

# Morphologie

- Morphologie ist der Teilbereich der Linguistik, der sich mit der internen Struktur von Wörtern befasst.
- Die wesentlichen Aufgaben der Morphologie in der Computerlinguistik sind
  - die Reduktion komplexer Wörter bzw. Wortformen auf ihre Bestandteile
  - die Identifikation von grammatischer Information, die in der Wortform kodiert ist (z.B. Kasus, Numerus, Tempus)

Anmerkung: Die folgende Übersicht über morphologische Phänomene ist stichwortartig. Mehr Information zur Morphologie gibt es in der Einführungsvorlesung Sprachwissenschaft.

## Teilbereiche der Morphologie

- Flexion:  
Deklination, Konjugation von Substantiven, Verben,  
Adjektiven, Pronomina  
*frag+te+st*  
*ge+frag+t*
- Derivation (Ableitung)  
*Er+kenn+ung*
- Komposition (Zusammensetzung)  
*Sprach+erkennung+s+technik*

# Elemente der morphologischen Struktur

- Stämme, Präfixe, Suffixe (in Flexion und Derivation)
  - Flexionsmorphologie:  
*frag+te+st*  
*ge+frag+t*
  - Derivationsmorphologie:  
*Er+kenn+ung*

# Elemente der morphologischen Struktur

- Stämme, Präfixe, Suffixe (in Flexion und Derivation)
  - Flexionsmorphologie:  
*frag+te+st*  
*ge+frag+t*
  - Derivationsmorphologie:  
*Er+kenn+ung*

# Elemente der morphologischen Struktur

- Stämme, Präfixe, Suffixe (in Flexion und Derivation)

- Flexionsmorphologie:

*frag+te+st*

*ge+frag+t*

- Derivationsmorphologie:

*Er+kenn+ung*

# Elemente der morphologischen Struktur

- Grund- , Bestimmungswörter, Fugenelemente (in der Komposition)

*Sprach+erkennung+s+technik*

*Sprach+erkennung+s+technik*

- Bestimmungswort: *Sprach* +  
Grundwort: *erkennung*
- Bestimmungswort: *Spracherkennung* +  
Fugenelement: *s*  
Grundwort: *technik*

# Elemente der morphologischen Struktur

- Stämme, Präfixe, Suffixe (in Flexion und Derivation)
- Grund- , Bestimmungswörter, Fugenelemente (in der Komposition)
- Modifikation von Stämmen (Umlaut, Ablaut)
  - *Mutter, Mütter*
  - *schwimmen, schwamm, geschwommen*

# Elemente der morphologischen Struktur

- Stämme, Präfixe, Suffixe (in Flexion und Derivation)
- Grund- , Bestimmungswörter, Fugenelemente (in der Komposition)
- Modifikation von Stämmen (Umlaut, Ablaut)
  - *Mutter, Mütter*
  - *schwimmen, schwamm, geschwommen*
- Morpho-phonologische Prozesse



# Morpho-phonologische Prozesse

- Systematische Modifikation, Einfügung und Tilgung von Lauten/Phonemen aus phonetisch-phonologischen Gründen

Einfügung:  $ba\underline{d} + \underline{s}t \rightarrow ba\underline{de}st$

Tilgung:  $ra\underline{s} + \underline{s}t \rightarrow ra\underline{st}$

Modifikation (phonetisch):  $[ba\underline{t}] + [\underline{e}s] \rightarrow [ba\underline{de}s]$

Modifikation (orthografisch, alt):  $na\underline{\beta} + \underline{e} \rightarrow na\underline{sse}$

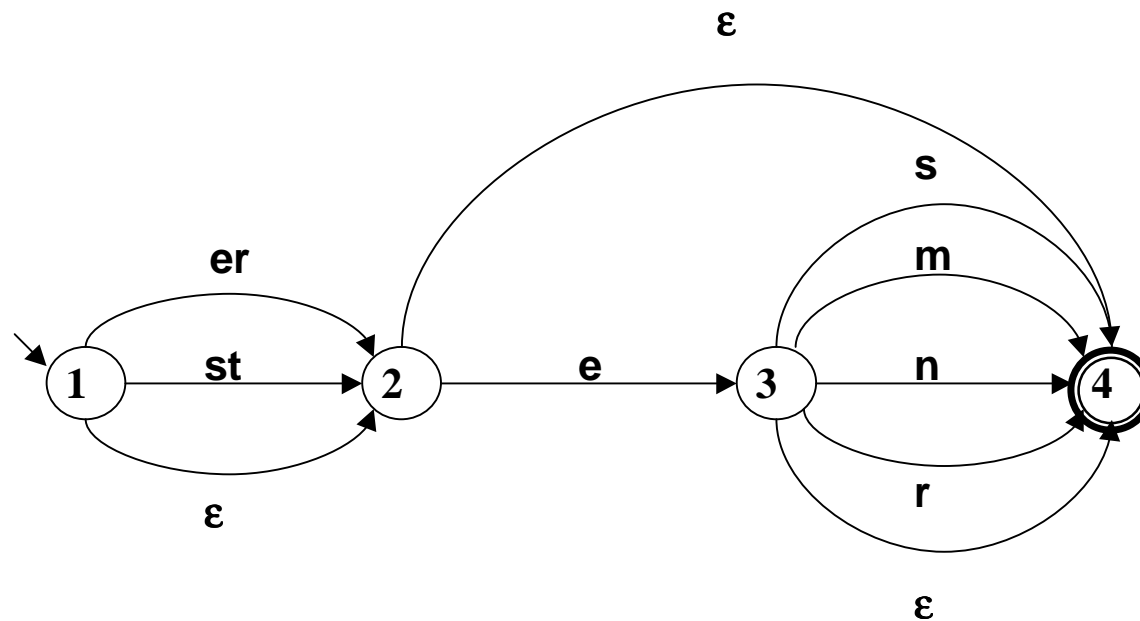
# Morphologische Verarbeitung in der Computerlinguistik

- Systeme zur morphologischen Analyse in der Computerlinguistik arbeiten meist auf der orthografischen Struktur (z.B. für Rechtschreib- und Grammatikkorrektur)
- Teilaufgaben der morphologischen Verarbeitung:
  - Lemmatisierung: Flexionsmorphologische Analyse: Ermittlung des Stammes/Lemmas („stemming“) und ggf. der in den Flexionsformen enthaltenen grammatischen Information
  - Derivativ- und Komposita-Zerlegung: Reduktion komplexer Wörter (Ableitungen und Zusammensetzungen) auf ihre Bestandteile
- Methodisches Werkzeug für alle Aufgaben morphologischer Analyse sind „Endliche Automaten“.
- Wir betrachten die Verwendung endlicher Automaten an der vergleichsweise einfachen Teilaufgabe der Lemma-Ermittlung.

# Adjektivflexion: Paradigma (nur sog. „starke Flexion“)

klein+er	klein+e	klein+es	klein+e
klein+es/en	klein+er	klein+es/en	klein+er
klein+em	klein+er	klein+em	klein+en
klein+en	klein+e	klein+es	klein+e
klein+er+er	klein+er+e	klein+er+es	klein+er+e
klein+er+es/en	klein+er+er	klein+er+es/en	klein+er+er
klein+er+em	klein+er+er	klein+er+em	klein+er+en
klein+er+en	klein+er+e	klein+er+es	klein+er+e
klein+st+ er	klein+st+e	klein+st+ es	klein+st+e
klein+st+es/en	klein+st+er	klein+st+es/en	klein+st+er
klein+st+em	klein+st+er	klein+st+em	klein+st+en
klein+st+en	klein+st+e	klein+st+es	klein+st+e

# Adjektivendungen: Darstellung durch Zustandsdiagramm



# Zustandsdiagramm

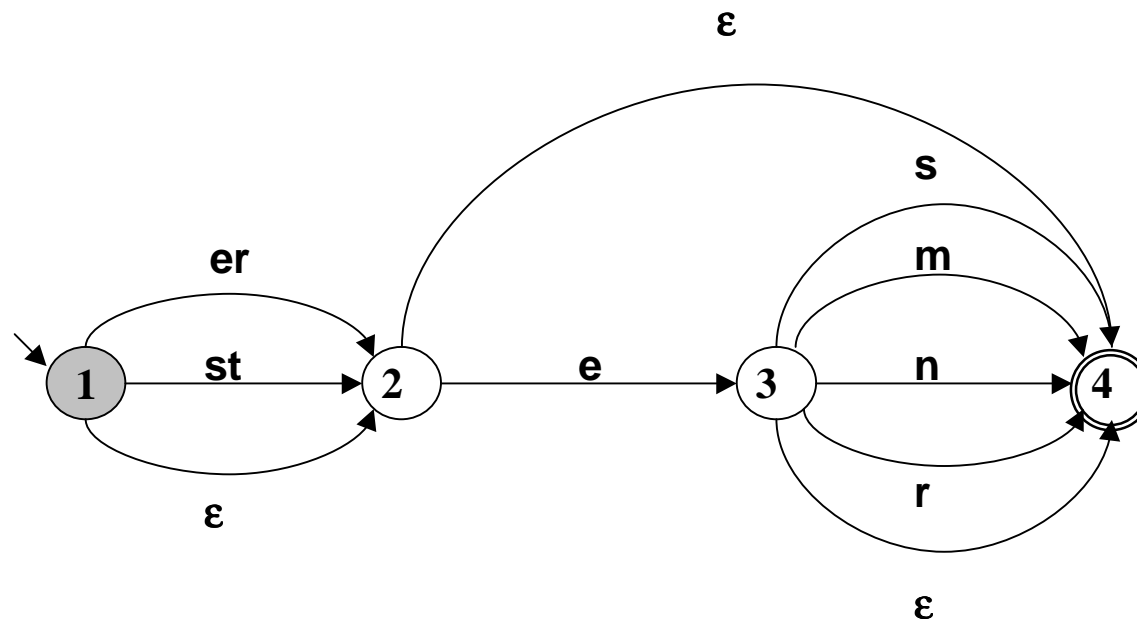
Ein Zustandsdiagramm besteht aus

- Knoten (Zuständen) (im Beispiel: 1,2,3,4)
- davon ein Startknoten (1)
- einem oder mehreren Zielknoten (4)
- Kanten zwischen den Knoten, die
  - gerichtet und
  - beschriftet sind
- Die Kanteninschriften bestehen aus Ketten von Symbolen über einem Alphabet (im Beispiel: e,r,m,n,s,t).
- Auch die leere Kette ( $\epsilon$ ) ist als Kantenbeschriftung zugelassen;  $\epsilon$  ist kein Symbol des Alphabets, sondern bezeichnet den Grenzfall der „aus 0 Symbolen bestehenden“ leeren Kette.

# Die Interpretation des Zustandsdiagramms

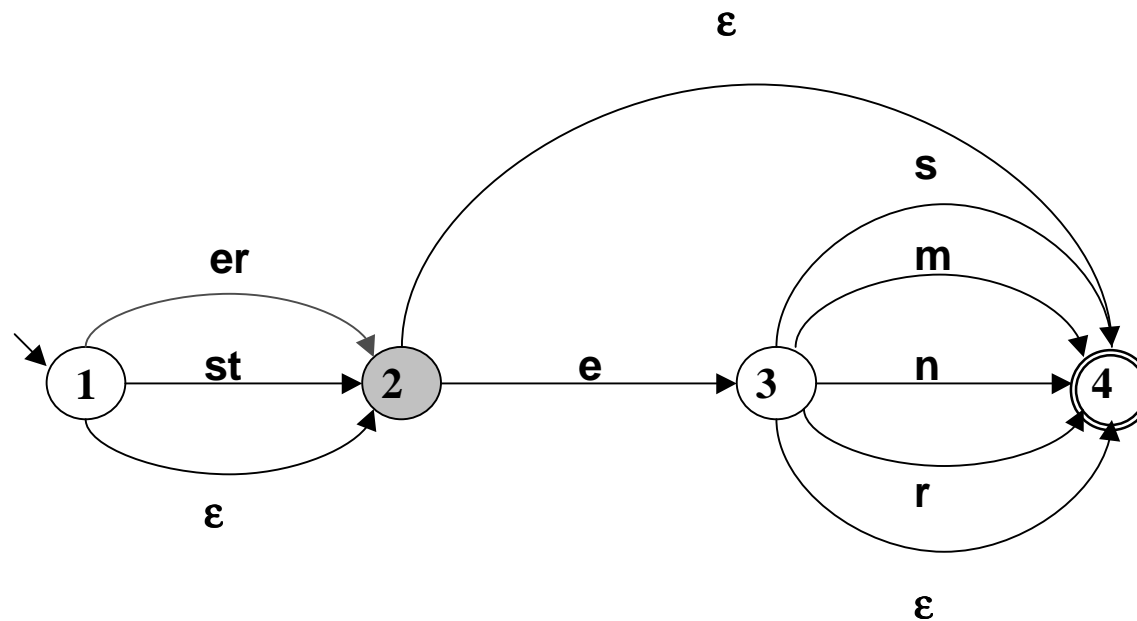
- Das Zustandsdiagramm beschreibt alle Symbolkombinationen oder „Worte“, die sich dadurch ergeben, dass man das Diagramm vom Startknoten zu einem Zielknoten durchläuft und die Inschriften der Kanten, die man dabei beschreitet, aufliest und aneinanderhängt. Man nennt die Menge der Worte, die sich so erzeugen lassen, die vom Diagramm beschriebene „Sprache“.
- Üblicherweise werden Diagramme verwendet, um Eingabeketten zu testen. Man spricht von endlichen Automaten, und sagt, dass ein Automat ein Wort akzeptiert.

# Funktion des Zustandsdiagramms [1]



klein eres

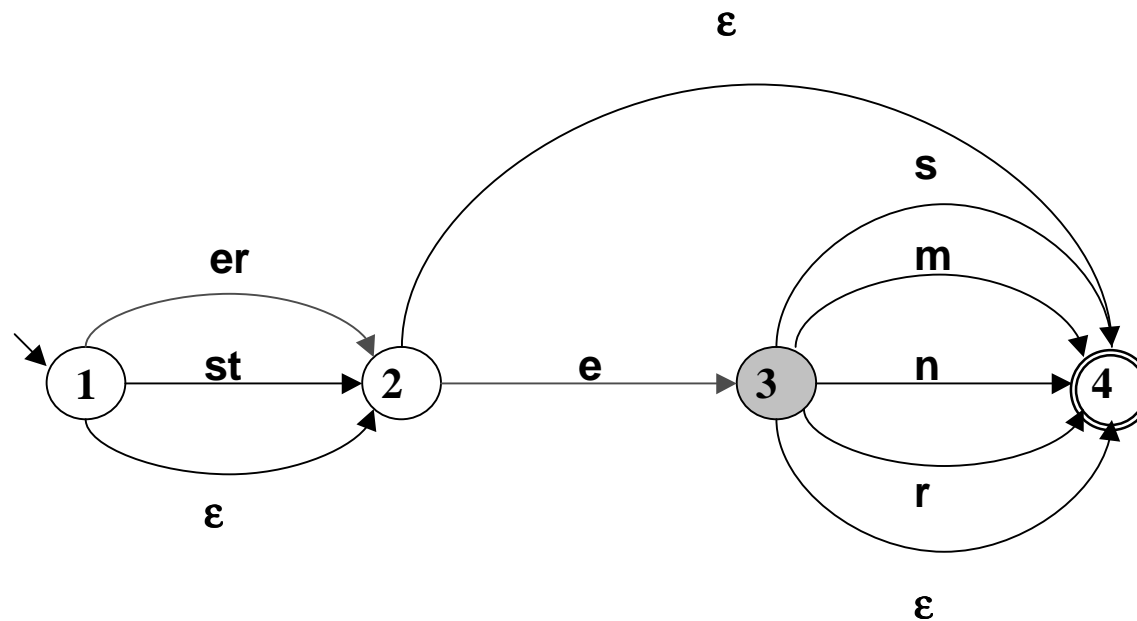
## Funktion des Zustandsdiagramms [2]



klein eres

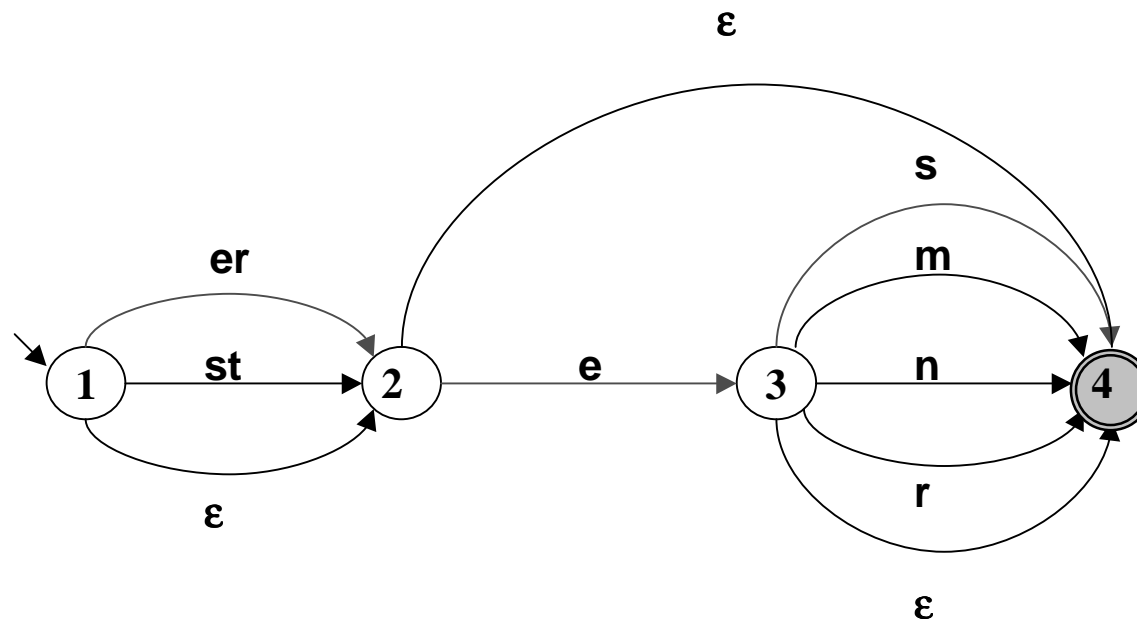


# Funktion des Zustandsdiagramms [3]



klein eres

# Funktion des Zustandsdiagramms [4]



klein eres\_

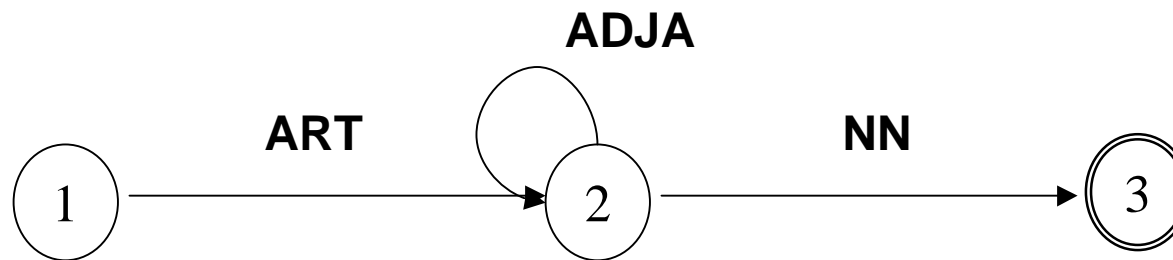
## Zustandsdiagramme: Ein zweites Beispiel [1]

- Wortarten kombinieren sich in bestimmter Weise zu Satzteilen
- Um zu testen, ob eine Wortfolge in einem Dokument eine erlaubte Abfolge von Wortarten darstellt, können Zustandsdiagramme benutzt werden.
- Das folgende Zustandsdiagramm akzeptiert bestimmte einfache Nominalausdrücke, wie

*der Wagen*

*eine interessante Vorlesung*

*das neue schöne rote Dach*



## Zustandsdiagramme: Ein zweites Beispiel [2]

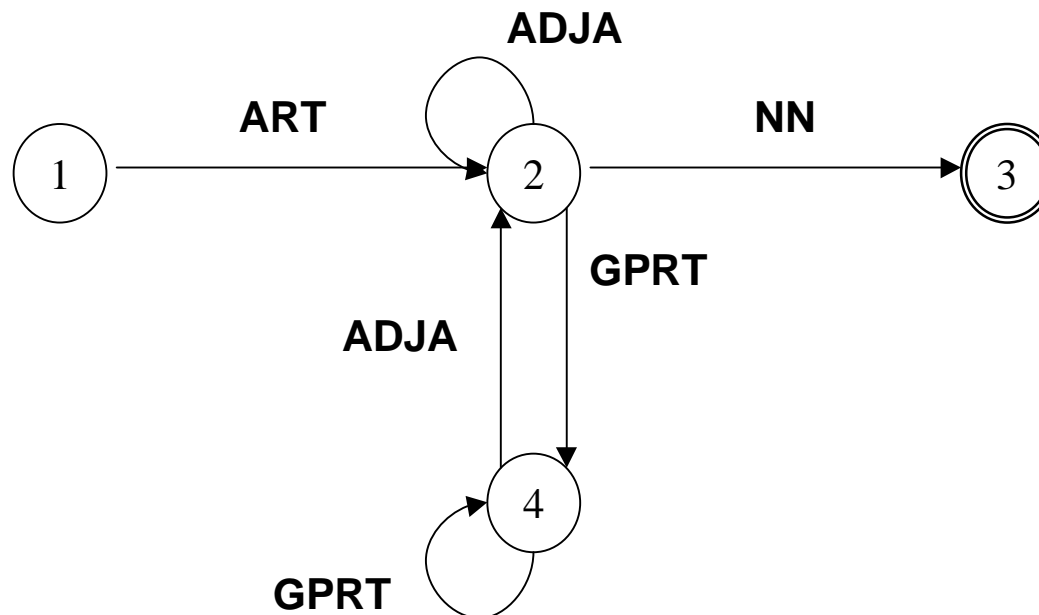
- Das „Alphabet“ des Zustandsdiagramms sind Wortart-Bezeichnungen („Wortart-Tags“ oder auch „POS-Tags“, POS für „part of speech“, engl. für „Wortart“), in unserem Beispiel ART, ADJA, NN
  - ART (Artikel)
  - ADJA (adjektivisches Attribut)
  - NN (Gattungssubstantiv, Gattungsnomen)
- Erkannte „Worte“ sind erlaubte Abfolgen von Wortartsymbolen, z.B. „ART NN“, „ART ADJA NN“, „ART ADJA ADJA ADJA NN“
- Im Gegensatz zum Adjektivendungsdiagramm akzeptiert das Nominalausdrucksdiagramm beliebig lange Worte und beschreibt eine unendliche Sprache. Grund: Es enthält eine Schleife, es ist zyklisch.

## Zustandsdiagramme: Ein zweites Beispiel [3]

Das abgebildete Diagramm akzeptiert auch Adjektive, die mit (einer oder mehreren) Gradpartikeln (GPRT) versehen sind, wie z.B.

*eine ziemlich interessante Vorlesung*

*das recht neue sehr sehr schöne rote Dach*



## Definitionen: Alphabet und Wort

- Ein Alphabet  $\Sigma$  ist eine endliche, nicht-leere Menge von Symbolen.
- Ein Wort  $w$  über dem Alphabet  $\Sigma$  ist eine endliche Kette von Symbolen aus  $\Sigma$ .
- Die Wortlänge  $|w|$  eines Wortes  $w$  ist die Anzahl der verketteten Symbole von  $w$ .
- Das leere Wort  $\varepsilon$  ist das Wort mit Wortlänge 0 ( $|\varepsilon|=0$ ).

## Definitionen: Sprache

- Eine Sprache über dem Alphabet  $\Sigma$  ist eine Menge von Worten über  $\Sigma$ .

Zwei besondere Sprachen:

- Die leere Wortmenge  $\emptyset$  heißt die „leere Sprache“.
- Die maximale Sprache, die die Menge aller Worte über dem Alphabet  $\Sigma$  umfasst, ist  $\Sigma^*$  (der „Stern“ von  $\Sigma$ ).

Anmerkung:

Für jedes Alphabet  $\Sigma$  gilt:  $\varepsilon \in \Sigma^*$ .

# Beispiele

Beispiel 1:

$$\Sigma = \{e, m, n, r, s, t\}$$

$$e, er, rrrrr, mnstmnst, \dots \in \Sigma^*$$

$$L = \{\varepsilon, e, er, em, en, es, ere, erer, erem, eren, eres, st, ste stem, sten, ster, stes\}$$

Beispiel 2:

$$\Sigma = \{\text{ART}, \text{ADJA}, \text{NN}\}$$

$$L = \{\text{ART NN}, \text{ART ADJA NN}, \text{ART ADJA ADJA NN}, \dots\}$$

Alternative Formulierung:

$$L = \{\text{ART ADJA}^n \text{NN}, \dots \mid n \in \mathbf{N}\}$$



## Beispiele

Beispiel 3:

$$\Sigma = \{0,1,2,3,4,5,6,7,8,9\}$$

$$L = \{x_1 \dots x_n y \mid n \in \mathbf{N}, x_i \in \Sigma \text{ für } 1 \leq i \leq n, y \in \{0,5\}, n \in \mathbf{N}\}$$

(die Menge der durch 5 teilbaren natürlichen Zahlen, wenn wir Ziffernfolgen mit 0-Präfixen ebenfalls zulassen)

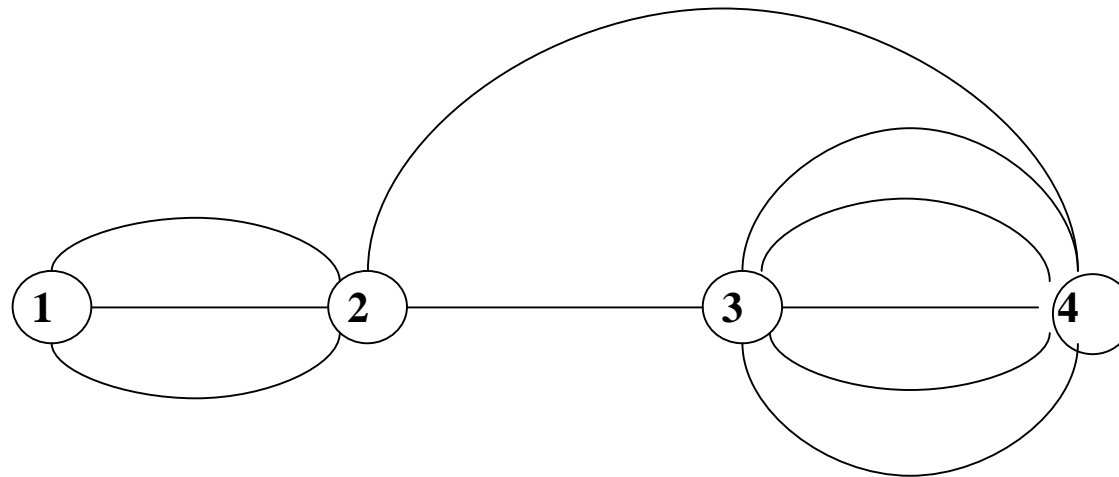
Bemerkungen:

- Mit  $\mathbf{N}$  bezeichnen wir hier die Menge der natürlichen Zahlen inklusive 0.
- $a^n$  ist die Kette, die durch n-faches Hintereinanderschreiben des Symbols  $a$  entsteht (für  $n=0$  ist  $a^n = \varepsilon$ )

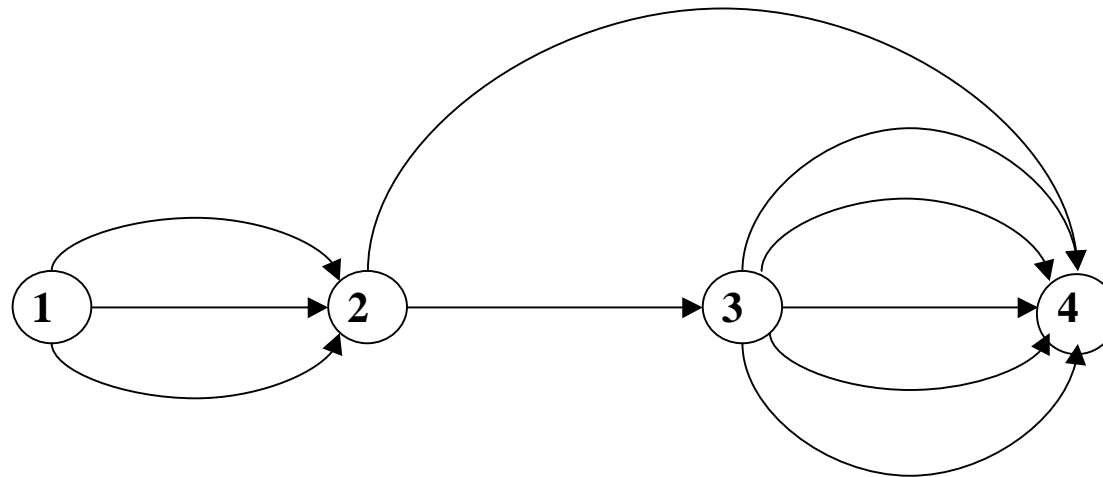
## Definition: Zustandsdiagramm [1]

Ein Zustandsdiagramm ist ein gerichteter Graph mit Kanteninschriften.

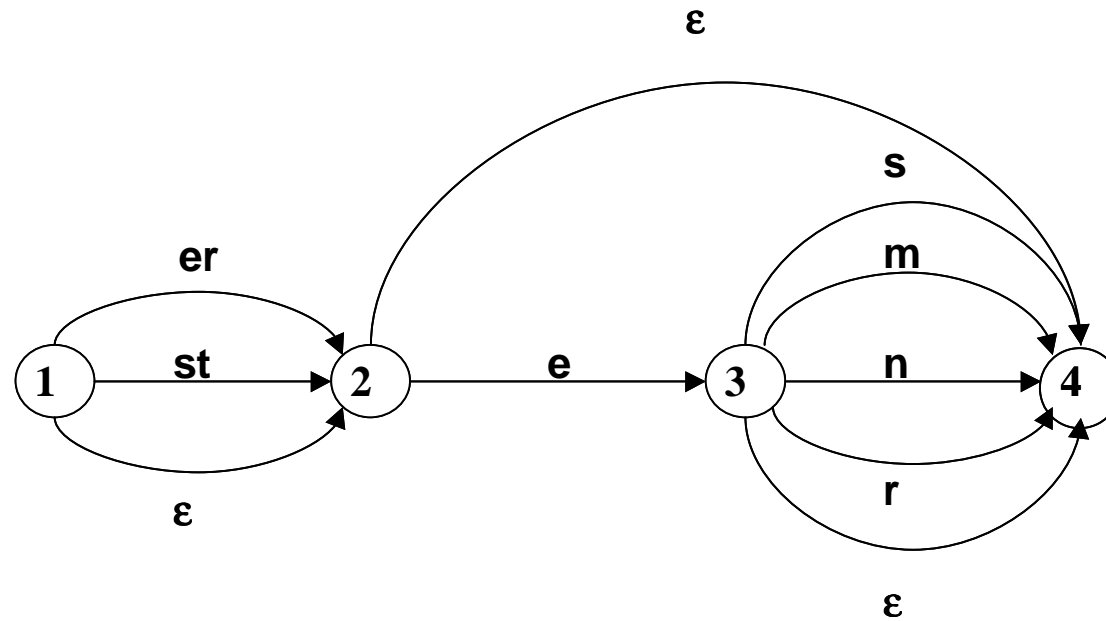
# Ein Graph



# Ein gerichteter Graph



# Ein gerichteter Graph mit Kanteninschriften



## Definition: Zustandsdiagramm [2]

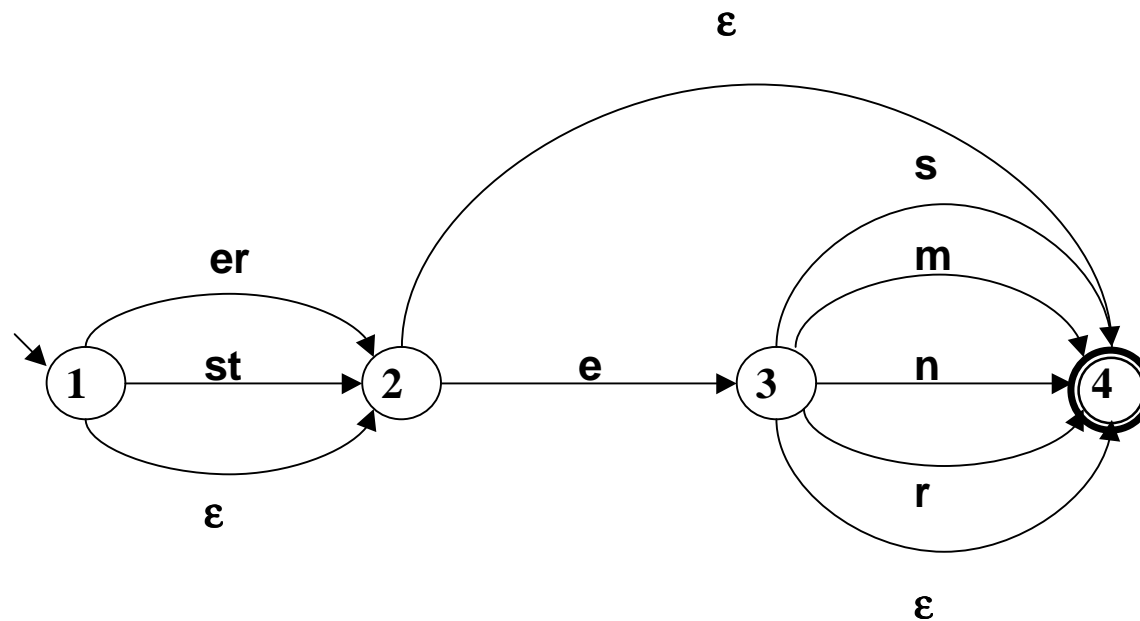
Formal wird ein Zustandsdiagramm definiert als ein Quintupel (Folge von 5 Elementen)

$A = \langle K, \Sigma, \Delta, s, F \rangle$ , wobei

- $K$  nicht-leere endliche Menge von Knoten (Zuständen)
- $\Sigma$  nicht-leeres Alphabet
- $s \in K$  Startzustand
- $F \subseteq K$  Menge von Endzuständen
- $\Delta : K \times \Sigma^* \times K$  Menge von beschrifteten Kanten (Übergangsrelation)

Anmerkung: Das Zustandsdiagramm heißt auch „nicht-deterministischer endlicher Automat“ (NEA), engl.: „non-deterministic finite-state automaton“ (NFA) (Erklärung später)

# Adjektivendungen: Zustandsdiagramm



## Beispiel: Das Adjektivendungs-Diagramm

NEA  $A = \langle K, \Sigma, \Delta, s, F \rangle$  mit

- $K = \{1, 2, 3, 4\}$  (Zustände)
- $\Sigma = \{e, m, n, r, s, t\}$  (Alphabet)
- $s = 1$  (Startzustand)
- $F = \{4\}$  (einziger Endzustand)
- $\Delta = \{ \langle 1, er, 2 \rangle, \langle 1, st, 2 \rangle, \langle 1, \varepsilon, 2 \rangle, \langle 2, e, 3 \rangle, \langle 2, \varepsilon, 4 \rangle, \dots \}$  (Übergangsrelation)



## Durch NEA akzeptiertes Wort/definierte Sprache

- Ein Wort  $w \in \Sigma^*$  wird durch den NEA  $A = \langle K, \Sigma, \Delta, s, F \rangle$  akzeptiert  
gdw. es eine Folge von Kanten (einen Pfad durch den NEA)  $\langle s, u_1, k_1 \rangle, \langle k_1, u_2, k_2 \rangle, \dots, \langle k_{n-1}, u_n, k_n \rangle$  gibt, sodass  $k_n \in F$  und  $u_1 u_2 \dots u_n = w$  (die Konkatenation, das Aneinanderhängen der Inschriften der durchlaufenen Kanten ergibt das Wort  $w$ ).
- Die vom NEA  $A = \langle K, \Sigma, \Delta, s, F \rangle$  definierte (akzeptierte) Sprache  $L(A)$  ist die Menge der von  $A$  akzeptierten Worte.

## Eine allgemeine methodische Bemerkung

- Die Definition des Zustandsdiagramms/NEA spezifiziert eine formale Notation, die für sich genommen keine Bedeutung hat.
- Durch die Definitionen der letzten Folie (akzeptiertes Wort/definierte Sprache) wird diese Datenstruktur interpretiert: Wir verwenden Zustandsdiagramme, um die Zugehörigkeit von Symbolketten zu Sprachen zu definieren und zu testen.
- Hinzu kommen muss ein handhabbares Verfahren, ein Algorithmus, um den Zugehörigkeitstest tatsächlich durchzuführen.

Diese drei Schritte sind für die Modellierung in der Computerlinguistik (und der Informatik) charakteristisch.