

## Teil 1: (Domänendefinition)

Die Domäne „Container- und I/O-Library“ setzt sich intuitiv zusammen aus dynamischen Datenstrukturen sowie Ein- und Ausgabe-Behandlung. Für letztere sind noch zusätzliche Funktionen zur Formatierung und Konvertierung der externen Repräsentation von Daten notwendig.

## Teil 2: (Domänenlexikon)

|                     |  |
|---------------------|--|
| Ausgabe:            | Informationen die das System der Aussenwelt übergibt, z.B. per Display, serieller Schnittstelle (oder dort angeschlossener Geräte),...   |
| Container:          | Ein Container ist ein Objekt, das größere Mengen anderer Objekte verwalten kann.   |
| Eingabe:            | Information die von aussen dem System hinzugeführt wird, z.B. via Tastatur, Sensoren, serieller Schnittstelle, usw.  |
| Konvertierung:      | Unterschiedliche Computersysteme verwenden verschiedene interne Darstellungen für Zeichenketten, Zahlen, etc. Vor einem eventuellen Datenaustausch sollten diese in ein passendes Format konvertiert werden. |
| Reguläre Ausdrücke: | Komplexes Musterfinden und -ersetzen in Zeichenketten. (nach POSIX 1003.2; vgl weiterführende Literatur)   |
| Whitespace:         | „Unsichtbare“ Zeichen, wie z.B. Leerzeichen, Tabulatoren, Zeilenumbrüche, etc.   |

## Teil 3: (Konzeptmodelle)

- Eingabe
  - Zeichenbasierte Eingabe
  - Zeichenkettenbasierte Eingabe
  - Voraussetzung: Eingabetreiber
- Ausgabe
  - Zeichenbasierte Ausgabe
  - Zeichenkettenbasierte Ausgabe
  - Voraussetzung: Ausgabetreiber
- Container
  - Container fester Grösse (wird bei Initialisierung festgelegt)
    - ◆ einmalige Speicheranforderung
  - Container variabler Grösse
    - ◆ automatische (wiederholte) Speicheranforderung
  - Voraussetzung: dynamische Speicherverwaltung

## Teil 4: (Feature-Modell)

- Container- and I/O-Library
  - formatting and conversion of strings
    - ◆ comparison
    - ◆ conversion
      - ◆ unicode
      - ◆ ebcDIC
      - ◆ ascii
  - length determination
  - lowercase
  - uppercase
  - regular expressions
  - splitting
  - numbers
    - ◆ base
    - ◆ fixed point
    - ◆ floating point
  - s2numbers
    - ◆ base
    - ◆ fixed point
    - ◆ floating point
  - substring search
  - whitespace truncating
- dynamic structures
  - binary tree
    - ◆ balanced
  - hash
  - linked list
    - ◆ insert\_at
    - ◆ delete\_at
  - priority list
  - ringbuffer
  - stack
    - expandable
    - fixed size
- input streams
  - char
  - string
  - number
    - ◆ fixed point
      - ◆ base
      - ◆ signed
    - ◆ floating point
- output streams
  - char
  - string
  - number
  - pointer
  - left justified
    - ◆ right justified
      - centered
  - ◆ free positioning

Legende:

● Alternative Feature

◆ Optional Feature

■ Mandatory Feature