

Systembeschreibung

jCAPT – Java Content Assembling and Packaging Tools

FIM Institut für Informationsverarbeitung und Mikroprozessortechnik <jcapt@fim.uni-linz.ac.at>

Inhaltsverzeichnis

1. Grundkonzepte	1
1.1. e-Learning	2
1.2. Arbeitsablauf	3
1.3. Lernpaket	4
1.4. Metadaten	5
2. Systembeschreibung	7
2.1. Eigenschaften	8
2.2. Zweck	9
2.3. Voraussetzungen	10
3. Funktionale Eigenschaften	13
3.1. Funktionen	14
3.2. Einschränkungen	20
4. Daten	21
4.1. Eingaben	22
4.2. Ausgaben	23

Kapitel 1. Grundkonzepte

Um den Anforderungen der Wirtschaft nach ständigem Wandel und hoher Flexibilität gerecht zu werden darf "lebenslangliches Lernen" nicht nur ein Schlagwort sein, sondern muss zusammen mit der Aneignung von problemorientiertem Wissen und Kompetenzen im Bereich der Selbstorganisation schon in der Ausbildung berücksichtigt werden. Die Entwicklung geht in Richtung selbstgesteuertes Lernen als aktiv-konstruktiver Prozess, wobei die Rolle des Lehrenden als Betreuer gemeinsam mit einfachen Möglichkeiten zum Austausch von Wissen und Materialien mit anderen Lernenden und eine Evaluierung als integrierte Rückmeldung über den Lernprozess grundlegend verschiedene didaktische Modelle und Werkzeuge erfordern.

Dieses neue Lernparadigma erfordert eine radikale Umstellung der Art, wie Wissen vermittelt wird. Doch nicht nur das *Wie* im Sinne der Aufbereitung des Lernstoffs sondern auch die Art der Kommunikation, deren Geschwindigkeit und zeitliche Verfügbarkeit stellen neue Anforderungen an die Produzenten von Lernmaterialien. Um der nötigen Interaktivität, der Möglichkeit zu selbstgesteuertem Lernen und der schnellen Entwicklung und Wartung der Unterlagen gerecht zu werden findet eine zunehmende Hinwendung zu den Techniken des Distance Learning/Teaching/Coaching statt.

1.1. e-Learning

Die neuen Formen des elektronischen Lernens (e-Learning) ermöglichen eine zeitgemäße Vermittlung von Wissen, bei denen der Lernende aktiv involviert ist. Dies beginnt schon beim selbstverantworteten Studium der präsentierten Unterlagen, geht über die eigenständige Lösung von integrierten Lernüberprüfungen und reicht bis zur Kollaboration mit anderen Lernenden im Rahmen von Diskussionen und beim Erarbeiten von Gruppenaufgaben. Der Vorteil, den e-Learning Technologien hier bieten, sind vielfältig:

- reichere Lernmaterialien (mit Animationen, interaktiven Experimenten, Selbsttests, etc.)
- keine örtliche und zeitliche Bindung des Lernenden (Stichwort Selbstorganisation)
- schnelle Überprüfbarkeit des Wissensstands durch den Lehrenden selbst
- sofortige Rückmeldung über Lernerfolg für den Lernenden
- einfachere und schnellere Wartung der Unterlagen
- vielfältige Möglichkeiten zur Kommunikation unter den Lernenden und mit dem Lehrenden

1.2. Arbeitsablauf

Ein typischer Arbeitsablauf im e-Learning, von der Kurserstellung bis zur -veröffentlichung sieht von der Seite der Lehrenden folgendermaßen aus:

1. Erstellung der Lernmaterialien:
 - Schreiben von Haupttext und Glossar
 - Anreicherung mit Bildern und Grafiken
 - Aufwertung zum Hypermedium durch Verlinkung
 - Erstellung bzw. Suche von erläuternden Animationen, Video- und Tonsequenzen
 - Entwicklung von interaktiven Inhalten (Experimente, Applets, Quizzes)
2. Organisation der Materialien auf einem Datenträger in einer physischen Struktur
3. Festlegung der logischen Struktur und Zusammenfügen zu einem Lernpaket (siehe Abschnitt 1.3, „Lernpaket“)
4. Anreicherung des Lernpakets mit Metadaten (siehe Abschnitt 1.4, „Metadaten“)
5. Zurverfügungstellung des Lernpakets:
 - Einbindung in ein Learning Management System (LMS)
 - Erstellung einer Offline-Version für die Ausgabe auf einer CD
 - Bereitstellung einer Version für mobile Geräte (z.B. PDAs)

1.3. Lernpaket

Als ein Lernpaket bezeichnet man die Kombination einer Menge von Lernmaterialien mit einer Strukturbeschreibung des durch diese Materialien gebildeten Kurs(teils). Diese Beschreibung wird von Anwendungen zum Betrachten des Pakets verwendet, um dem Benutzer den Aufbau und Ablauf des Kurses zu vermitteln und auf die mit den Kursteilen verknüpften Lernmaterialien zuzugreifen. Oft ebenfalls Teil der Beschreibung sind Metadaten, also Daten über Daten, wie sie in Abschnitt 1.4, „Metadaten“ beschrieben werden. Um einen reibungslosen Austausch von Lernpaketen zwischen den e-Learning Systemen verschiedener Hersteller zu gewährleisten, haben sich Standards herausgebildet. Als de-facto Industriestandard gilt hier die CPS Content Packaging Specification vom IMS Global Learning Consortium.

1.3.1. Content Packaging Specification

Beschreibung des CPS Standards ...

1.4. Metadaten

Von enormer Bedeutung beim e-Learning sind Metadaten zu den Kurs(teilen). Erst sie erlauben es, das richtige Lernmaterial zu finden und in seiner Eignung für ein bestimmtes Lern/Lehr-Vorhaben zu bewerten. Hierbei werden oft nicht nur allgemeine Informationen wie Titel, Autor, Sprache, Format, Verwendungsbestimmungen und technische Voraussetzungen gegeben. Noch viel wichtiger sind Daten zur Klassifikation des Inhalts und pädagogische Informationen zur Altersstufe, der Zielgruppe, der semantischen Dichte und des didaktischen Aufbaus.

Wie schon bei den Lernpaketen haben sich auch hier mehrere Standards herausgebildet, die sich jedoch im Inhalt in weiten Teilen gleichen und nur Unterschiede in der Notation aufweisen. Ein sehr umfassender Standard ist LOM Learning Object Metadata der Working Group 12 des Learning Technology Standards Committee (LTSC) von IEEE. Darauf aufbauend existieren der IMS LRM Learning Resource Metadata Standard und lokale Standards wie beispielsweise der Metadaten-Standard des Bundesministerium für Bildung Wissenschaft und Kunst (BMBWK) für Österreich.

1.4.1. Learning Object Metadata

Beschreibung des LOM Standards ...

1.4.2. IMS Learning Resource Metadata

Beschreibung IMS MD Standards ...

1.4.3. Österreichische Metadaten des BMBWK

Beschreibung des BMBWK Standards ...

Kapitel 2. Systembeschreibung

2.1. Eigenschaften

jCAPT ist ein graphischer Editor für Lernpakete. Es lässt sich zur Bearbeitung und Unterstützung folgender unter Abschnitt 1.2, „Arbeitsablauf“ erwähnter Teilschritte einsetzen:

1. Material-Organisation (physische Struktur)
2. Lernpaket aufbauen und bearbeiten (logische Struktur)
3. (automatische) Metadatenerstellung und -bearbeitung
4. Export des Lernpakets: gepackt für LMS bzw. als Offline-Version

jCAPT bietet die Möglichkeit zur *Sammlung* und *Organisation* von Lernmaterial zu einem Grundstock an Ressourcen. Mit diesen kann anschließend die *logische Struktur* des Lernpakets nach der IMS Content Packaging Specification (siehe Abschnitt 1.3.1, „Content Packaging Specification“) aufgebaut werden. Um einen Überblick über die verwendeten Materialien zu behalten, können diese in einer *Vorschau* innerhalb von jCAPT betrachtet werden. Ebenso existiert eine Vorschau für das Lernpaket, um schon bei der Erstellung einen Eindruck vom Endprodukt zu erhalten.

jCAPT unterstützt die Erstellung und Bearbeitung von *Metadaten* für das Gesamtpaket, sowie auch für Teile des Pakets und – auf unterster Granularitätsebene – für die verwendeten Ressourcen. Unterstützt werden die Standards IMS LRM, LOM und der Metadatenstandard des BMBWK für Österreich. Es werden hier sowohl die *maskenbasierte* Bearbeitung als auch das direkte Arbeiten mit der *Baumstruktur* der Metadaten ermöglicht. Zur Erleichterung einer aufwändigen Eingabe von Metadaten können diese aus den im Lernpaket verwendeten Ressourcen *automatisch* gewonnen werden.

Der *Export* eines Lernpakets kann durch jCAPT einerseits als *ZIP-Archiv* der Ressourcen und des Manifests (zur Verwendung in einem Learning Management System) und andererseits als *Offline-Version* erfolgen. Eine Offline-Version enthält sämtliche Ressourcen des Pakets, eine *Navigationskomponente* und eine *Metadatenanzeige*. Die Navigationskomponente spiegelt die erstellte logische Struktur des Lernpakets wider; die Metadatenanzeige stellt die eingegebenen Metadaten dar. Zur Wahrung der Plattformunabhängigkeit sind diese Teile als (dynamische) HTML-Seiten realisiert, die Anzeige des Lernpakets ist somit in einem Browser möglich.

Neben der Erstellung von neuen Lernpaketen unterstützt jCAPT auch das *Bearbeiten* von bereits bestehenden Lernpaketen, sofern diese auf Basis von unterstützten Standards angelegt sind.

2.2. Zweck

jCAPT wurde mit dem Ziel geschaffen, die Erstellung und Bearbeitung von Lernpaketen in zeitgemäßer und standardisierter Weise zu ermöglichen. Zielgruppe sind sowohl Einsteiger als auch fortgeschrittene Benutzer, die genaue Kontrolle über Inhalt und Aufbau des Lernpakets verlangen. Eine weitere Zielsetzung von jCAPT war darüber hinaus die Integration schon bestehender Anwendung (zur Offline-Konvertierung und automatisierten Metadaten-Extraktion) in einen Lernpaket-Editor, um den Arbeitsfluss zu optimieren.

2.3. Voraussetzungen

Die Voraussetzungen für die Anwendung von jCAPT lassen sich in nicht-technische und technische einteilen. *Nicht-technische Voraussetzungen* sind:

- Grundlegende Kenntnisse im Umgang mit einem Computersystem:
 - Programme starten, zwischen ihnen wechseln und beenden
 - Datei- und Ordnerkonzept u.a. mit Dateieindungen
 - Dateiverwaltung: erstellen, umbenennen, kopieren, verschieben, löschen von Dateien und Ordnern
 - Installation und Konfiguration von (Hilfs-)Anwendungen
- Fähigkeiten im Umgang mit einer graphischen Benutzeroberfläche (GUI):
 - Menüs (Hauptmenü und Kontextmenü) und Schaltflächen
 - Dialoge und Fenster (verschieben, vergrößern, verkleinern, minimieren, maximieren, schließen)
 - Eingabekomponenten (Textfelder, Komboboxen, ...)
 - Baumstrukturen
 - Drag and Drop
 - Copy, Cut, Paste
- Kenntnis des CPS Standards (siehe Abschnitt 1.3.1, „Content Packaging Specification“):
 - Knotentypen (z.B. Organizations, Organization, Item, Resource) und deren Attribute
 - Aufbau eines Lernpakets (Baumstruktur)
 - Abhängigkeiten von externen und zwischen Ressourcen
 - Möglichkeiten zur Metadateneinbindung
- Kenntnis eines Metadaten Standards (siehe Abschnitt 1.4, „Metadaten“):
 - Bedeutung der einzelnen Felder
 - Pflichtangaben
 - Alternativen (z.B. sprachlicher Natur)
 - Vokabular (standardisierte Werte)
- Verfügbarkeit von in einem Lernpaket zu verpackende Ressourcen

Die technischen Voraussetzungen seien hier nur grob angeführt, eine genauere Auflistung findet sich in der Installationsanleitung:

- Microsoft Windows®
- Sun Java™ Runtime Environment 1.4.2 oder neuer
- Internet Explorer oder Mozilla Browser
- Programme zum Betrachten/Bearbeiten der jeweiligen Ressourcen, beispielsweise:
 - Adobe Acrobat Reader®

- Microsoft Office (Word, PowerPoint, Excel, ...)
- Macromedia Flash Player
- Windows Media Player, Quicktime, Real Player

Kapitel 3. Funktionale Eigenschaften

Die in Abschnitt 2.1, „Eigenschaften“ genannten Möglichkeiten von jCAPT werden im Folgenden näher beleuchtet. Es handelt sich hierbei um eine detailliertere Beschreibung der funktionalen Möglichkeiten sowie der Einschränkungen in Bezug auf die Be- und Verarbeitung von Lernpaketen und Metadaten unter jCAPT. Nicht Teil dieser Beschreibung ist die konkrete Bedienung zur Verwendung der Funktionen. Eine dahingehende Dokumentation findet sich im aufgabenorientierten Teil der Bedienungsanleitung.

3.1. Funktionen

3.1.1. Material-Organisation

jCAPT bietet grundlegende Möglichkeiten zur Sammlung und physischen Strukturierung von Lernmaterial, welches später im Manifest des Lernpakets strukturiert werden soll. Bei der Erstellung eines neuen Lernpakets muss dazu ein Ordner angegeben werden, welcher als *Basisordner* dient. Dieser Ordner kann leer sein oder schon Ressourcen enthalten. Außerhalb von jCAPT können in der Folge durch entsprechende Programme für Dateimanagement (z.B. Windows Explorer) jederzeit Datei- und Ordneroperationen in diesem Ordner durchgeführt werden, um die Materialien zu sammeln bzw. zu organisieren. Dahingehende Änderungen können bei geöffnetem Lernpaket im Editor durch entsprechende Benutzeraktionen sichtbar und verwendbar gemacht werden, eine automatische Erkennung der Änderungen im Dateisystem ist nicht vorgesehen.

Neben der Möglichkeit der Material-Organisation durch externe Programm und der Übernahme dieser Änderungen bietet jCAPT auch interne Funktionen zur Erledigung der Sammel- und Organisationsaufgaben:

- Ordner anlegen
- Datei(en) zu Ordner(n) hinzufügen, entweder
 - als alleinstehende Datei(en)
 - als Datei(en) mit ihren Abhängigkeiten (z.B. Bilder in HTML-Seiten)
- Datei(en) und Ordner löschen
- Datei(en) und Ordner in der Ordnerstruktur unterhalb des Basisordners
 - kopieren
 - verschieben

Die somit gesammelten und organisierten Materialien bilden die Ressourcen, aus denen das Lernpaket aufgebaut ist. Ihr Inhalt kann jeweils in einem Vorschauenfenster betrachtet werden. Dies gilt nur für Dateien, deren Typ durch die Vorschau unterstützt werden. Standardmäßig sind dies (mit den Dateierendungen):

- statische Webseiten: htm, html
- Bilder: gif, jpg, jpeg, png
- PDFs: pdf
- PowerPoint-Präsentationen: ppt
- Textdateien: txt
- ShockWave-Dateien: swf

Die Liste der unterstützten Dateitypen kann je nach den Fähigkeiten der Vorschau erweitert oder eingeschränkt werden. Ebenso besteht die Möglichkeit, eine Datei mit ihrer assoziierten Anwendung aus jCAPT heraus zu starten. Dies funktioniert mit allen Dateien, welche im darunterliegenden Betriebssystem eine derartige Assoziation besitzen.

3.1.2. Lernpaket aufbauen und bearbeiten

Wie schon im vorangegangenen Unterkapitel erwähnt, muss bei der Erstellung eines Lernpakets ein *Basisordner* angegeben werden. In dieses werden das Manifest und dazugehörige Schemendefinitionen gespeichert. Das Manifest besteht wie jedes XML-Dokument aus einer im Schema festgelegten Hierarchie (Baum) von XML-Elementen (Knoten) und kann in jCAPT auf folgende Arten erstellt werden:

- *Manuelles Anlegen* der Knoten und Ausfüllen der Werte für die Attribute
- Intuitives Erzeugen und Verschieben der Knoten *mittels Drag-and-Drop*, wobei Attributwerte meist automatisch ausgefüllt und abhängige Knoten ebenfalls erzeugt werden

Diese zwei Ansätze schließen sich jedoch nicht aus; die einzelnen Knoten können je nach Eignung auf die eine oder andere Weise erzeugt und bearbeitet werden. Zum Anlegen der Grundstruktur eignet sich Drag-and-Drop besser, für Feinheiten muss oft auf die manuelle Bearbeitung umgestellt werden.

Ebenso wie bei der Material-Organisation (siehe Abschnitt 3.1.1, „Material-Organisation“) besteht auch hier die Möglichkeit der Einblendung einer *Vorschau* für den Inhalt einer Ressource. Das Öffnen der mit einer Ressource assoziierten Anwendung ist in ebenfalls Fall möglich. Details zu den Modalitäten der Vorschau finden sich in Abschnitt 3.1.1, „Material-Organisation“.

Um einen Eindruck vom fertigen Paket zu erlangen, kann auch für dieses eine Vorschau erstellt werden. Diese zeigt jene Organisation (siehe Abschnitt 3.1.2.6, „Organisation (*organization*)“), welche gerade gewählt ist, die eingestellte Standard-Organisation, falls kein Knoten innerhalb einer Organisation gewählt ist, bzw. die erste Organisation des Manifests, wenn auch keine Standard-Organisation festgelegt ist.

Im Folgenden werden die einzelnen Knoten und Attribute der IMS Content Packaging Specification und die Möglichkeit zu ihrer Erstellung, Bearbeitung und Vorschau genannt.

3.1.2.1. Manifest (*manifest*)

Dieser Knoten wird als oberster Knoten bei der Erzeugung des Manifests angelegt und kann weder verschoben noch kopiert oder gelöscht werden. Die folgenden Attribute können bearbeitet werden:

- Basis (*xml:base*): Ausgangspunkt für relative Pfade
- Identifikation (*identifier*): verpflichtende Angabe einer im Manifest eindeutigen Identifikation; wird beim Erstellen automatisch mit einem Wert (in der Form `MANIFEST-XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX`, wobei für jedes X eine hexadezimale Ziffer stehen kann) versehen
- Version (*version*): Version des Manifests

3.1.2.2. Metadaten (*metadata*)

Metadaten-Knoten enthalten keine Attribute und können gemäß der IMS Content Packaging Specification für folgende Knoten als deren Kinder erzeugt werden:

- Manifest (*manifest*)
- Organisation (*organization*)
- Inhalt (*item*)
- Ressource-Referenz (*resource*)

Da diese die Existenz eines Metadaten-Knoten nicht vorschreiben kann man ihn auch wieder löschen. Ebenso ist ein Kopieren bzw. Verschieben an eine andere passende Stelle des Manifests möglich. Die Beschreibung der Bearbeitung des Inhalts eines Metadaten-Knoten erfolgt bis auf die zwei Ausnahmen Schema und Schema Version unter Abschnitt 3.1.3, „Metadatenerstellung und -bearbeitung“).

3.1.2.3. Schema (*schema*)

Dieser Knoten beschreibt das im Manifest verwendete Schema. Er besitzt keine Attribute und kann als Kindknoten eines Metadaten-Knoten angelegt und wieder gelöscht werden.

3.1.2.4. Schema Version (`schemaversion`)

Dieser Knoten beschreibt die Version des im Manifest verwendeten Schema. Er besitzt keine Attribute und kann als Kindknoten eines Metadaten-Knoten angelegt und wieder gelöscht werden.

3.1.2.5. Organisationen (`organizations`)

Der Organisationten-Knoten liegt unterhalb des Manifest-Knotens und kann nach seiner Erzeugung beim Anlegen eines neuen Lernpakets weder verschoben noch kopiert oder gelöscht werden. Die folgenden Attribute können bearbeitet werden:

- Standard-Organisation (`default`): verpflichtende Angabe der Identifikation jener Organisation, welche standardmäßig (z.B. beim Darstellen des Pakets) benutzt wird.

3.1.2.6. Organisation (`organization`)

Dieser Knoten liegt unter dem Organisationen-Knoten und enthält eine Hierarchie von Inhalts-Knoten bestimmte logische Strukturierung des Inhalts. Bearbeitbare Attribute sind:

- Identifikation (`identifier`): verpflichtende Angabe einer im Manifest eindeutigen Identifikation; wird beim Erstellen und Einfügen aus der Zwischenablage automatisch mit einem Wert (in der Form `ORG-XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXXXXXX`, wobei für jedes X eine hexadezimale Ziffer stehen kann) versehen
- Struktur (`structure`): Struktur der Organisation; beim Erstellen wird der vom Standard vorgesehene Wert `hierarchical` gesetzt

Ein Organisationen-Knoten kann beliebig viele Organisation-Knoten enthalten. Diese können kopiert, ausgeschnitten, wieder eingefügt und gelöscht werden. Ebenso ist eine Änderung ihrer Reihenfolge möglich. Der angezeigte Name des Knotens richtet sich nach den in Abschnitt 3.1.2.7, „Titel (`title`)“ beschriebenen Regeln.

3.1.2.7. Titel (`title`)

Der Titel-Knoten kann unter einem Organisation- als auch einen Inhalt-Knoten liegen und gibt dessen textuelle Beschreibung an. Unter jCAPT wird dieser Knoten nicht direkt dargestellt sondern ist über den jeweiligen Vaterknoten zu bearbeiten. Ist dort die Beschreibung eingetragen, so wird der Knoten implizit erzeugt und der dazugehörige Organisation- bzw. Inhalt-Knoten wird mit diesem Namen dargestellt. Fehlt sie, wird er gelöscht und der dazugehörige Organisation- bzw. Inhalt-Knoten wird mit den symbolischen Namen `Organisation` bzw. `Inhalt` als Namen dargestellt. Der Knoten besitzt keine Attribute.

3.1.2.8. Inhalt (`item`)

Dieser Knoten liegt entweder unter einem Organisation- oder einem anderen Inhalt-Knoten und beschreibt einen Teil des Inhalts innerhalb der logischen Struktur der Organisation. Bearbeitbare Attribute sind:

- Identifikation (`identifier`): verpflichtende Angabe einer im Manifest eindeutigen Identifikation; wird beim Erstellen und Einfügen aus der Zwischenablage automatisch mit einem Wert (in der Form `ITEM-XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXXXXXX`, wobei für jedes X eine hexadezimale Ziffer stehen kann) versehen
- Referenziertes Element (`identifizierref`): eine Referenz auf die Identifikation eines Ressourc-Referenz-Knoten oder eines (Sub-)Manifests; jCAPT erlaubt die Auswahl aus der Liste von Ressourc-Referenz-Knoten und verhindert somit die aufwendige und fehleranfällige Eingabe von Identifikationen

- Sichtbar (`isVisible`): gibt an, ob der Inhaltsknoten beim Darstellen des Lernpakets angezeigt werden soll
- Parameter (`parameters`): Parameter, welche der referenzierten Ressource beim Start übergeben werden

Ein Organisation- bzw. Inhalt-Knoten kann beliebig viele Inhalt-Knoten enthalten. Diese können kopiert, ausgeschnitten, wieder eingefügt und gelöscht werden. Ebenso ist eine Änderung ihrer Reihenfolge möglich. Der angezeigte Name des Knotens richtet sich nach den in Abschnitt 3.1.2.7, „Titel (`title`)“ beschriebenen Regeln.

Für einen Inhalt-Knoten zeigt das Vorschauenfenster die referenzierte Ressource des angegebenen Ressource-Referenz-Knotens, sofern diese existiert und ihr Typ von der Vorschau unterstützt wird. Diese Ressource kann auch in ihrer assoziierten Anwendung geöffnet werden.

3.1.2.9. Ressource-Referenzen (`resources`)

Der Ressource-Referenzen-Knoten liegt unterhalb des Manifest-Knotens und kann nach seiner Erzeugung beim Anlegen eines neuen Lernpakets weder verschoben noch kopiert oder gelöscht werden. Die folgenden Attribute können bearbeitet werden:

- Basis (`xml:base`): Ausgangspunkt für relative Pfade von referenzierten Ressourcen

3.1.2.10. Ressource-Referenz (`resource`)

Dieser Knoten kann in beliebiger Zahl als Kind des Ressource-Referenzen-Knotens vorkommen. Die bearbeitbaren Attribute sind:

- Identifikation (`identifier`): verpflichtende Angabe einer im Manifest eindeutigen Identifikation; wird beim Erstellen und Einfügen aus der Zwischenablage automatisch mit einem Wert (in der Form `RES-XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXXXX`, wobei für jedes X eine hexadezimale Ziffer stehen kann) versehen
- Typ (`type`): gibt den Typ der Ressource an; die Werte `text/html` und `webcontent` (Standard) können eingestellt werden
- Basis (`xml:base`): Ausgangspunkt für relativen Pfade zu Ressourcen
- Link (`href`): eine Referenz (z.B. URL) zum "Einstiegspunkt" der Ressource; kann relativ sein, auch wenn keine Basis angegeben ist, in diesem Fall bezieht sich der Pfad auf das Basisorder

Ein Ressource-Referenz-Knoten kann kopiert, ausgeschnitten, eingefügt und gelöscht werden. Ebenso ist eine Änderung ihrer Reihenfolge möglich, wobei dies jedoch nur Auswirkungen auf die Platzierung des XML-Elements im Manifest nicht jedoch auf die Anzeige des Lernpakets hat. Der angezeigte Name des Ressource-Referenz-Knotens entspricht dem gegebenen Link und bei dessen Fehlen der Identifikation. Fehlt auch diese, so wird der symbolische Name `Resource` angezeigt.

Für einen Ressource-Referenz-Knoten zeigt das Vorschauenfenster die referenzierte Ressource, sofern diese existiert und ihr Typ von der Vorschau unterstützt wird. Diese Ressource kann auch in ihrer assoziierten Anwendung geöffnet werden.

3.1.2.11. Abhängigkeit (Datei/URL) (`file`)

Der Datei-Knoten kann in beliebiger Zahl als Kind eines Ressource-Referenz-Knotens vorkommen. Das folgende Attribut kann bearbeitet werden:

- Link (`href`): eine Referenz (z.B. URL) auf eine Datei; kann relativ sein, auch wenn keine Basis angegeben ist, in diesem Fall bezieht sich der Pfad auf den Basisordner

Ein Abhängigkeit (Datei/URL)-Knoten kann kopiert, ausgeschnitten, eingefügt und gelöscht werden. Ebenso ist eine Änderung ihrer Reihenfolge (zwischen den Knoten dieses Typs) möglich, wobei dies jedoch nur Auswirkungen auf die Platzierung des XML-Elements im Manifest nicht jedoch auf die Anzeige des Lernpakets hat. Der angezeigte Name des Knotens entspricht dem gegebenen Link. Bei dessen Fehlen wird der symbolische Name *Datei* angezeigt.

Für einen Abhängigkeit (Datei/URL)Knoten zeigt das Vorschauenfenster die referenzierte Ressource, sofern diese existiert und ihr Typ von der Vorschau unterstützt wird. Diese Ressource kann auch in ihrer assoziierten Anwendung geöffnet werden.

3.1.2.12. Abhängigkeit (Ressource-Referenz) (dependency)

Dieser Knoten kann in beliebiger Zahl als Kind eines Ressource-Referenz-Knotens vorkommen. Das folgende Attribut kann bearbeitet werden:

- Ressource-Referenz (*identifizierref*): die Identifikation des benötigten Ressource-Referenz-Knoten; jCAPT erlaubt die Auswahl aus der Liste von Ressource-Referenz-Knoten und verhindert somit die aufwendige und fehleranfällige Eingabe von Identifikationen

Ein Abhängigkeit (Ressource-Referenz)-Knoten kann kopiert, ausgeschnitten, eingefügt und gelöscht werden. Ebenso ist eine Änderung ihrer Reihenfolge (zwischen den Knoten dieses Typs) möglich, wobei dies jedoch nur Auswirkungen auf die Platzierung des XML-Elements im Manifest nicht jedoch auf die Anzeige des Lernpakets hat. Der angezeigte Name des Knotens entspricht dem der referenzierten Ressource-Referenz. Bei deren Fehlen wird der symbolische Name *Abhängigkeit* angezeigt.

Für einen Abhängigkeit (Ressource-Referenz)-Knoten zeigt das Vorschauenfenster die referenzierte Ressource der angegebenen Ressource-Referenz, sofern diese existiert und ihr Typ von der Vorschau unterstützt wird. Diese Ressource kann auch in ihrer assoziierten Anwendung geöffnet werden.

3.1.3. Metadatenerstellung und -bearbeitung

Wie in der IMS Content Packaging Specification (siehe) vorgesehen können unter jCAPT folgende Knoten-Typen im Lernpaket mit Metadaten ausgezeichnet werden:

- Manifest (*manifest*)
- Organisation (*organization*)
- Inhalt (*item*)
- Ressource (*resource*)

Unterstützt werden hierfür die folgenden Standards:

- IMS Learning Resource Metadata in den Versionen 1.1, 1.2.1 und 1.2.2
- IEEE LOM Learning Object Metadata Version 1.0
- Österreichische Metadatenspezifikation für elektronische Lehr-/Lernressourcen Version 1.32

jCAPT unterstützt die Erstellung und Bearbeitung von *Metadaten* für das Gesamtpaket, sowie auch für Teile des Pakets und – auf unterster Granularitätsebene – für die verwendeten Artefakte. Unterstützt werden die Standards IMS LRM, LOM und der Metadatenstandard des Bundesministerium für Bildung Wissenschaft und Kunst für Österreich. Es werden hier sowohl die *maskenbasierte* Bearbeitung als auch das direkte Arbeiten mit der *Baumstruktur* der Metadaten ermöglicht. Zur Erleichterung einer mühsamen Eingabe von Metadaten können diese aus den im Lernpaket verwendeten Artefakten *automatisch* zu gewinnen.

3.1.4. Lernpaketexport

jCAPT unterstützt sowohl den Export des Lernpakets als ZIP-Archiv als auch in Form einer Offline-Version beispielsweise für die Erstellung von Lern-CDs. Das Zippen packt alle Dateien unterhalb des Basisordners des Lernpakets in eine ZIP-Datei, deren Name und Speicherort gewählt werden kann.

Beim Erstellen einer Offline-Version des Lernpakets wird im gewählten Ausgabeverzeichnis eine materialisierte Sicht auf die Ressourcen erstellt. Diese Materialisierung enthält die Ressourcen und eine Navigationskomponente, welche das Betrachten des Lernpakets mit seinen Ressourcen in einem Browser ermöglicht. Ebenso werden die Metadaten des Manifests dargestellt und Hilfen zur Kursnavigation wie beispielsweise eine Roadmap gegeben. Beim Export werden vielfältige Möglichkeiten für Einstellungen geboten:

- Titel des Lernpakets
- Darzustellende Organisation
- Ausgabeordner für materialisierte Seiten (verpflichtend)
- Zu verwendende Seitenvorlagen (für Anzeigestil und Aufbau der materialisierten Seiten; verpflichtend)
- Verweis auf die Übersichtsseite (z.B. zum Rücksprung auf einen CD-Index)
- Prüfung auf Referenzen auf Dateien außerhalb des Ausgabeordners
- Packen in ein ZIP-Archiv
- Sprache der zu materialisierenden Metadaten
- Submanifestkomprimierung
- Unsichtbarkeitsvererbung
- Inhalt der Startseite
- Bild in der Kopfzeile
- Taxonomiefilterung
- Roadmapeinstellungen
- Indizierung

Für folgende Einstellungen werden Standardwerte gesetzt, welche vom Benutzer verändert werden können:

- Titel: der Name des Basisordners
- Darzustellende Organisation: die im Organisationen-Knoten (siehe Abschnitt 3.1.2.5, „Organisationen (organizations)“) eingestellte
- Ausgabeordner: der Basisordner
- Seitenvorlagen: die mitgelieferten Vorlagen in der Systemsprache bzw. Englisch, wenn diese nicht unterstützt wird
- Sprache der zu materialisierenden Metadaten: x-none um Metadaten einer jeden Sprache anzuzeigen

Die getätigten Einstellungen können zur Wiederverwendung in einer vom Benutzer festgelegten Datei gespeichert werden. Ein Laden von gespeicherten Einstellungen ist ebenfalls möglich. Während des Exports der Offline-Version werden die Ausgaben des Konverters angezeigt, um eventuelle Probleme (z.B. falsche Referenzen) erkennen zu können. Anschließend besteht die Möglichkeit der Anzeige der Offline-Version des Lernpakets im Standardbrowser des Systems.

3.2. Einschränkungen

3.2.1. Material-Organisation

Das Mitimportieren von Abhängigkeiten einer zu importierenden Ressource funktioniert nur bei Dateien, für welche ein Parser verfügbar ist. Es sind dies (mit den Dateiendungen):

- Statische Webseiten: htm, html, shtm, shtml
- Stylesheets: css
- JavaScript-Dateien: js
- PHP-Seiten: php, php3, php4
- ASP-Seiten: asp, aspx, asa
- ColdFusion-Seiten: cfm

3.2.2. Lernpaket aufbauen und bearbeiten

Beim Einfügen eines über eine Identifikation verfügenden Knotens (Organisation, Ressource, Inhalt) aus der Zwischenablage wird diese Identifikation neu generiert, selbst wenn sie im Manifest noch nicht vorhanden ist (z.B. weil der Knoten vorher ausgeschnitten wurde). Etwaige Referenzen auf diesen Knoten werden somit ungültig und müssen neu gesetzt werden.

jCAPT garantiert die Speicherung eines im Sinne des XML-Schemas korrekt validierenden Manifest, erzwingt jedoch nicht das Vorhandensein gewisser verpflichtender Angaben, so wie es die IMS Content Packaging Specification vorsieht:

- Identifikation für die Knoten Manifest, Organisation, Inhalt und Ressource
- Standard-Organisation beim Organisationen-Knoten
- Link beim Datei-Knoten
- Datei-Knoten für Ressourcen, bei denen der Link nicht absolut ist

Ebenso wird nicht geprüft, ob die Referenzen auf die Standard-Organisation und Ressourcen alle auf gültige und existierende Knoten verweisen.

3.2.3. Metadatenerstellung und -bearbeitung

3.2.4. Lernpaketexport

Es wird der gesamte Inhalt des Basisordners zu einem ZIP-Archiv gepackt, selbst wenn gewisse Dateien nicht im Manifest referenziert werden. Dies gilt auch für die Kopien der Ressourcen, welche im Zielordner angelegt werden.

Kapitel 4. Daten

Da es sich bei jCAPT um ein interaktives graphisches Programm handelt, kommt der Beschreibung der Ein- und Ausgabedaten nicht jene Bedeutung zu, wie sie es beispielsweise bei einer Applikation zum Video-Streaming hätte. Nichtsdestotrotz sei dieser Aspekt von jCAPT hier besprochen, wenn auch nicht in der Detailliertheit und Ausführlichkeit, wie dies bei nicht-interaktiven Programmen sicherlich nötig ist. Die Unterteilung der einzelnen Unterkapitel entspricht wiederum der unter Abschnitt 2.1, „Eigenschaften“ aufgezählten Teilschritte.

4.1. Eingaben

4.1.1. Material-Organisation

- Basisordner des Lernpakets, eventuell mit den Namen zu erstellender Ordner für den vollständigen Pfad
- Version des für das Manifest zu verwendenden Schemas der IMS Content Packaging Specification
- Name und Pfad von neu anzulegenden Ordnern unterhalb des Basisordners
- Lernmaterial mit beliebigem Dateityp und dessen Abhängigkeiten zum Importieren
- Pfad zu jenen Dateien, welche in der Vorschau bzw. der assoziierten Anwendung betrachtet werden sollen

4.1.2. Lernpaket aufbauen und bearbeiten

4.1.3. Metadatenerstellung und -bearbeitung

4.1.4. Lernpaketexport

Beim Export als ZIP-Archiv werden folgende Eingabedaten benötigt:

- Geöffnetes Lernpaket
- Name und Pfad des zu erstellenden ZIP-Archivs

Soll eine Offline-Version des Lernpakets erstellt werden, so sind folgende Eingabedaten erforderlich:

- Geöffnetes Lernpaket mit validem, standardkonformem Aufbau
- korrekt referenzierte Ressourcen unterhalb des Basisordners
- Einstellungen gemäß der Auflistung in Abschnitt 3.1.4, „Lernpaketexport“ mit folgenden verpflichtenden Angaben:
 - Ausgabeordner für materialisierte Seiten
 - Zu verwendende Seitenvorlagen
- eventuell eine Konfigurationsdatei für die im vorigen Punkt erwähnten Einstellungen
- korrekte Seitenvorlage im angegebenen Vorlagenordner

4.2. Ausgaben

4.2.1. Material-Organisation

- Leeres Manifest in der gewünschten Version der IMS Content Packaging Specification im Basisordner
- Ordnerstruktur unterhalb des Basisordners
- Importiertes Lernmaterial mit Abhängigkeiten in der Ordnerstruktur
- interne Vorschau der gewählten Dateien
- Öffnen der assoziierten Applikation mit dafür gewählten Dateien

4.2.2. Lernpaket aufbauen und bearbeiten

4.2.3. Metadatenerstellung und -bearbeitung

4.2.4. Lernpaketexport

Der Export als ZIP-Archiv erstellt eine ZIP-Datei, welche den gesamten Inhalt des Basisordners mit dem Manifest und den benötigten Schemata enthält. Diese Datei wird im gewählten Ausgabeverzeichnis gespeichert.

Beim Erstellen einer Offline-Version des Lernpakets wird im gewählten Ausgabeverzeichnis eine materialisierte Sicht auf die Ressourcen erstellt. Diese Materialisierung enthält die Ressourcen und eine Navigationskomponente, welche das Betrachten des Lernpakets mit seinen Ressourcen in einem Browser ermöglicht. Ebenso werden die Metadaten des Manifests dargestellt und Hilfen zur Kursnavigation wie beispielsweise eine Roadmap gegeben.