

Einführung in Topic Maps

Adi Bekier

Kohlrauschweg 19
60486 Frankfurt
Adi.Bekier@web.de

Abstract. Wenn man unterschiedliche Informationen miteinander verschmelzen möchte, benötigt man eine einheitliche Syntax. Ein anderes Problem besteht darin, dass wild zusammen gewürfelte Informationssammlungen schnell den Überblick vermissen lassen. Mit *Topic Maps* kann man ein übergeordnetes Modell entwerfen und Informationen mischen, die sich sonst nicht vermischen lassen würden und kann das ganze immer noch übersichtlich halten. *Topic Maps* lassen sich auf jedem Gebiet anwenden. Sie dienen der Strukturierung von Wissen aus Sicht des Menschen.

1. Einführung

Wozu benötigt man eine Idee wie Topic Maps überhaupt?

Um es ganz grob zu erklären, stellt man sich einfach ein Buch vor. Jedes wissenschaftliche Buch, das etwas auf sich hält, besitzt auch einen Index. Nun *Topic Maps* sind so etwas wie ein Index, nur dass sie komplexer sind und wesentlich mehr können.

Nehmen wir als Beispiel den genannten Index aus einem Buch. Wenn man zwei Bücher verschmelzen und zusätzlich noch eine große Menge eMails und eine Datenbank voller Informationen integrieren möchte, wird es sehr kompliziert. Jetzt bräuchte man einen übergeordneten Zusammenhang um da noch durchblicken zu können. So etwas könnte man mit *Topic Maps* hinkriegen. *Topic Maps* definieren die Beziehungen der Indizes zueinander und auch die Verweise auf die eigentlichen Informationen.

Zu jeder großen Ausarbeitung gehört auch eine Literaturrecherche. Man sucht sich Inhalte aus verschiedenen externen Quellen (Bücher, Internet) zusammen und bringt sie noch zu seinem eigenen Dokument in Verbindung. Hier wird es schwer nachher den Überblick zu behalten oder anschließend die einzelnen Textpassagen deren Quellen zuzuordnen. Dieses Problem kann man auch mit *Topic Maps* lösen wie man nachher sieht.

In einer *Topic Map* konzentriert man sich weniger auf den Inhalt der Informationen des Themas, als mehr darauf wovon die Informationen handeln und die Beziehungen in der das Thema zu anderen Themen stehen. Man kann so die Abhängigkeiten und Relationen zwischen Gegenständen oder Sachverhalten zeigen, ohne allzu sehr ins Detail gehen zu müssen und den Überblick nicht zu verlieren. Manchmal kann das den Ausschlag geben, um in einer Informationsflut den Durchblick zu behalten und die Informationen für den Menschen bearbeitbar zu halten.

Und das ist wiederum das Entscheidende: *Topic Maps* sind im Wesentlichen für Menschen gefertigt. Sie sollen die Navigation oder Suche in einem großen Dokument erleichtern und zwar für Menschen, denn Computer haben dank ausgeklügelter Algorithmen und Programme sicherlich keine Probleme damit.

Man kann also zusammenfassend sagen, dass eine *Topic Map* drei grundlegende Funktionen erfüllt:

1. Sie bietet die Möglichkeit die Zugänglichkeit von Informationen zu erweitern.
2. Sie bietet flexible, erweiterbare Datenstruktur für Applikationen.
3. Sie bietet eine Integrationsschicht zwischen bestehenden Applikationen.

Ich werde nun im zweiten Kapitel etwas zur Entstehung und Geschichte von *Topic Maps* erzählen. Im dritten Kapitel werde ich den Aufbau und die Syntax von *Topic Maps* beschreiben und präzise auf die einzelnen Bestandteile einer *Topic Map* eingehen, bevor ich im vierten Kapitel eine Zusammenfassung gebe.

2. Entstehung von Topic Maps

Topic Maps haben ihre ursprünglichen Wurzeln im Jahre 1990. Damals schon versuchte Steve Newcomb Indizes, Glossare und Thesauri zusammenzuführen. Man sah den Nutzen von einer Art globalem Index, denn dieses Problem besteht schon lange Zeit. Es gibt so viele Dinge, die unter den Menschen verschiedene Namen haben. Wie praktisch wäre es, wenn man die verschiedenen Namen auf ein und dieselbe Sache referenzieren könnte, ohne den Überblick zu verlieren?

Der allererste Standard von *Topic Maps* entstand im Jahr 2000 (ISO Standard 13250:2000). Dieser baute damals noch auf der Architektur von *SGML* ([Standard Generalized Markup Language](#)) auf und benutzte zum Verlinken [Hytime](#), welches ein älteres Linking-Modul aufbauend auf *SGML* war.

Bald darauf wurde eine Gruppe geformt, die es sich zur Aufgabe machte *Topic Maps* in *XML* ([Extensible Markup Language](#)) zu formulieren, anstatt in *SGML*. Also entstand im Jahr 2001 die *XTM* ([XML Topic Maps](#)) Spezifikation. Diese Version baut auf dem ISO-Standard auf. Außerdem nutzt sie auch das [Xlink-Modell](#) zum Verlinken.

3. Aufbau und Syntax von Topic Maps

Topics Maps <topicMap>

Topic Maps bauen auf *XML* auf. Man baut Containerelemente, die mit einem <topicMap> geöffnet und mit einem </topicMap> wieder geschlossen werden, auf. Im folgenden Beispiel sieht man das Hauptelement <topicMap> welches die *Topic Map* darstellt. Dieses Element führt auch den [XTM 1.0-Namespace](#) sowie den [Xlink-Namespace](#) ein. *Namespace* (dt. Namensraum) ist der Kontext der eindeutigen Zuordnung eines Namens zu einem Objekt. Auf *Topic Maps* bezogen definiert es die

Variablen und deren Namen (man muss sich nicht drum sorgen, das die Variablen mit schon bestehenden Variablen konfrontieren) und die Art der Verlinkung.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE topicMap SYSTEM "xhtml.dtd">
<topicMap
xmlns="http://www.topicmaps.org/xtm/1.0/"
xmlns:xlink="http://www.w3.org/1999/xlink">
</topicMap>
```

Topic <topic> (dt. Thema oder Gebiet)

Topics können alles sein, eine Person, ein Land oder die *Webseite der Universität Frankfurt* (<http://www.uni-frankfurt.de>). *Topics* sind die grundlegende Essenz in den *Topic Maps*.

Hier sieht man das Beispiel von oben, in welches ein *Topic*-Element eingebunden wird, nämlich <topic id>.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE topicMap SYSTEM "xhtml.dtd">
<topicMap
xmlns="http://www.topicmaps.org/xtm/1.0/"
xmlns:xlink="http://www.w3.org/1999/xlink">
<topic id="Webseite der Universität Frankfurt"/>
</topicMap>
```

Subject Identifier <subjectIdentity> (dt. Identifikation)

Subject Identifier (auch *Subject Indicators* genannt) dienen dazu Themen zueinander zuzuordnen. Man kann sie sich als eine Art Postleitzahlen oder Matrikelnummern (jedem Subjekt fest zugewiesene Identifikationsnummern) für Themen vorstellen. Sie unterscheiden ebenfalls ob unterschiedlich Bezeichnungen dasselbe Thema beschreiben.

Am folgenden Beispiel sieht man, dass mit <subjectIdentity> das *Thema* eine eindeutige *Identifikation* erhält. Einmal handelt es sich hierbei um die *Webseite der Universität Frankfurt* (<topic id="Webseite der Universität Frankfurt"> verweist mit <resourceRef> auf die Adresse), ein andermal handelt es sich um die *Einrichtung "Universität Frankfurt"* deren Adresse „<http://www.uni-frankfurt.de>“ lautet (<topic id="Webseite der Universität Frankfurt"> verweist mit <subjectIndicatorRef> auf die Adresse). In Wirklichkeit lautet die Adresse der Universität natürlich nicht „<http://www.uni-frankfurt.de>“, sondern es die Postadresse, aber für dieses Beispiel wurde das vernachlässigt.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE topicMap SYSTEM "xhtml.dtd">
<topicMap xmlns=http://www.topicmaps.org/xtm/1.0/
```

```

xmlns:xlink="http://www.w3.org/1999/xlink">
<topic id="Webseite der Universität Frankfurt">
<subjectIdentity>
<resourceRef xlink:href="http://www.uni-frankfurt.de"/>
</subjectIdentity>
</topic>
<topic id="Einrichtung Universität Frankfurt">
<subjectIdentity>
<subjectIndicatorRef xlink:href="http://www.uni-
frankfurt.de"/>
</subjectIdentity>
</topic>
</topicMap>

```

Associations <association> (dt. Beziehung)

Themen können *Beziehungen* zueinander haben. Dabei wird nicht gesagt um was für eine Art der Relation es sich handelt (Symmetrie, Transitivität, Reflexivität). Es können mehrere *Themen* an einer *Beziehung* teilhaben.

Im folgenden Beispiel benutze ich Prof. Zicari als *Thema*. In dieser *Topic Map* ist Prof. Zicari ein "Member" der Universität Frankfurt. Hier wird mit <association> eine Beziehung gezeigt. Deren Art wird mit <member> spezifiziert, indem mit <topicRef> gezeigt wird zu welchem *Thema* die *Beziehung* besteht. In diesem Beispiel wird sogar auf die Art des *Members* eingegangen, indem man sagt wer von beiden das "Individual" Zicari und welches die "University" Universität Frankfurt ist.

```

<?xml version="1.0" encoding="UTF-8"?>
<topicMap xmlns="http://www.topicmaps.org/xtm/1.0/">
xmlns:xlink="http://www.w3.org/1999/xlink">
<topic id="member-of">...</topic>
<topic id="individual">...</topic>
<topic id="university">...</topic>
<topic id="Zicari">...</topic>
<topic id="Einrichtung Universität Frank-
furt">..</topic>
<association>
<instanceOf>
<topicRef xlink:href="#member-of"/>
</instanceOf>
<member>
<roleSpec><topicRef
xlink:href="#university"/></roleSpec>
<topicRef xlink:href="Universität Frankfurt"/>
</member>
<member>
<roleSpec><topicRef
xlink:href="#individual"/></roleSpec>
<topicRef xlink:href="Zicari"/>
</member>
</association>
</topicMap>

```

Occurrences <occurrence> (Verknüpfung)

Occurrences stellen eine *Verknüpfung* auf externe Webressourcen oder Dokumente dar. Eine *Verknüpfung* kann z.B. ein Verweis auf ein Tondokument (wie eine MP3) sein. Es könnte aber auch einen Link auf seine Webseite darstellen. Grob gesagt stellt eine *Occurrence* einen Verweis auf eine Quelle dar, an der man mehr Informationen über das Thema erhält.

In unserem Beispiel mit der Universität erhält das Thema *Universität* eine *Verknüpfung* mit einem Verweis auf ihre Webseite, wo man weitere Informationen vorfindet, die die Universität beschreiben. Diese *Verknüpfung* wird mit <occurrence> eingeleitet. Vor allem aber wird die Verlinkung mit <topicRef> hergestellt, welches in einem <instanceOf>-Element liegt.

```
<?xml version="1.0" encoding="UTF-8"?>
<topicMap xmlns="http://www.topicmaps.org/xtm/1.0/"
xmlns:xlink="http://www.w3.org/1999/xlink">
<topic id="homepage"></topic>
<topic id="Einrichtung Universität Frankfurt">
<subjectIdentity>
<subjectIndicatorRef xlink:href="http://www.uni-
frankfurt.de"/>
</subjectIdentity>
<occurrence>
<instanceOf><topicRef
xlink:href="#homepage"/></instanceOf>
<resourceRef xlink:href="http://www.uni-frankfurt.de"/>
</occurrence>
</topic>
</topicMap>
```

Scope <scope> (Gültigkeitsbereich)

Wenn man *Themen* eindeutige Namen vergibt, kann es durchaus vorkommen, dass es zwei unterschiedliche *Themen* mit gleichen Namen gibt (z.B. gleicher Vorname und Nachname) gibt. Also wurden so genannte *Scopes* erfunden. *Scopes* sind *Gültigkeitsbereiche* in denen *Themen* einmalig sind.

Gültigkeitsbereiche kann man z.B. dazu benutzen um Quellen für Anfänger von Quellen für Fortgeschrittene oder Profis zu unterscheiden. Auch kann man zeitlich unterscheiden, denn manche Städte hatten z.B. durch die Jahrhunderte hindurch verschiedene Namen gehabt. *Gültigkeitsbereiche* machen also durchaus einen Sinn. Deswegen schaut man sich hierzu am besten ein Syntaxbeispiel an.

Im Folgenden sieht man einen *Gültigkeitsbereich* in der Anwendung und zwar wieder mit dem Beispiel der Universität. Es gibt einen deutschen und einen englischen Namen für „*Universität*“. Außerdem hat man einen *Gültigkeitsbereich*, in dem man den Aufruf der Seite mit Studieninformationen nur für Studenten einschränkt.

Wie man sieht, kann der *Gültigkeitsbereich* also sowohl auf die *Themen* selber (<baseName>), als auch auf *Verknüpfungen* und auf *Beziehungen* angewandt werden, indem man jedes Mal einen Nachfolger mit <instanceOf> definiert.

```
<!DOCTYPE topicMap SYSTEM "file:/home/kal/projects/xmljournal/xtml.dtd">
<topicMap xmlns="http://www.topicmaps.org/xtm/1.0/"
xmlns:xlink="http://www.w3.org/1999/xlink">
<!-- Topics for languages: en = English, de=German -->
<topic id="en"/>
<topic id="de"/>
<!-- Topics for user types -->
<topic id="student"/>
<!-- Topics for occurrence types -->
<topic id="Studieninformationen"/>
<topic id="Einrichtung Universität Frankfurt">
<baseName>
<scope>
<topicRef xlink:href="#en"/>
</scope>
<baseNameString>University of Frankfurt</baseNameString>
</baseName>
<baseName>
<scope>
<topicRef xlink:href="#de"/>
</scope>
<baseNameString>Universität Frankfurt</baseNameString>
</baseName>
<occurrence>
<instanceOf>
<topicRef xlink:href="#Studieninformationen"/>
</instanceOf>
<scope>
<topicRef xlink:href="#student"/>
</scope>
<resourceRef xlink:href="http://www.uni-frankfurt.de/zsb/studium.htm"/>
</occurrence>
</topic>
</topicMap>
```

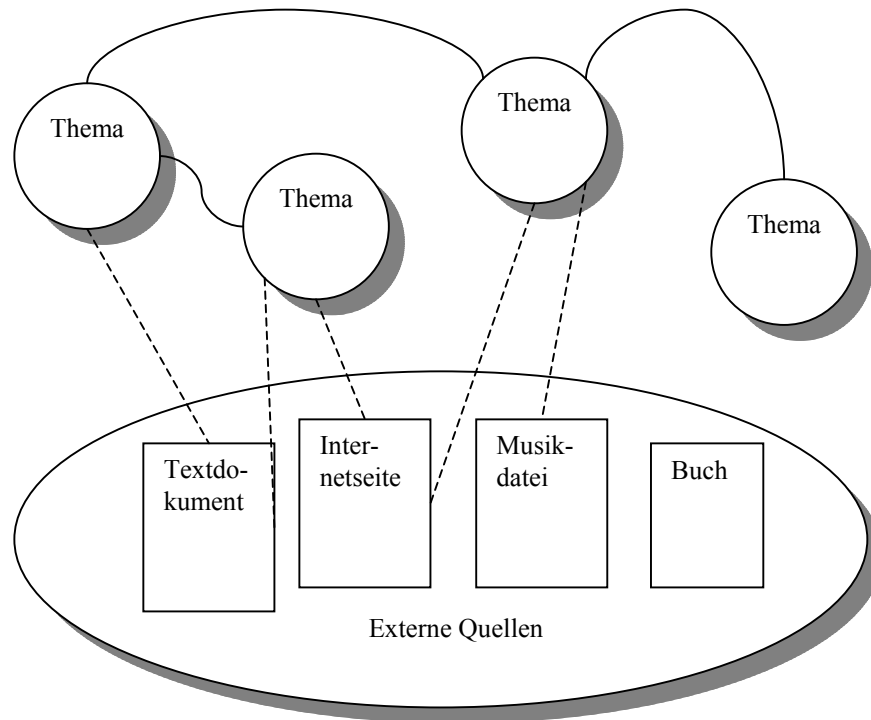


Fig. 1. Hier sieht man ein grafisches Beispiel für eine Topic Map. Die Kugeln sind die Themen, verbunden durch die Beziehungen (durchgehende Linien) und ihre Verknüpfungen auf externe Quellen, auch Occurrences genannt (gestrichelte Linien).

4. Zusammenfassung

Um es noch mal zusammenfassend aufzuzählen, eine *Topic Map* besteht aus *Topics* (Themen), *Subject Identifiers* (Identifikationen), *Associations* (Beziehungen), *Occurrences* (Verknüpfungen), und *Scopes* (Gültigkeitsbereiche).

Wir haben gesehen wie nützlich *Topic Maps* sein können um Wissen zu strukturieren. Dabei wurde gezeigt, dass dies alles mit der sehr verbreiteten XML-Syntax zu bewerkstelligen ist.

Das alles darf man aber noch nicht als „den“ Ansatz schlechthin betrachten. *Topic Maps* ist ein sehr neuer Standard ist, der sich erst noch auf breiter Front durchsetzen muss. Dabei muss man auch andere konkurrierende Standards wie z.B. RDF im Auge behalten. Wie sich dieses Gebiet entwickeln wird, kann nur die Zukunft zeigen.

Verwendete Literatur

1. Richard Widhalm und Thomas Mück, Springer (2002). Topic Maps
2. XML-Journal: <http://www.sys-con.com/xml/articleprint.cfm?id=507>
3. The TAO of Topic Maps: <http://www.ontopia.net/topicmaps/materials/tao.html>
4. EasyTopicMaps: <http://www.easytopicmaps.com/>