.

$$\exists(\boldsymbol{k_1} \vdash \boldsymbol{k_2} \vdash \cdots \boldsymbol{k_n}): \exists \boldsymbol{k_1} \vdash^* \boldsymbol{k_n}$$

Beispiel: Der Automat $\boldsymbol{A} = (\{\mathbf{0}, \mathbf{1}\}, \{\boldsymbol{s_0}, \boldsymbol{s_1}, \boldsymbol{s_2}, \boldsymbol{s_3}, \boldsymbol{s_4}\}, \boldsymbol{\delta}, \boldsymbol{s_0}, \{\boldsymbol{s_4}\})$ folge dem Zustandsdiagramm



$$\boldsymbol{w} = \mathbf{1010}$$

Dann gilt $\quad \boldsymbol{k} \vdash^* \boldsymbol{k}' \equiv \boldsymbol{k} = (\boldsymbol{s_0}, \mathbf{1010}) \vdash (\boldsymbol{s_1}, \mathbf{010}) \vdash (\boldsymbol{s_2}, \mathbf{10}) \vdash (\boldsymbol{s_3}, \mathbf{0}) \vdash (\boldsymbol{s_4}, \boldsymbol{\varepsilon}) = \boldsymbol{k}'$

mit der Startkonfiguration $\boldsymbol{k} = (\boldsymbol{s_0}, \mathbf{1010})$ und der Endkonfiguration $\boldsymbol{k}' = (\boldsymbol{s_4}, \boldsymbol{\varepsilon})$.

Von DEA akzeptierte Sprachen

Akzeptierte Sprachen:

Führen die Elemente $\boldsymbol{w} \in \boldsymbol{L} \subseteq \boldsymbol{\Sigma}^*$ vom Startzustand $\boldsymbol{s_0}$ zu einem akzeptierenden Zustand $\boldsymbol{s_i} \in \boldsymbol{F}$, so gilt die Sprache $\boldsymbol{L}$ als die vom Automaten $\boldsymbol{A}$ akzeptierte Sprache.

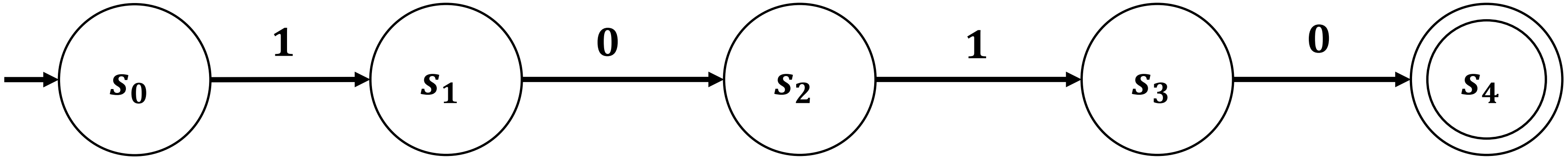
D.h., es gilt $$\boldsymbol{\delta^*(s_0, w) = s_i \in F}$$

bzw. $$\boldsymbol{(s_0, w) \vdash^* (s_i \in F, \varepsilon)}$$

Allgemein gilt also für akzeptierte Sprachen: ein Automat $\boldsymbol{A}$ akzeptiert die Sprache

$$\boldsymbol{L(A) = \{w \in \Sigma^* | (s_0, w) \vdash^* (s_i \in F, \varepsilon)\}}$$

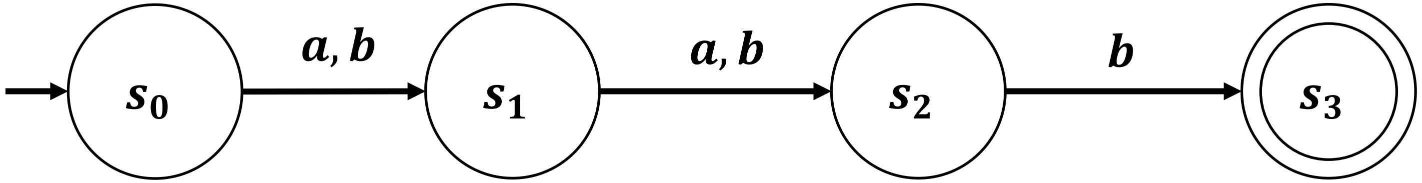
Beispiel: $$\boldsymbol{L(A) = \{1010\}}$$



Von DEA akzeptierte Sprachen – Beispiele (1)

Beispiel 1:

$$L_{3b_1} = \{w \in \Sigma^+ | w = a_1 a_2 a_3 \land a_3 = b\}, \Sigma = \{a, b\}$$



$$A_{3b_1} = (\{a, b\}, \{s_0, s_1, s_2, s_3\}, \delta, s_0, \{s_3\})$$

Alternative Notation der Sprache L_{3b_1} :

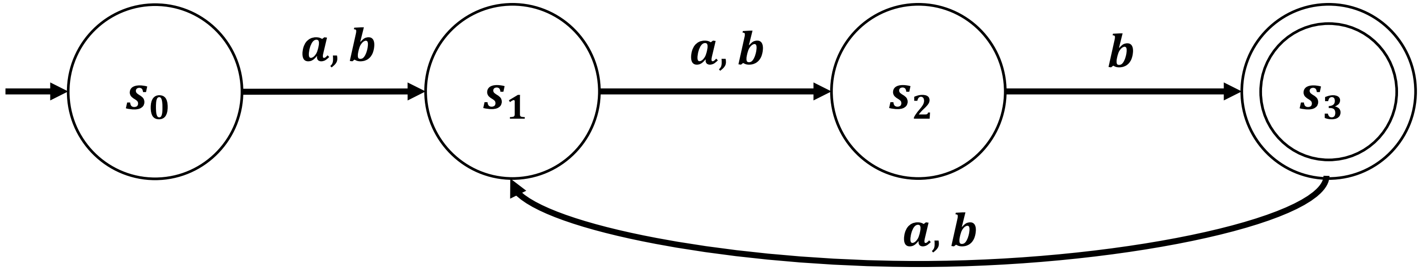
$$L_{3b_1} = \{w \in \{a, b\}^+ | w = xyz, x, y \in \{a, b\}, z = b\}$$

Von DEA akzeptierte Sprachen – Beispiele (2)

Beispiel 2:

$$L_{3b_2} = \left\{ w \in \Sigma^+ \middle| w = a_1 \dots a_{i \bmod 3=0} \wedge a_{j \bmod 3=0} = b, i \geq 3, 1 \leq j \leq i \right\}$$

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



$$A_{3b_2} = (\{a, b\}, \{s_0, s_1, s_2, s_3\}, \delta, s_0, \{s_3\})$$

Alternative Notation der Sprache L_{3b_2}:

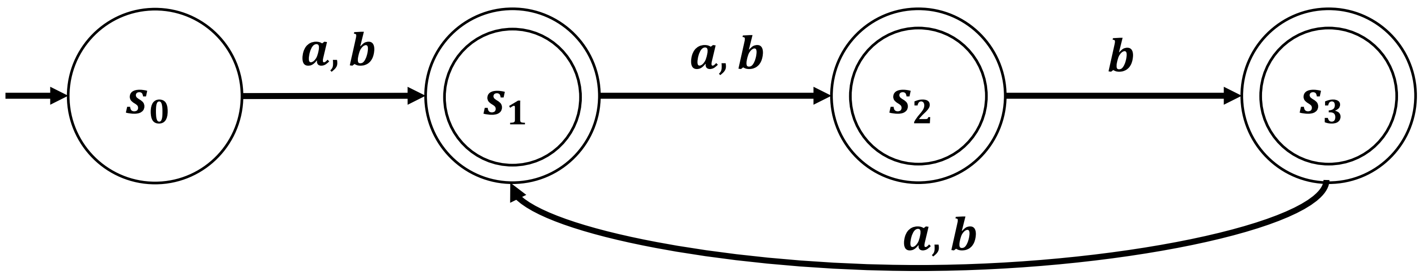
$$L_{3b_2} = \left\{ w \in \{a, b\}^+ | w = \left\{ \{a, b\} \circ \{a, b\} \circ \{b\} \right\}^i, i \geq 1 \right\}$$

Von DEA akzeptierte Sprachen – Beispiele (3)

Beispiel 3:

$$L_{3b_3} = \left\{w \in \Sigma^+ \middle| w = a_1 \ldots a_i \wedge a_{j \bmod 3=0} = b, i \geq 1; 1 \leq j \leq i\right\}$$

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



$$A_{3b_3} = (\{a, b\}, \{s_0, s_1, s_2, s_3\}, \delta, s_0, \{s_1, s_2, s_3\})$$

Vollständige DEA (1)

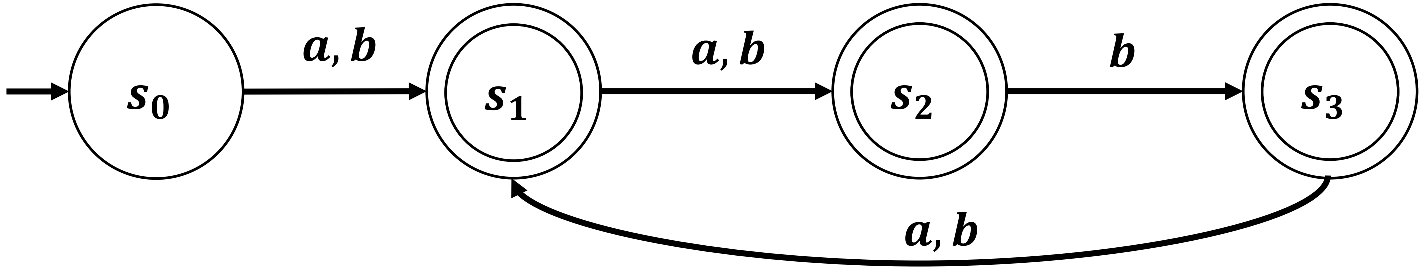
Fragestellung:

Was geschieht, falls im Zustand s_2 ein a eingelesen wird (A_{3b_3})?

$$\delta(s_2, a) = ?$$

D.h., die Zustandsübergangsfunktion ist nicht vollständig definiert.

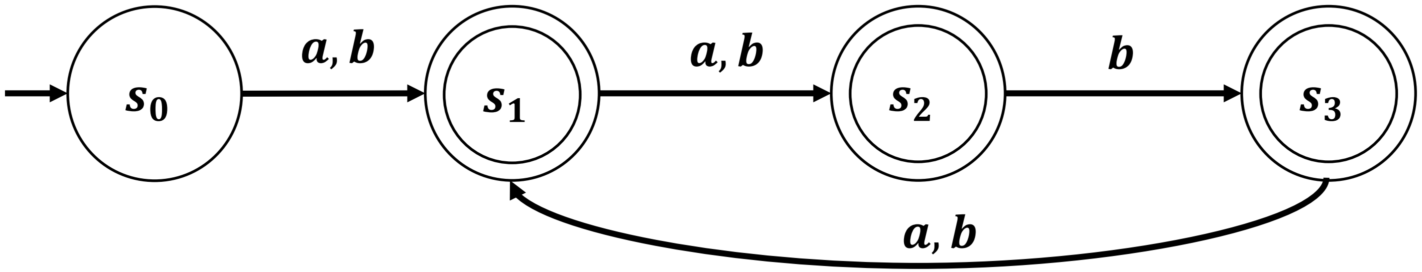
$$Def(\delta) \neq S \times \Sigma$$



Erweiterung des DEA zu einem **vollständigen DEA** durch die Erweiterung des Definitionsbereichs von δ auf das kartesische Produkt $S \times \Sigma$. δ als totale Funktion statt als partielle Funktion.

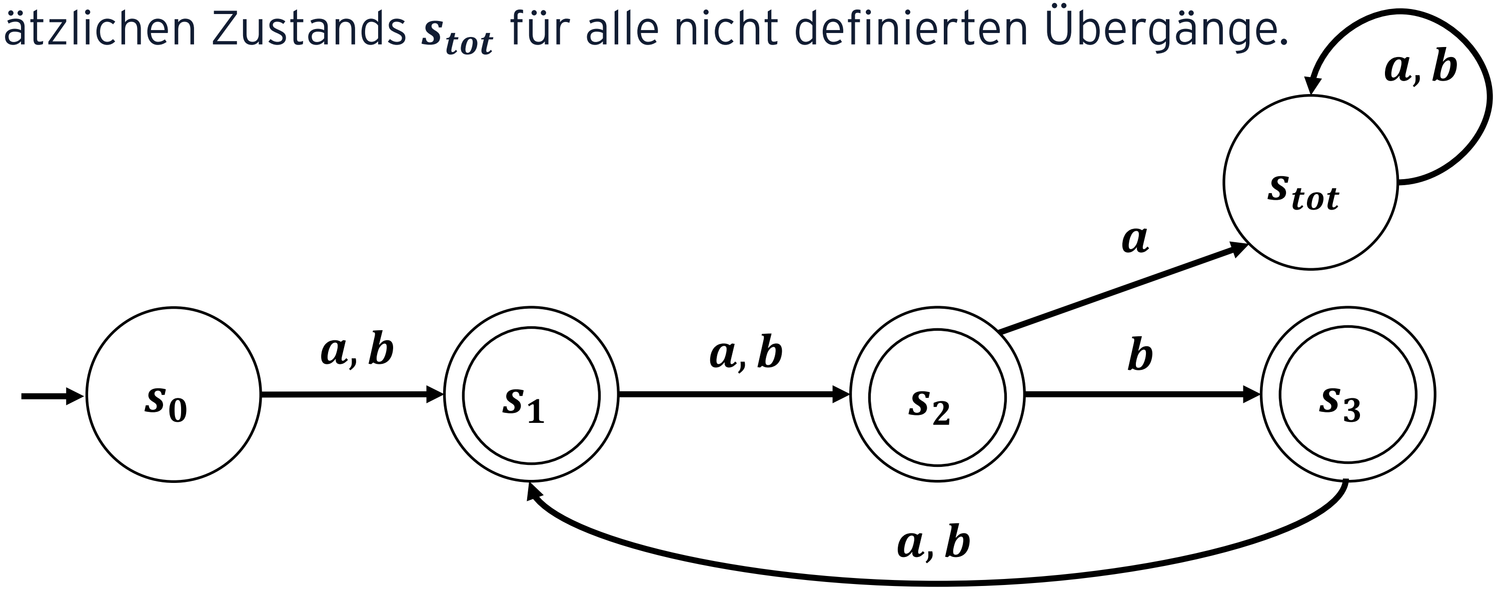
Vollständige DEA (2)

Beispiel $\boldsymbol{A_{3b_3}}$:



Erweiterung:

Einbinden eines zusätzlichen Zustands $\boldsymbol{s_{tot}}$ für alle nicht definierten Übergänge.



Vollständige DEA (3)

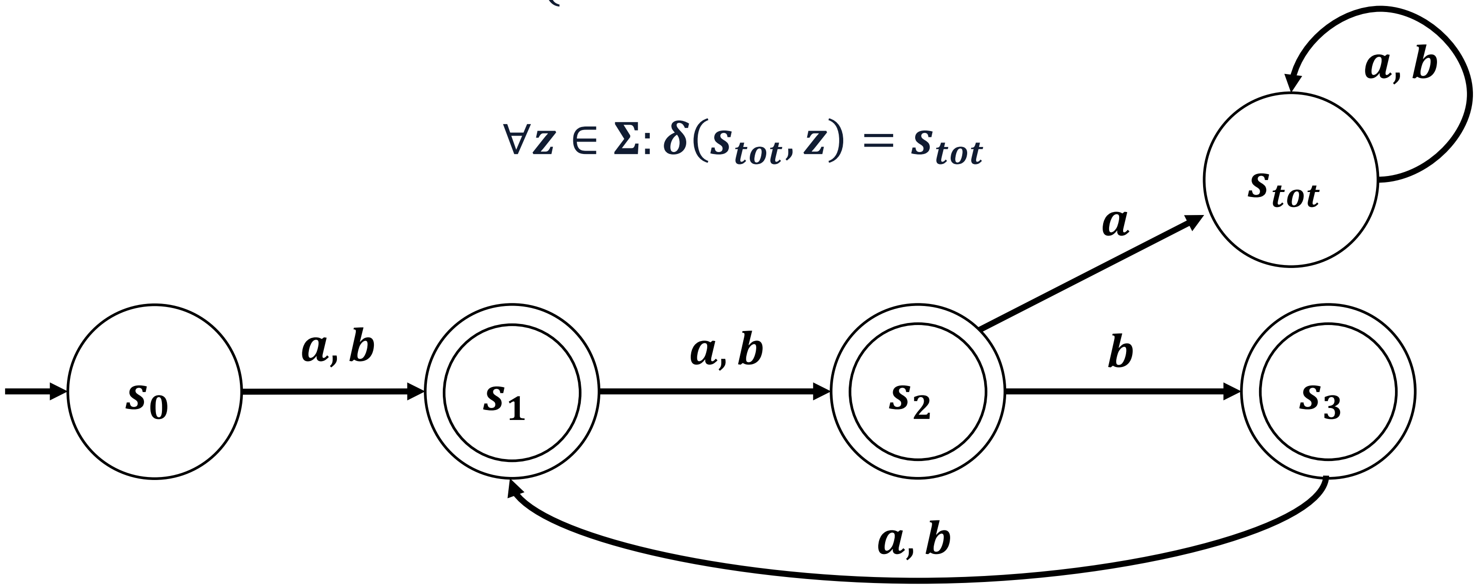
Formal abgebildet wird ein vollständiger Automat demnach wie folgt:

$$A_{3b3,total} = (\Sigma, S \cup \{s_{tot}\}, \delta_{total}, s_0, F), s_{tot} \notin S$$

mit

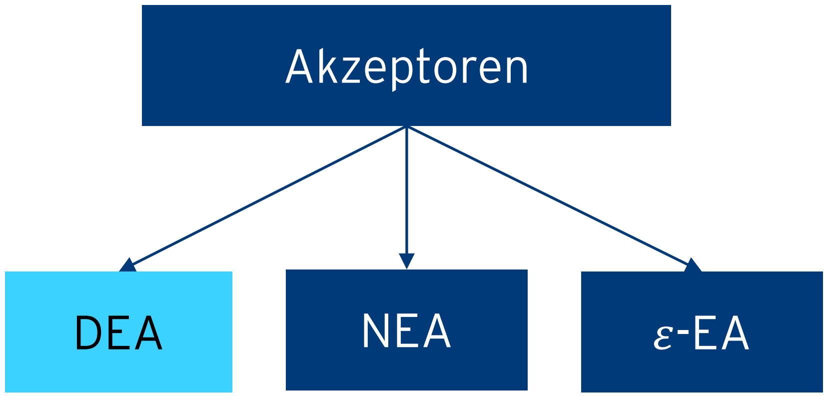
$$\delta_{total} = \begin{cases} \delta(s,z)\ gdw(s,z) \in Def(\delta) \\ s_{tot}, sonst \end{cases}$$

$$\forall z \in \Sigma: \delta(s_{tot}, z) = s_{tot}$$



Aufgaben 1-3

Klasse der von DEA akzeptierten Sprachen



Klassifikation:

Die Klasse der von einem DEA akzeptierten Sprachen über $\boldsymbol{\Sigma}$ wird mit $\boldsymbol{DFA_{\Sigma}}$ notiert (*DFA - Deterministic Finite Automata*).

$$\boldsymbol{DFA_{\Sigma}} = \bigcup_{L \subseteq \Sigma^*, \exists \text{ DEA } A : L = L(A)} \boldsymbol{L}$$

Reguläre Sprachen

Reguläre Sprachen:

Eine Sprache $\boldsymbol{L} \subseteq \boldsymbol{\Sigma}^*$ heißt regulär, falls es einen endlichen Automaten $\boldsymbol{A}$ gibt, der $\boldsymbol{L}$ akzeptiert, d.h.,

$$\boldsymbol{L} = \boldsymbol{L}(\boldsymbol{A})$$

Die Klasse der regulären Sprachen über einem Alphabet $\boldsymbol{\Sigma}$ wird mit $\boldsymbol{REG_\Sigma}$ notiert.

Es gilt:

$$\boldsymbol{REG_\Sigma} = \boldsymbol{DFA_\Sigma}$$

Äquivalenz von Automaten:

$$\boldsymbol{A'} \equiv \boldsymbol{A} \text{ gdw. } \boldsymbol{L}(\boldsymbol{A'}) = \boldsymbol{L}(\boldsymbol{A})$$

Erinnerung: Definition von $\boldsymbol{L}(\boldsymbol{A})$ über Konfigurationsfolgen oder alternativ als

$$\boldsymbol{L}(\boldsymbol{A}) = \{\boldsymbol{w} \in \boldsymbol{\Sigma}^* | \boldsymbol{\delta}^*(\boldsymbol{s_0}, \boldsymbol{w}) \in \boldsymbol{F}\}$$

ausdrücken.

Abgeschlossenheit der Klasse der regulären Sprachen

Sprachen sind abgeschlossen unter den Mengenoperationen.

Für die Menge der regulären Sprachen $\boldsymbol{REG_{\Sigma}}$ gilt:

- Die Menge der regulären Sprachen ist abgeschlossen unter Vereinigung
- Die Menge der regulären Sprachen ist abgeschlossen unter Schnittmengenbildung.
- Die Menge der regulären Sprachen ist abgeschlossen unter Differenzmengenbildung.
- Die Menge der regulären Sprachen ist abgeschlossen unter Konkatenation.
- Die Menge der regulären Sprachen ist abgeschlossen unter Anwendung des Kleene-Sterns.
- Die Menge der regulären Sprachen ist abgeschlossen unter Komplementbildung.
- Die Menge der regulären Sprachen ist abgeschlossen unter Spiegelung (Funktion $\boldsymbol{mirr}: \boldsymbol{P}(\boldsymbol{\Sigma}^*) \rightarrow \boldsymbol{P}(\boldsymbol{\Sigma}^*)$, Teil 2)

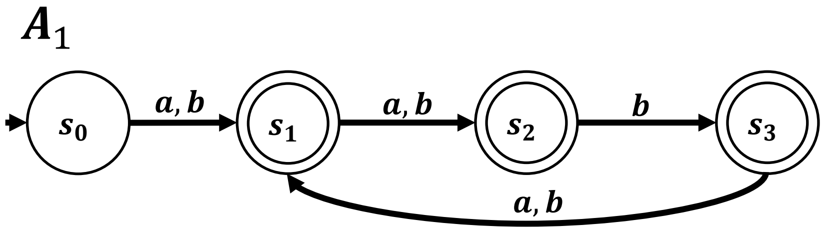
Abgeschlossenheit von unter Vereinigung (1)

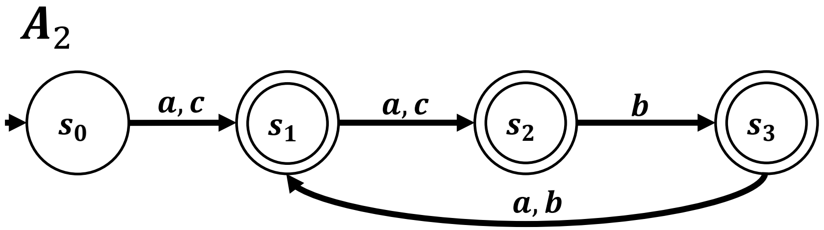
Die Beweise der Abgeschlossenheitseigenschaften können wir in Kürze leichter führen.

Für den Moment als Beweisskizze für die Abgeschlossenheit unter Vereinigung:

Zu $\boldsymbol{L_1}$ und $\boldsymbol{L_2}$ reguläre Sprachen muss es also DEAs $\boldsymbol{A_1}$ und $\boldsymbol{A_2}$ geben mit $\boldsymbol{L(A_1) = L_1}$und $\boldsymbol{L(A_2) = L_2}$.

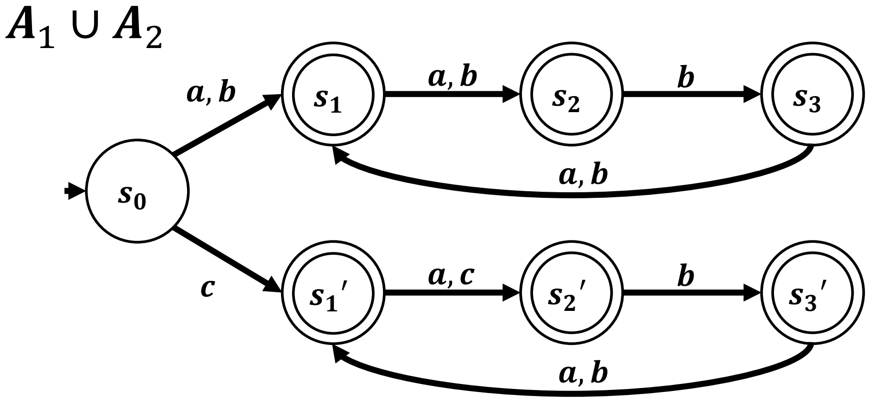
Z.B.





Für den Beweis ist ein DEA $\boldsymbol{A_3}$ zu konstruieren mit $\boldsymbol{L(A_3) = L_1 \cup L_2}$.

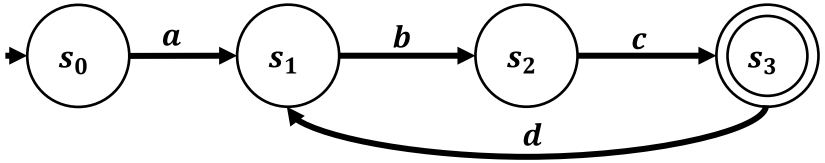
In diesem Beispiel:



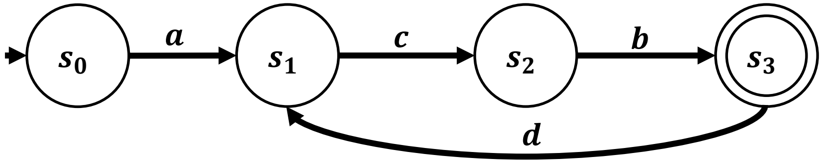
Abgeschlossenheit von unter Vereinigung (2)

Solche Beweise bleiben noch skizzenhaft, da es weitere Fälle zu beachten gibt:

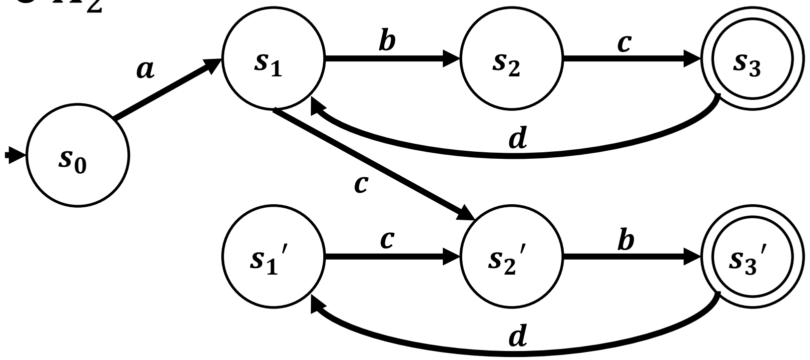
$\boldsymbol{A}_1$



$\boldsymbol{A}_2$



$\boldsymbol{A}_1 \cup \boldsymbol{A}_2$



NORDAKADEMIE
HOCHSCHULE DER WIRTSCHAFT

NORDAKADEMIE gAG Hochschule der Wirtschaft

Köllner Chaussee 11 · 25337 Elmshorn · Tel.: +49 (0) 4121 4090-0 · E-Mail: info@nordakademie.de · Web: www.nordakademie.de