



Hochschule für Technik
und Wirtschaft Berlin

University of Applied Sciences

Fachbereich IV
Angewandte Informatik

Hausarbeit

über das Thema

Versuch einer *'lege artis' Analyse* der Software Device Registration

Hendrik Hofmann
Matr.Nr.: 539721
7.Semester
12. Dezember 2016

Prüfer:
Prof. Dr.-Ing. habil.
Dierk Langbein,
Onur Yavuz

I Inhaltsverzeichnis

1 Grundlagen	1
1.1 Der BMECat	1
1.1.1 Terminologie	1
1.1.2 Transaktionen	1
1.1.3 Aufbau	1
1.2 Informelle Aufgabenbeschreibung	8
1.3 Zielstellung	9

1 Grundlagen

1.1 Der BMECat

Der BMECat ist ein vom 'Bundesverband Materialwirtschaft, Einkauf und Logistik e.V.' in Zusammenarbeit mit dem 'eBusiness Standardization Committee' entwickelter XML Standard mit dem Ziel den Austausch von Produktkatalogen zwischen Lieferanten und beschaffenden Organisationen zu standardisieren und somit zu vereinfachen.¹.

1.1.1 Terminologie

Ein **Produktkatalog** ist die Menge aller benötigten Daten, welche vom katalogerzeugenden Unternehmen an das katalogempfangende Unternehmen übermittelt werden sollen.

Ein **Katalogdokument** ist eine XML-Datei, in der der Produktkatalog im BMECat-Format gespeichert und zum Katalogempfänger übermittelt wird.

Eine **Kataloggruppe** ist ein Datenbereich, der eine Gruppe definiert, welcher gleichartige Artikel zugeordnet werden können. Diese wird im BMecat-Format durch das Element **CATALOG_STRUCTURE** abgebildet.

Ein **Kataloggruppensystem** ist ein hierarchischer Baum von verknüpften Kataloggruppen. Es wird im BMecat-Format durch das Element **CATALOG_GROUP_SYSTEM** abgebildet.²

1.1.2 Transaktionen

Im BMECat wird zwischen den 3 verschiedenen Transaktionsarten

- **T_NEW_CATALOG** - Übertragung eines neuen Produktkataloges
- **T_UPDATE_PRODUCTS** - Aktualisierung von Produktdaten
- **T_UPDATE_PRICES** - Aktualisierung von Preisinformationen

unterschieden. Die Unterscheidung geschieht um die Größe eines Katalogdokumentes zu reduzieren. Es muss so z.B. nicht ein kompletter Produktkatalog übertragen werden, falls sich bei einem *odermehreren* Artikeln der Preis ändert.

1.1.3 Aufbau

Ein BMECat-Dokument besteht aus einer Folge von KANN und MUSS Feldern, den dazugehörigen Datentypen und Feldlängen und ist folgendermaßen aufgebaut:

¹BMECat V1.2 Spezifikation, Seite 5

²BMECat V1.2 Spezifikation, Seite 7

1. XML-Deklaration und Header-Bereich (mit Informationen über Kataloganbieter und Empfänger, Bezeichnung und Erstellungsdatum des Kataloges etc.)

Bsp. für einen Header:

```
<HEADER>
  <GENERATOR.INFO> Kann </GENERATOR.INFO>
  <CATALOG> Muss </CATALOG>
  <BUYER> Kann </BUYER>
  <SUPPLIER> Muss </SUPPLIER>
</HEADER>
```

Bsp. für XML Deklaration:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE BMECAT SYSTEM "bmecat_new_catalog.dtd">
<BMECAT version="1.2" xml:lang="de" xmlns="http://www.
bmecat.org/bmecat/1.2/bmecat_new_catalog">
```

2. Produktgruppensystem (Baumstruktur der Produktgruppen mit den Attributwerten Root, Node und Leaf)

```
<CATALOG.STRUCTURE type="root">
  <GROUP.ID>1</GROUP.ID>
  <GROUP.NAME>Katalog</GROUP.NAME>
  <PARENT.ID>0</PARENT.ID>
  <GROUP.ORDER>1</GROUP.ORDER>
</CATALOG.STRUCTURE>
<CATALOG.STRUCTURE type="node">
  <GROUP.ID>2</GROUP.ID>
  <GROUP.NAME>Spiele & Konsolen</GROUP.NAME>
  <PARENT.ID>1</PARENT.ID>
</CATALOG.STRUCTURE>
<CATALOG.STRUCTURE type="leaf">
  <GROUP.ID>7</GROUP.ID>
  <GROUP.NAME>PlayStation 4</GROUP.NAME>
  <PARENT.ID>2</PARENT.ID>
</CATALOG.STRUCTURE>
```

3. Artikel (mit Attributen und Werten)

```
<ARTICLE mode="new">
  <SUPPLIER.AID>9057320097280</SUPPLIER.AID>
  <ARTICLE.DETAILS>
    <DESCRIPTION.SHORT>GTA 5</DESCRIPTION.SHORT>
    <DESCRIPTION.LONG>Das tolle neue Spiel</DESCRIPTION.LONG>
    <EAN>87126723434</EAN>
    ... weitere Attribute ...
  </ARTICLE.DETAILS>
  ... weitere Felder ...
</ARTICLE>
```

4. Zuordnung der Artikel zu den Produktgruppen.

```

<ARTICLE_TO_CATALOGGROUP_MAP>
  <ART_ID>9057320097280</ART_ID>
  <CATALOG_GROUP_ID>7</CATALOG_GROUP_ID>
</ARTICLE_TO_CATALOGGROUP_MAP>

```

— Übersicht der im BMECat verwendeten Datentypen — noch einfügen —

Im folgenden Abschnitt wird jeder Teilbereich mit seinen Unterelementen, wie sie in vorliegender Arbeit verwandt wurden, graphisch dargestellt und kurz erläutert. Rot hervorgehoben sind jeweils die MUSS-Felder, welche zwingend in einem gültigen BMECat Dokument vorkommen müssen, grün die KANN-Felder. Ein Plus + Zeichen hinter dem Elementnamen indiziert, dass dieses Element mehrfach an dieser Stelle vorkommen kann, jedoch mindestens einmal. Ein Asterisk * zeigt an, dass dieses Element einmal, mehrfach oder nicht vorkommen kann. Das

Header

Im Header werden allgemeine Informationen über das Katalogdokument hinterlegt und Default Werte gesetzt. Das Element **CATALOG** enthält dabei Informationen zur Identifikation und Beschreibung des Produktkataloges, wie z.B. die Katalog Id, die Katalogversion oder die für das Dokument geltende Sprache sowie Elemente zum setzen von Standard-Werten wie z.B. die für das Katalogdokument geltende Währungsangabe ³

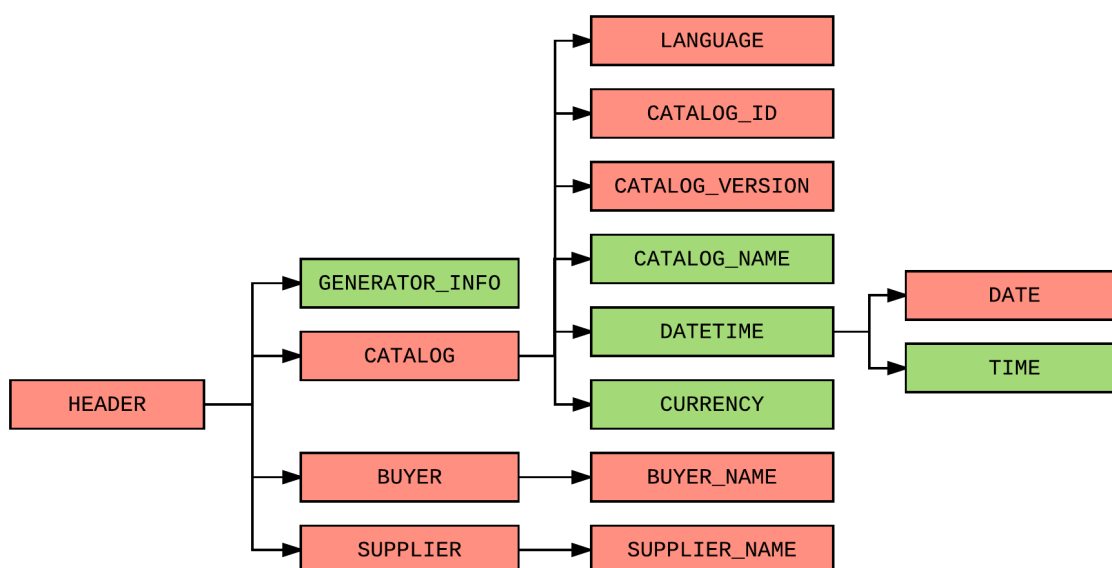


Abbildung 1: Headerstruktur

Die Transaktion T_NEW_CATALOG

³BMECat V 1.2 Spezifikation, Seite 27,29

Diese Transaktion wird verwendet, um einen Produktkatalog neu zu übertragen. Das empfangende System reagiert dabei je nach übertragener CATALOG_ID, CATALOG_VERSION und LANGUAGE unterschiedlich. Dieser Zusammenhang wird später noch erläutert.

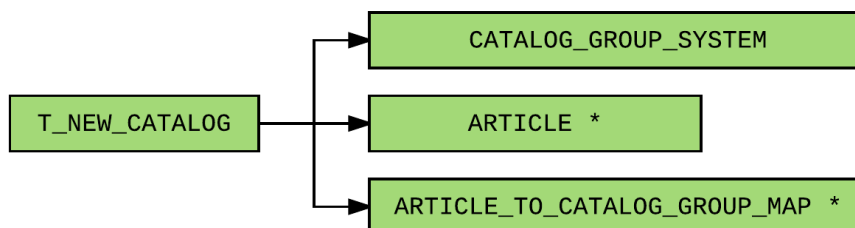


Abbildung 2: T_NEW_CATALOG

Die Transaktion T_UPDATE_PRODUCTS

Bei dieser Transaktion werden Artikeldaten übertragen und gegebenenfalls einer Kataloggruppe zugeordnet. Je nach Kennung des Artikels (s.u.) werden die übertragenen Artikel im Zielsystem entweder hinzugefügt, gelöscht oder die Artikeldaten werden komplett ersetzt. Der Artikel wird immer komplett ausgetauscht, eine Änderung von einzelnen Datenfeldern innerhalb eines Artikels ist nicht möglich. Wie der Grafik entnommen werden kann ist bei dieser Transaktion nur die Übertragung von Produktdaten und die Zuordnung von Produkten zu Kataloggruppen möglich.

⁴

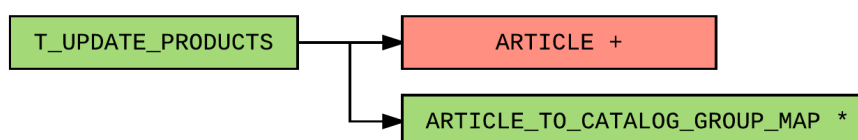


Abbildung 3: T_UPDATE_PRODUCTS

Das Element T_UPDATE_PRODUCTS verfügt zusätzlich über das Attribut prev_version, welches die Anzahl der vorausgegangenen Updates bzw. die Nummer des übertragenen Updates enthält.

```
<T_UPDATE_PRODUCTS prev_version="91" >...</T_UPDATE_PRODUCTS>
```

Die Elemente CATALOG_GROUP_SYSTEM und CATALOG_STRUCTURE

⁴vgl. BMECat V 1.2 Spezifikation, Seite 52

Im Element `CATALOG_GROUP_SYSTEM` werden die `GROUP_SYSTEM_ID` und der `GROUP_SYSTEM_NAME` bekannt gemacht sowie die Katalogstruktur `CATALOG_STRUCTURE` beschrieben. Dabei gibt es genau ein Wurzelement, sowie beliebig viele Knoten und Blätter. Jedes Element hat dabei eine als `GROUP_ID` bezeichnete ID und wird über `PARENT_ID` die dem jeweiligen Elternelement zugeordnet. Die Zuordnung der Artikel zu den Artikelgruppen erfolgt mit dem Element `ARTICLE_TO_CATALOG_GROUP_MAP` das weiter unten beschrieben wird.

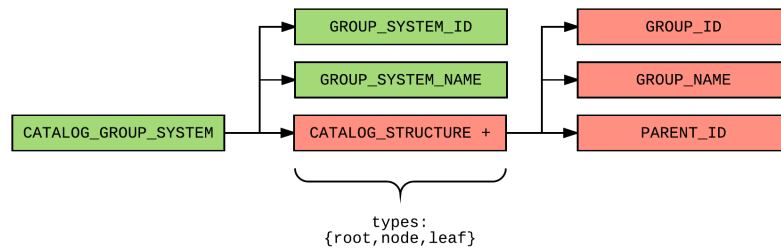


Abbildung 4: `CATALOG_GROUP_SYSTEM` und `CATALOG_STRUCTURE`

Das Element ARTICLE

Das Artikelelement schließlich enthält Informationen über einen Artikel, wie Überschrift, Beschreibung, Bilder, Preisinformationen, eine **eindeutige** Artikelnummer usw. Die Artikelnummer wird über das Element SUPPLIER_AID bekanntgegeben, handelt es sich um einen Variantenartikel, so bildet sich die Artikelnummer aus der SUPPLIER_AID und der SUPPLIER_AID_SUPPLEMENT. Dies ist hier jedoch nicht umgesetzt. Die als *eCl@ass* und *Zolltarifnummer* zusammengefassten ARTICLE_FEATURES werden explizit von Mercateo verlangt.

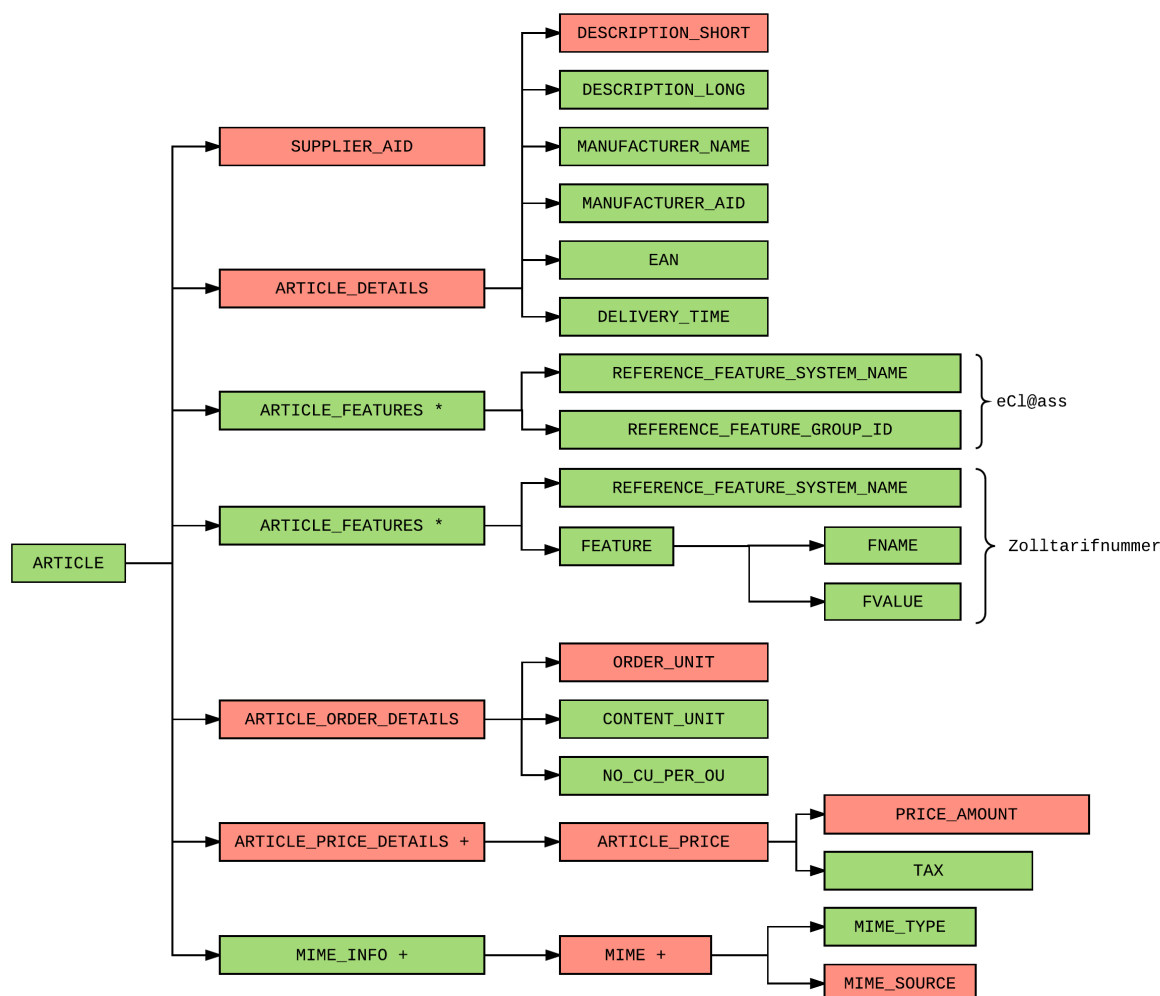


Abbildung 5: Article

Das Element ARTICLE verfügt über das Attribut mode, welches Informationen darüber enthält, ob es sich um die Anlage eines neuen Artikel, ein Update der Artikelinformationen oder die Löschung eines Artikels handelt.

```

<ARTICLE mode="new">...</ARTICLE>
<ARTICLE mode="update">...</ARTICLE>
<ARTICLE mode="delete">...</ARTICLE>
  
```


Das Element ARTICLE_TO_CATALOG_GROUP_MAP

Um Produkte ihren Kategorien zuordnen zu können wird das Element `ARTICLE_TO_CATALOGGROUP_MAP` verwandt. Es erfolgt hier eine Verknüpfung aus der eindeutigen Artikelnummer und der `GROUP_ID` welcher der Artikel zugeordnet werden soll. Eine Mehrfachzuordnung ist möglich, d.h. ein Artikel kann in unterschiedliche Kategorien „eingehängt“ werden.

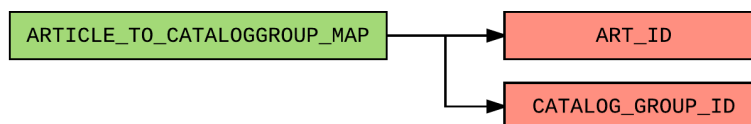


Abbildung 6: `ARTICLE_TO_CATALOG_GROUP_MAP`

Im Kontext der Transaktion `T_UPDATE_PRODUCTS` verfügt das Element zusätzlich über das Attribut `mode`, mit welchem angegeben wird, ob es sich um eine Neuzuweisung zu einer Kategorie handelt oder der Artikel aus einer Kategorie entfernt werden soll.

```
<ARTICLE_TO_CATALOGGROUP_MAP mode="new">...</ARTICLE_TO_CATALOGGROUP_MAP>  
<ARTICLE_TO_CATALOGGROUP_MAP mode="delete">...</ARTICLE_TO_CATALOGGROUP_MAP>
```

Zusammenspiel verschiedener Transaktionen

Die folgende Grafik zeigt, wie das empfangende System bei der Transaktion `T_NEW_CATALOG` je nach übergebener `CATALOG_ID`, `CATALOG_VERSION` und `LANGUAGE` reagiert.

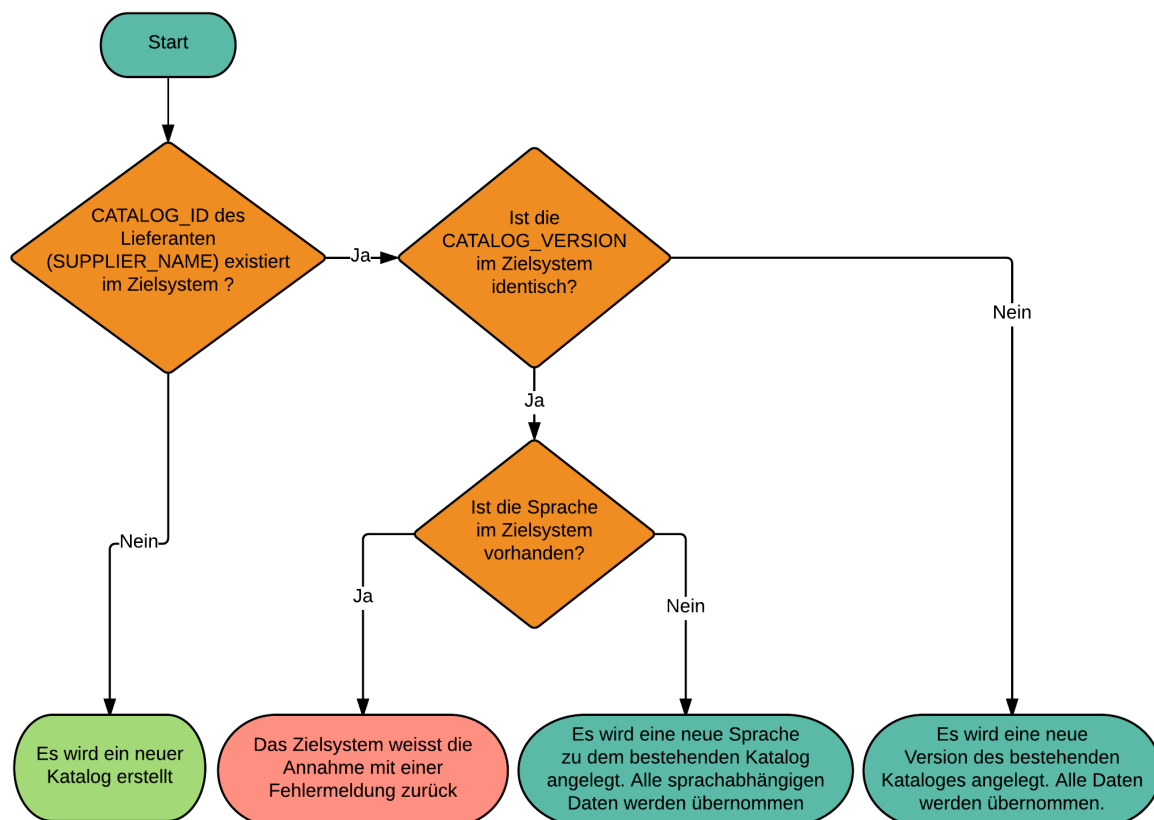


Abbildung 7: Flowchart T_NEW_CATALOG

Kommt die Transaktion T.UPDATE_PRODUCTS zur Anwendung, gilt es folgendes zu beachten:⁵

- Die übertragene CATALOG_ID des jeweiligen Lieferanten und die dazugehörige CATALOG_VERSION müssen im Zielsystem bereits vorhanden sein.
- Das Attribut prev_version muss bei der ersten anderen Transaktionsart nach T_NEW_CATALOG, (T.UPDATE_PRODUCTS, T.UPDATE_PRICES) auf '0' gesetzt werden.
- Danach wird es bei jeder solchen Transaktion um '1' erhöht.

1.2 Informelle Aufgabenbeschreibung

Ziel der Arbeit ist es die von der Software iTool aus verwaltbaren, in verschiedenen Tabellen einer SQL-Datenbank gehaltenen Produkt-, Katalog- Kategorie- und Herstellerdaten in ein von Mercateo verarbeitbares Format (dem BMECat) zu bringen. Dabei gilt es, den Anforderungen der Spezifikationen sowohl das BMECat, als auch der besonderen Anforderungen seitens Mercateo zu genügen. Es soll möglich sein, die erwähnten Daten aus dem UI des iTool her-

⁵vgl. hierzu BMECat V 1.2 Spezifikation, Seite 52

aus nach dem CRUD-Prinzip zu bearbeiten. Die eigentliche Erstellung der unterschiedlichen Kataloge (neuer Katalog bzw. Produktupdatekatalog) erfolgt dabei (automatisiert) über die CakePHP Shell. Kataloge können dabei für unterschiedliche Verkäufer erstellt werden. Zudem soll es Mercateo ermöglicht werden Bestandsdaten zu einer bestimmten Artikelnummer über ein Webinterface abzurufen.

1.3 Zielstellung

Folgende Funktionalitäten sollen implementiert werden:

- Die in iTool hinterlegten Produkt- bzw. Herstellerdaten sollen in ein gültiges und vollständiges BMECat Dokument entsprechend der Mercateo Anforderungen überführt werden. Dabei ist insbesondere auf die Unterschiede und Besonderheiten der notwendigen beiden Transaktionsarten "T_NEW_CATALOG also die Erstellung eines neuen Kataloges - und "T_UPDATE_PRODUCTS also der Änderungen von Produktdaten, sowie dem löschen und neu erstellen von Produkten - zu achten .
 - Gültig bedeutet in diesem Fall, dass Struktur und Inhalt des Dokuments fehlerfrei gegen die entsprechende XSD Datei laufen, d.h. die Felder müssen in der richtigen Reihenfolge unter Beachtung der Datentypen und Längenbegrenzungen sowie Formatlimitierungen (z.B. keine Sonderzeichen in der SKU (o.ä.)) geschrieben werden.
 - Vollständig heißt, dass zum einen mindestens jene Felder im BMECat Dokument vorkommen, die die BMECat Spezifikation verlangt. Zusätzlich müssen jene Felder vorkommen, die die Mercateo Spezifikation erfordert und zwar unter zusätzlicher Beachtung der Limitierungen bzw. Besonderheiten jener Spezifikation.
- Die Produktkategoriestruktur des iTool soll in das Kataloggruppensystem des BMECat überführt werden.
- die Implementierung der Katalogerstellungslogik erfolgt in einer Cake Shell. Liegt noch kein Katalog vor, wird ein neuer Katalog erstellt; ein Updatekatalog wird erstellt, wenn es Änderungen bei den Produktdaten gab.
- Kataloge können für unterschiedliche Verkäufer erstellt werden.
- Die Produkt und Katalogdaten können über die Benutzeroberfläche des iTool eingesehen bzw. verändert werden.
- Es soll Mercateo ermöglicht werden Bestandsdaten zu den im Katalog vermerkten Produkten über ein Webinterface abzurufen.
- Bei der Katalogerstellung ist darauf zu achten, dass es zu keinen Arbeitsspeicherüberläufen kommen kann.

1. Fowler, Martin ; Beck, Kent ; Brant, John ; Opdyke, William ; Roberts, Don ; Gamma, Erich: Refactoring : Improving the Design of Existing Code. 1. Aufl.. Amsterdam: Addison-Wesley, 2012.
2. Martin, Robert C.: Clean Code - Refactoring, Patterns, Testen und Techniken für sauberen Code : Deutsche Ausgabe. 1. Aufl.. Heidelberg: MITP-Verlags GmbH & Co. KG, 2013.
3. Hauer, Phillip ; philliphauer.de; <http://www.philippbauer.de/study/se/design-pattern.php>; Abgerufen am 29.3.2016;