# Anhang A: SIMPLE EXMARaLDA-Konventionen

1. Jede Zeile beginnt mit der Sigle des Sprechers der Äußerung, gefolgt von einem Doppelpunkt. Zwei Sprecher dürfen sich nicht dieselbe Sigle teilen, und die Groß- und Kleinschreibung in den Sprechersiglen ist relevant (d. h., dass beispielsweise „Tom“ und „TOM“ als zwei verschiedene Sprechersiglen behandelt werden):

TOM: .....

TIM: .....

1. Pro Zeile wird eine Äußerung transkribiert. Jede Zeile wird mit einem Zeilenendezeichen (carriage return) beendet. Leerzeilen zur übersichtlicheren Strukturierung sind zulässig.

TOM: Hallo, Tim!

TIM: Hallo, Tom.

1. Eine etwaige Transkription von non-verbalen Handlungen, die die Äußerung begleiten (d. h. parallel zu ihr stattfinden), kann der Äußerung in eckigen Klammern vorangestellt werden.

TOM: [winkt] Hallo, Tim!

TIM: [winkt] Hallo, Tom.

1. Eine etwaige Annotation der Äußerung (z. B. eine Übersetzung) kann der Äußerung in geschweiften Klammern nachgestellt werden. Dies geschieht in der gleichen Zeile, in der auch die dazugehörige Äußerung steht.

TOM: [winkt] Hallo, Tim! {Salut, Tim!}

TIM: [winkt] Hallo, Tom. {Salut, Tom!}

1. Sich überlappende Äußerungsteile verschiedener Sprecher werden mit spitzen Klammern eingefasst. Der schließenden spitzen Klammer folgt eine beliebige Zeichenkette, die die Überlappung indiziert, und eine weitere schließende spitze Klammer. Die Indizierung sollte aus Gründen der Lesbarkeit durch Zahlen erfolgen, diese müssen jedoch nicht aufsteigend geordnet sein (notwendig ist nur, dass sie eindeutig sind). Für eine bessere Lesbarkeit, können die sich überlappenden Äußerungsteile mittels Leerzeichen oder Tabulatoren eingerückt werden.

TOM: [winkt] Hallo, <Tim!>1> {Salut, Tim!}

TIM: [winkt] <Hallo>1>, Tom. {Salut, Tom!}

1. Eckige, geschweifte und spitze Klammern dürfen nur in der oben definierten Art und Weise verwendet werden und sollten sonst in der Transkription nicht vorkommen

# Anhang B: Segmentierungs-Algorithmen

## Allgemeines zum Segmentieren

### Einführung

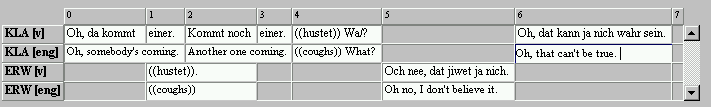
Das Segmentieren ist ein Arbeitsschritt, der in der Regel nach der Fertigstellung der Transkription auf dieselbe angewendet wird. Das Segmentieren kann vielfältigen Zwecken dienen, die sich – allgemein gesprochen – unter den folgenden beiden Schlagworten zusammenfassen lassen:

* Die automatische Generierung zusätzlicher Darstellungsformen für eine Transkription, beispielsweise die Ausgabe einer Transkription als Äußerungsliste (vgl. z. B. "Transcription > Transformation" oder "File > Output > GAT transcript").
* Das Unterteilen der Transkription in relevante (sprachliche) Analyseeinheiten, die bei einer computergestützten Auswertung einer Transkription oder eines Transkriptionskorpus zur Anwendung kommen. Beispielsweise setzt das Analyseinstrument „Alphabetische Wortliste“ (vgl. z. B. Transcription > Word list“) eine Segmentierung der Transkription in Wörter voraus, und auch das einfache Zählen von Einheiten (vgl. z. B. „Transcription > Count segments“) erfordert, dass die zu zählenden Einheiten vorher segmentiert wurden.

Diese Einführung erläutert die allgemeine Funktionsweise des Segmentierens. Für eine detaillierte Beschreibung einzelner Menüpunkte sei auf die betreffenden Stellen in der Funktions-Referenz verwiesen.

### Was wird segmentiert?

Zunächst ist festzuhalten, dass zu segmentierendes Material sich in aller Regel nur in Spuren des Typs T(ranscription) befindet, also dort, wo normalerweise das verbale Handeln eines Sprechers nach orthographischer oder literarischer Umschrift beschrieben wird. Spuren des Typs D(escription), also z. B. Spuren in denen non-verbale Handlungen, Gestik oder Mimik festgehalten wird, werden in der Regel ebenso wenig segmentiert wie Spuren des Typs A(nnotation), in denen sich annotierende Elemente wie Übersetzungen, Kommentare etc. befinden. So sind in dem folgenden Transkriptionsausschnitt nur die erste und dritte Spur Gegenstand der Segmentierung, während die zweite und vierte Spur (die als Übersetzungen jeweils den Typ „A“ haben) nicht berücksichtigt werden:



Innerhalb einer Spur des Typs „T“ erfolgt die Segmentierung Schritt für Schritt, anhand der Einheit der Segmentkette („segment chain“). Eine solche Segmentkette ist definiert als eine zeitlich ununterbrochene Folge von Ereignissen. Im Editor lassen sich solche Segmentketten gut erkennen: Sie sind die (standardmäßig) weiß unterlegten Abschnitte zwischen zwei (standardmäßig) grau unterlegten Abschnitten. So enthält das obige Beispiel genau vier Segmentketten:

**KLA:** Oh, da kommt einer. Kommt noch einer. ((hustet)) Wa/?

**ERW:** ((hustet)).

**ERW:** Och nee, dat jiwet ja nich.

**KLA:** Oh, dat kann ja nich wahr sein.

### Wie wird segmentiert?

Die eigentliche Segmentierung erfolgt, indem auf die Segmentketten der zu segmentierenden Spuren eine sog. Endliche Maschine („Finite State Machine“) angewandt wird. Diese ist ein einfacher Algorithmus, der Äußerungsendzeichen, Worttrenner etc. „erkennt“ und anhand dieser Informationen Segmentketten in kleinere Einheiten zerlegt. Da sich die Benennung und Bedeutung solcher Einheiten von Transkriptionssystem zu Transkriptionssystem unterscheidet (z. B. Äußerung in HIAT vs. Phrasierungseinheit in GAT) und jedes Transkriptionssystem andere Endzeichen für seine Einheiten verwendet (z. B. die fünf Äußerungsendzeichen in HIAT vs. fünf Zeichen für eine abschließende Tonhöhenbewegung in GAT), enthält der Partitur-Editor verschiedene Endliche Maschinen für verschiedene Transkriptionssysteme – welche der Partitur-Editor anwendet, stellen sie über "Edit > Preferences..." ein.

Die Endliche Maschine nutzt also die Regelmäßigkeiten der einzelnen Transkriptionssysteme, um Segmentketten in kleinere Einheiten zu zerlegen. So kann z. B. in dem gegebenen Beispiel über die HIAT-Segmentierung anhand der verwendeten Punkte und Fragezeichen (die gemäß HIAT eine Äußerung abschließen) eine Unterteilung der Segmentketten in Äußerungen erfolgen:

**KLA:** Oh, da kommt einer**.**

**ERW:** ((hustet))**.**

**KLA:** Kommt noch einer**.**

**KLA:** ((hustet)) Wa/**?**

**ERW:** Och nee, dat jiwet ja nich**.**

**KLA:** Oh, dat kann ja nich wahr sein**.**

In gleicher Weise wird anhand der doppelten runden Klammern erkannt, dass die eingeschlossenen Zeichenketten nicht-phonologisches Material beschreiben:

**KLA:** ...

**ERW:** **((**hustet**))**.

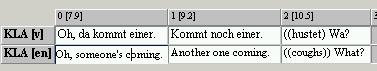
**KLA:** ...

**KLA:** **((**hustet**))** Wa/?

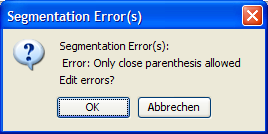
### Fehlerursachen beim Segmentieren

Weil der Segmentierungsalgorithmus auf den Regelmäßigkeiten der Transkriptionssysteme beruht, kann es zu Problemen beim Segmentieren kommen, wenn beim Transkribieren gegen diese Regelmäßigkeiten verstoßen wird – d. h. gewisse Transkriptionszeichen nicht gemäß der Konvention eingesetzt werden.

So ist im folgenden Beispiel die nicht-phonologische Einheit „hustet“, deren Beginn in HIAT durch ein Paar öffnender runder Klammer signalisiert wird, nicht konventionsgemäß, d. h. nicht durch ein korrespondierendes Paar schließender Klammern, beendet worden:



Bei Menüpunkten, die eine Segmentierung voraussetzen (z.B. Transcription > Count segments...) erscheint daher folgende Fehlermeldung:



Diese enthält eine Angabe zur Ursache des Fehlers – „Only close parenthesis allowed“, bedeutet, dass an der betreffenden Stelle nur eine schließende (Doppel-)Klammer stehen darf – und bietet Ihnen an, alle Segmentierungsfehler in einem eigenen Dialog (siehe "Transcription > Segmentation Errors...") zu bearbeiten.

## 

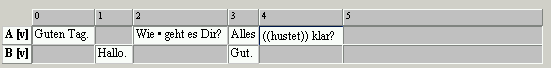
## Segmentierung: „HIAT: Utterance and Words“

Alle Zeichen, die nicht in der folgenden Tabelle aufgeführt sind, werden im EXMARaLDA Partitur-Editor als Bestandteile von Wörtern behandelt (sofern sie nicht Teil eines nicht-phonologischen Datums sind).

|  |  |  |
| --- | --- | --- |
| **Name** | **Standard-Werte** | **Erläuterung** |
| UtteranceEndSymbols | **. | ! | ? | … | ˙ |** | Markieren das Ende einer Äußerung (optional gefolgt von einem Leerzeichen). |
| SpaceSymbols | **|** | Markiert das Ende eines Wortes oder folgt auf „Äußerungsendzeichen“ bzw. auf „übrige Interpunktion“. |
| Quote | **" |** | Markieren den Beginn und das Ende von Redewiedergaben. Äußerungsendzeichen innerhalb von Redewiedergaben werden ignoriert. |
| OpenParenthesis | **( |** | Doppeltes Auftreten markiert den Beginn eines nicht-phonologischen Segments. Einfaches Auftreten wird wie „übrige Interpunktion“ behandelt. |
| CloseParenthesis | **) |** | Doppeltes Auftreten markiert das Ende eines nicht-phonologischen Segments. Einfaches Auftreten wird wie „übrige Interpunk­tion“ behandelt. |
| MiscellaneousPunctuation | **' | : | ; | , | - | \_ | ‿ | — | / |** | Markieren intrasegmentale Phänomene und werden als Interpunktion segmentiert. |
| PauseSymbols | **• | · |** | Markiert Pausen und wird als nicht-phonologisches Datum segmentiert. |

Beispiel:

Die zweite Segmentkette von Sprecher A wird mit der Segmentierung: „HIAT: Utterance and Words“ ...



... folgendermaßen in Äußerungen, Wörter (W), Interpunktion (IP) und nicht-phonologische Segmente (Non-pho) segmentiert:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Segmentkette | | | | | | | | | | | | | | | |
| Äußerung | | | | | | | | | | Äußerung | | | | | |
| **W** | **IP** | **Non-Pho** | **IP** | **W** | **IP** | **W** | **IP** | **W** | **IP** | **W** | **IP** | **Non-Pho** | **IP** | **W** | **IP** |
| Wie |  | • |  | geht |  | es |  | Dir | ? | Alles | (( | hustet | )) | klar | ? |

| Mögliche Fehler | | |
| --- | --- | --- |
| Ursache | Beispiel | Fehlermeldung |
| Sprecherbeitrag beginnt mit schließender Klammer. | **)** Ich mache eine Äußerung | Error: No parentheses closing, No utterance end symbol, No end of input allowed |
| Sprecherbeitrag beginnt mit Äußerungsendzeichen. | **!** Ich mache eine Äußerung | Error: No parentheses closing, No utterance end symbol, No end of input allowed |
| Schließende Klammer folgt auf Äußerungsendzeichen. | Ich mache eine (Äußerung.**)** | Error: No closed parenthesis, no further utterance end symbol allowed |
| Äußerungsendzeichen folgt auf Äußerungsendzeichen. | Ich mache eine Äußerung.**!** | Error: No closed parenthesis, no further utterance end symbol allowed |
| Schließende Klammer folgt auf Äußerungsendzeichen und Leerzeichen. | Ich mache eine (Äußerung. **)** | Error: No close parenthesis, no utterance end symbol allowed |
| Äußerungsendzeichen folgt auf Äußerungsendzeichen und Leerzeichen. | Ich mache eine Äußerung. **!** | Error: No close parenthesis, no utterance end symbol allowed |
| Pausensymbol innerhalb von doppelten runden Klammern. | Ich ((geht • zur Tür)) hau ab. | Error: No end of input, no open parenthesis, no pause symbol allowed |
| Dritte öffnende runde Klammer. | Ich ((geht **(**oder rennt) zur Tür)) hau ab. | Error: No end of input, no open parenthesis, no pause symbol allowed |
| Sprecherbeitrag endet, ohne dass doppelte runde Klammern geschlossen wurden. | Ich hab Husten ((hustet. | Error: No end of input, no open parenthesis, no pause symbol allowed |
| Nach doppelter geöffneter Klammer folgt auf eine schließende Klammer etwas anderes als eine weitere schließende Klammer. | Ich ((hustet)**m**ache eine Äußerung. | Error: Only close parenthesis allowed |
| Innerhalb eines Wortes beginnt eine Redewiedergabe. | Ich mach**"**e eine Äußerung." | Error: No opening quote allowed |
| Sprecherbeitrag endet, ohne dass eine begonnene Redewiedergabe durch ein zweites Anführungszeichen abgeschlossen wurde. | Und er sagt: "Ich mache eine Äußerung | Error: No end of input allowed |

## Segmentierung: „DIDA: Words“

Alle Zeichen, die nicht in der folgenden Tabelle aufgeführt sind, werden im EXMARaLDA Partitur-Editor als Bestandteile von Wörtern behandelt (sofern sie nicht Teil eines nicht-phonologischen Datums sind).

|  |  |  |
| --- | --- | --- |
| **Name** | **Standard-Werte** | **Erläuterung** |
| CAPITALS | **A** | **B** | **C** | **D** | **E** | **F** | **G** | **H** | **I** | **J** | **K** | **L** | **M** | **N** | **O** | **P** | **Q** | **R** | **S** | **T** | **U** | **V** | **W** | **X** | **Y** | **Z** | **Ä** | **Ö** | **Ü** | | Nur zur Beschreibung nicht-morphemisierter Äußerungen, nicht als Bestandteil von Wörtern. |
| PLUS | **+** | | Markiert einen schnellen Anschluss am Beginn einer Segmentkette. |
| NUMBERS\_AND\_COMMA | **0** | **1** | **2** | **3** | **4** |  **5** | **6** | **7** | **8** | **9** | **,** | | Nur als Bestandteile von Zeitangaben (als Teil einer Pause oder einer unverständlichen Passage). |
| PAUSE | **\*** | | Markiert Beginn und Ende von Pausen. |
| WORD\_SEPARATORS | **→** | **↟**| **↑** | **↓** | **<** | **>** | **-** | **/** | | Zeichen für suprasegmentale Phänomene, sind nicht Bestandteile von Wörtern. |
| EQUALS\_SIGN | **=** | | Markiert eine Verschleifung oder steht innerhalb einer nicht-morphemisierten Äußerung. |
| SPACE | | | Leerzeichen, kann an verschiedenen Stellen auftreten, markiert oft den Anfang oder das Ende eines Segmentes. |
| OPEN\_PARENTHESIS | **(** | | Markiert den Beginn einer unverständlichen oder schwer verständlichen Passage. |
| CLOSE\_PARENTHESIS | **)** | | Markiert das Ende einer unverständlichen oder schwer verständlichen Passage. |
| OPEN\_SQUARE\_BRACKET | **[** | | Markiert den Beginn einer Auslassung. |
| CLOSE\_SQUARE\_BRACKET | **]** | | Markiert das Ende einer Auslassung. |
| AMPERSAND | **&** | | Doppeltes Auftreten markiert einen Referenzbereich in einer Sprecherzeile. |
| PERIOD\_OR\_ELLIPSIS | **.** | | **…** | | Nur innerhalb unverständlicher Passagen. |
| COLON | **:** | | Entweder innerhalb eines Wortes (als Dehnungszeichen) oder innerhalb einer Zeitangabe. |

Beispiel:

Die folgende Segmentkette von Sprecher X wird mit der Segmentierung: „DIDA: Words“ ...



... folgendermaßen in Wörter (W), Interpunktionszeichen (IP), nicht-morphemisierte Äußerungen (NMÄ) und Pausen (PAUSE) segmentiert:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Segmentkette | | | | | | | | | | | | | |
| **W** | **IP** | **NMÄ** | **IP** | **IP** | **W** | **IP** | **IP** | **W** | **IP** | **PAUSE** | **IP** | **W** | **IP** |
| ja:“ |  | HUSTET |  | ( | was | ) |  | denn |  | \*1,5\* |  | sonst | ↑ |

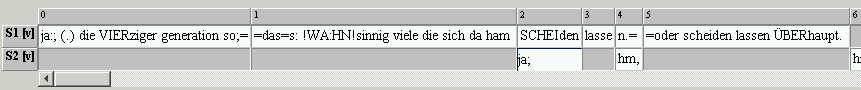
|  |  |  |
| --- | --- | --- |
| Mögliche Fehler | | |
| Ursache | Beispiel | Fehlermeldung |
| Großbuchstaben innerhalb von Wörtern | j**A** | Error: Nicht erlaubt: Großbuchstabe, offene Klammer, geschlossene Klammer, Punkt oder Ellipse, Zahl oder Komma |
| Kleinbuchstaben innerhalb nicht-morphemisierter Äußerungen | Ich mache**]** eine Äußerung. | Error: Nicht erlaubt: offene Klammer, geschlossene Klammer, Zahl oder Komma, Doppelpunkt, Punkt oder Ellipse, Pausenzeichen, Prosodiezeichen, Wortbestandteil |
| ... | ... | … |

## Segmentierung: „GAT: Intonation Units“

|  |  |  |
| --- | --- | --- |
| **Name** | **Standard-Werte** | **Erläuterung** |
| IU\_END\_SYMBOLS | **?** | **,** | **-** | **;** | **.** | | Markiert abschließend die Tonhöhenbewegung einer Phrasierungseinheit |
| OPEN\_PARENTHESIS | **(** | | Markiert Beginn einer Pause oder einer schwer verständlichen Passage. Punkte zwischen runden Klammern werden nicht als Endzeichen einer Phrasierungseinheit behandelt. |
| CLOSE\_PARENTHESIS | **)** | | Markiert Ende einer Pause oder einer schwer verständlichen Passage. Punkte zwischen runden Klammern werden nicht als Endzeichen einer Phrasierungseinheit behandelt. |
| CLOSE\_ANGLE | **>** | | Markiert Ende einer Kommentarannotation  (z.B.: <lachend< was?>) und kann hinter dem abschließenden Zeichen einer Phrasierungseinheit auftreten. |
| SPACE | | | Kann hinter dem abschließenden Zeichen einer Phrasierungseinheit auftreten. |
| EQUALS | **=** | | Markiert eine Verschleifung zweier Phrasierungseinheiten. Bei doppeltem Auftreten wird das erste Zeichen der ersten, das zweite Zeichen der zweiten Phrasierungseinheit zugeordnet. Bei einfachem Auftreten entscheidet die Verwendung eines Leerzeichens darüber, welcher Phrasierungseinheit das Zeichen zugeordnet wird. |

Beispiel:

Die folgende Segmentkette von Sprecher S1 wird mit der Segmentierung: „GAT: Intonation Units“ ...



... folgendermaßen in Phrasierungseinheiten (PE) segmentiert:

|  |  |  |  |
| --- | --- | --- | --- |
| **Segmentkette** | | | |
| **PE** | **PE** | **PE** | **PE** |
| ja:; | (.) die VIERziger generation so;= | =das=s: !WA:HN!sinnig viele die sich da ham SCHEIden lassen.= | =oder scheiden lassen ÜBERhaupt. |

## Segmentierung: „CHAT: Utterance“

|  |  |
| --- | --- |
| **Name** | **Standard-Werte** |
| PERIOD | **.** |
| QUESTION\_MARK | **?** |
| EXCLAMATION\_MARK | **!** |
| SPACE |  |

Erläuterung: Alle Äußerungsendzeichen in CHAT (also sowohl die „Basic Utterance terminators“ als auch die „Special Utterance Terminators“, vgl. MacWhiney 2000: 60 und 66 ff) setzen sich aus Zeichenketten zusammen, die mit einem Punkt, einem Fragezeichen oder einem Ausrufezeichen enden. Der Segmentierungs-Algorithmus erlaubt zusätzlich ein (optionales) Leerzeichen hinter diesen Äußerungsendzeichen.

Beispiel:

Die folgende Segmentkette von Sprecher CHI wird mit der Segmentierung: „CHAT: Utterance“ ...



... folgendermaßen in Äußerungen (U) segmentiert:

|  |  |  |  |
| --- | --- | --- | --- |
| **Segmentkette** | | | |
| **U** | **U** | **U** | **U** |
| Mummy! | I want something+... | Something to drink. | Can I? |

## Segmentierung: „IPA: Words and Syllables“

Der IPA-Segmentierungsalgorithmus segmentiert eine Transkription, die nach IPA-Konventionen angefertigt wurde, in Wörter und Silben. Die Details dieser Konventionen wurden in

Thoma, Dieter & Tracy, Rosemarie (2005): L1 and Early L2: What's the difference? Vortrag, DGfS-Jahrestagung in Köln.

vorgestellt. Eine schriftlich publizierte Version dieser Konventionen existiert z.Z. noch nicht. Die Konventionen sind aber, was die für die Segmentierung relevanten Zeichen anbelangt, denkbar einfach: Wörter werden mit einem Leerzeichen abgeschlossen, verschiedene Silben eines Wortes durch Punkte voneinander getrennt. Die Kennzeichnung von Silbengrenzen ist dabei optional, d.h. die Wort-Segmentierung funktioniert unabhängig von einer etwaigen weiteren Unterteilung der Wörter in Silben.

Im Unterschied zu den übrigen Segmentierungsalgorithmen werden bei der IPA-Segmentierung nicht generell alles Spuren des Typs ‚T’ segmentiert, sondern nur solche, denen darüber hinaus die Kategorie ‚v-pho’ (Kleinschreibung beachten!) zugewiesen wurde.

|  |  |
| --- | --- |
| **Name** | **Standard-Werte** |
| WordBoundaries | **<SPACE>** |
| SyllableBoundaries | **.** |

Erläuterung: Wörter werden durch Leerzeichen, Silben (optional) durch Punkte voneinander getrennt. Darüber hinaus erfolgt keine Überprüfung des konventionsgemäßen Gebrauchs von Transkriptionszeichen, insbesondere wird nicht überprüft, ob die verwendeten Zeichen Teil des IPA-Inventars sind – alle Zeichen außer Leerzeichen und Punkt werden als Bestandteile von Wörtern bzw. Silben interpretiert.

Beispiel:

Die folgende Segmentkette von Sprecher X wird mit der Segmentierung: „IPA: Words and syllables“ ...



... folgendermaßen in Wörter (w), Silben (sl), Wortgrenzen (wb) und Silbengrenzen (sb) segmentiert:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Segmentkette** | | | | | | | | | | | | | | | | | | | | |
| **hɜns.xen klain giŋ a.lain ɪn di: wai.tə wɜlt hi.nain** | | | | | | | | | | | | | | | | | | | | |
| **w** | | | **wb** | **w** | **wb** | **w** | **wb** | **w** | | | **wb** | **w** | **wb** | **w** | **wb** | **w** | **wb** | **w** | **wb** | **w** |
| hɜns.xen | | |  | klain |  | giŋ |  | a.lain | | |  | ɪn |  | di: |  | wai.tə |  | wɜlt |  | hi.nain |
| **sl** | **sb** | **sl** |  | **sl** |  | **sl** |  | **sl** | **sb** | **sl** | **...** | | | | | | | | | |
| hɜns | . | xen |  | klain |  | giŋ |  | a | **.** | lain |

|  |  |  |
| --- | --- | --- |
| Mögliche Fehler | | |
| Ursache | Beispiel | Fehlermeldung |
| Zwei Silben- oder Wortgrenzen folgen aufeinander | hɜns**..**xen  hɜns **.**xen | Error: No syllable or word boundary, no end of input allowed |
| Eine Segmentkette beginnt mit einer Wort- oder Silbengrenze | **.**hɜns.xen | Error: No syllable or word boundary, no end of input allowed |