

# NetCommunity.Discussion

Abschlußbericht zum Projekt

**„Evaluierung von Bulletin Board – Software“**

im Softwarelabor der Fachhochschule Heilbronn

erarbeitet von

Name:	Jan Niggemann
Straße:	Max – Planck – Str. 37 UG 34
PLZ / Ort:	74081 Heilbronn
Telefon:	07131 / 56 92 19
Handy:	0177 / 39 55 828
e-Mail:	jan.niggemann@swl.fh-heilbronn.de
Homepage:	<a href="http://linux.fh-heilbronn.de/~nigge">http://linux.fh-heilbronn.de/~nigge</a>

GEGENSTAND DES PROJEKTES .....	3
BEWERTUNGSKRITERIEN.....	3
<i>Funktionalität</i> .....	3
<i>Zuverlässigkeit</i> .....	3
<i>Benutzbarkeit</i> .....	3
<i>Effizienz</i> .....	3
<i>Übertragbarkeit</i> .....	3
VORBEREITUNGEN / INSTALLATION .....	4
<i>Installation des Betriebssystems</i> .....	4
<i>Installation des Webservers</i> .....	4
<i>Installation der JRE</i> .....	4
<i>Installation des JSDK</i> .....	4
<i>Installation der Datenbank</i> .....	4
<i>Installation des JDBC Treibers</i> .....	4
<i>Installation der Servlet-Engine</i> .....	5
<i>Installation der NetCommunity</i> .....	5
PROBLEME BEI INSTALLATION UND BETRIEB .....	6
<i>Schlechtes Handbuch</i> .....	6
<i>Änderungen an config-files</i> .....	6
<i>Laufstabilität</i> .....	6
GENERELLE BETRACHTUNG DER SOFTWARE.....	7
<i>Konzept und Ziele der Entwickler</i> .....	7
<i>Realisierung als Servlets</i> .....	7
<i>Anlegen von Benutzern</i> .....	7
<i>Anlegen von Benutzergruppen</i> .....	7
<i>LDAP – Anbindung</i> .....	7
<i>postinstallativer administrativer Aufwand</i> .....	7
<i>Abschließende Betrachtung</i> .....	7
<i>Abschließende Bewertung</i> .....	8
<i>Erklärungen zu den einzelnen Punkten</i> .....	8
<b>ANHÄNGE</b> .....	<b>10</b>
BESCHREIBUNG DES TESTRECHNERS .....	10
BESCHREIBUNG DES PROTOTYPEN.....	10
<i>Benutzergruppen</i> .....	10
<i>Nicknames</i> .....	10
<i>Definition der Benutzergruppen</i> .....	10
<i>Webserver</i> .....	10
LITERATUR .....	10
<i>Endnoten</i> .....	11

## **Gegenstand des Projektes**

Gegenstand des Projektes war die Evaluierung des Softwarepakets ‚NetCommunity‘ von Cassiopeia<sup>1</sup>. Geprüft werden sollte, ob das Modul ‚Discussion‘ besser als andere Programme das Ziel erfüllt, die Funktionalität eines „Diskussionsbereichs“, engl. „bulletin – board“ bereitzustellen.

## **Bewertungskriterien**

Bei der Bewertung der Software wurde auf folgende Punkte besonderen Wert gelegt:

### **Funktionalität**

Angemessenheit	Ist die Software in der Lage, alle benötigten Funktionen zu erfüllen?
Richtigkeit	Funktioniert die Software erwartungsgemäß, oder gibt es noch Bugs?
Sicherheit	Entsteht durch die Software z.B. eine Sicherheitslücke?

### **Zuverlässigkeit**

Reife	Sind Funktionen und Objekte sorgfältig programmiert?
Fehlertoleranz	Stürzt die Software gleich bei kleinen Bedienungsfehlern ab?
Wiederherstellbarkeit	Ist es möglich, nach einem Systemcrash den Zustand schnell wieder zu restaurieren?

### **Benutzbarkeit**

Verständlichkeit	Ist die Software intuitiv?
Erlernbarkeit	Wie schnell lernen Benutzer und / oder Administrator den Umgang mit der SW?
Bedienbarkeit	Ist die Oberfläche gelungen, oder muß man sich umständlich durch Menüs klicken oder tippen?

### **Effizienz**

Zeitverhalten	Wie lang ist die Reaktionszeit auf einzelne Events?
Verbrauchsverhalten	Wie ressourcenhungrig ist die Software?

### **Übertragbarkeit**

Anpaßbarkeit	Werden zur Anpassung an die Umgebung Templates benutzt?
Installierbarkeit	Wie einfach gestaltet sich der Installationsvorgang?

## **Vorbereitungen / Installation**

Nach einem Anruf bei Cassiopeia wurde per e-Mail eine 30-Tage Testlizenz für ‚NetCommunity‘ geschickt, welche die Benutzung der Software in keiner Weise einschränkte.

Nun waren folgende Schritte nötig, um mit der Evaluierung zu beginnen:

### **Installation des Betriebssystems**

Als Betriebssystem kam RedHat<sup>2</sup> 6.2 zum Einsatz. Benutzt wurde eine leicht veränderte Standard – Installation, erweitert um einige Patches und Updates.

### **Installation des Webservers**

Als Webserver kam die freie Software Apache<sup>3</sup> in der Version 1.3.12 zum Einsatz. Diese Version ist aktuell als Produktionsversion empfohlen. Nach dem Entpacken der Quellen wurde ein dynamischer Webserver (DSO) mit beinahe allen Modulen kompiliert.

### **Installation der JRE**

Um NetCommunity zu installieren wurde zunächst die benötigte Umgebung geschaffen. Hier kamen die IBM JRE 1.1.8 und das IBM SDK 1.1.8.<sup>4</sup> zum Einsatz. Beide lagen als rpm<sup>5</sup> vor, ein kompilieren o.ä. war nicht nötig. Der Befehl ‚rpm –hiv IBM\*‘ reichte aus, um beide Pakete ordnungsgemäß zu installieren.

### **Installation des JSDK**

Das benötigte JSDK konnte erst nach langer Suche auf Suns<sup>6</sup> unübersichtlichen Web – Seiten gefunden werden. Bis Heute existiert nur eine Solaris – Version (Version 2.0), die aber auch unter Linux läuft. Hier reichte es aus, das JSDK zu entpacken und zu installieren.

### **Installation der Datenbank**

Als Datenbank kam die relationale SQL-Datenbank MySQL<sup>7</sup> zur Anwendung, die unter UNIXen lizenzfrei genutzt werden kann. Die aktuell als Produktionsversion empfohlene Version ist hier 3.22.32 und lag als (Pseudo-) Source – RPM vor, das dann jedoch leider nur ein tar –Archiv enthielt. Nach dem Entpacken des Archivs reichte ein ‚./configure‘ um das System automatisch zu konfigurieren. Das System wurde in allen Punkten automatisch richtig erkannt. Das anschließende ‚make‘ kompilierte das Programm. Als Compiler kam hier der egcs 1.1.2-30 zum Einsatz. Ein anschließendes ‚make install‘ installierte das fertig kompilierte Programm, ein shell - Skript legte die initial benötigten Tabellen an.

### **Installation des JDBC Treibers**

Hier wurde ein GJT<sup>8</sup> - JDBC – Treiber benutzt. Nach dem Download reichte es aus, das Archiv zu entpacken und ‚make all‘ zu benutzen. Da sich das JDK nach /usr/jdk118 installiert hatte, das Makefile des JDBC Treibers aber mehrfach /usr/local/jdk118 benutzte, wurde ein Symlink gemacht. Das nach dem kompilieren entstandene Verzeichnis mm.mysql.jdbc wurde dann noch nach /usr/local geschoben, damit der FHS<sup>9</sup> (Filesystem Hierarchy Standard) erfüllt ist. Der von Cassiopeia in der Dokumentation vorgeschlagene Treiber kam nicht in Frage, da er veraltet ist und einer restriktiven Lizenz unterliegt (Vgl. auch „Probleme bei der Installation“, Seite 6)

## Installation der Servlet-Engine

Als Servlet-Engine wurde die aktuell zur Produktion freigegebene Version JServ 1.1<sup>10</sup> gewählt. Nach dem entpacken des tar - Archivs wurde JServ dem Online - Handbuch entsprechend (mit den benötigten Pfadnamen, sowie `-with-apxs`) konfiguriert und kompiliert.

## Installation der NetCommunity

Kurze Zeit nach dem Telefonat mit Cassiopeia wurde die erforderliche Lizenz und das gesamte (!) Softwarepaket ‚NetCommunity‘ per e-Mail geschickt. Obwohl bereits in einem Fax das Betriebssystem (Linux) angegeben worden war, war die Software in einem ZIP-Archiv gepackt, obwohl man unter UNIX eigentlich ein tar.gz erwartet hätte. Ein kompatibler Entpacker dekomprimierte und entpackte dann das Archiv, das NetCommunity 8.2 enthielt. Die entstehende Verzeichnisstruktur schob ich dann in ein Unterverzeichnis des document-root des Webservers:

```
/www                document-root
/www/servlets/cassiopeia  Verzeichnisse und Dateien der NetCommunity
```

Nach der Konfiguration und der Registrierung der Servlets bei JServ sollte NetCommunity eigentlich laufen. Leider gab es hier durch die unzureichende Dokumentation viele Probleme, so daß erst nach vielen Tagen Versuche unternommen werden konnten, daß Programm zu starten.

Es wird hier dringend empfohlen, die von Cassiopeia empfohlenen Pfadnamen, Zonen-Namen usw. zu benutzen, da sich ansonsten die Installation als extrem schwierig gestaltet (vgl. „Probleme bei der Installation“, „Änderungen an config-files“). Auch in der Betriebsphase bei z.B. einem Wechsel des zuständigen Administrators o.ä. kann er / sie sich evtl. schneller im Handbuch zurecht finden, wenn die darin benutzten Namen mit der Installation übereinstimmen.

## Probleme bei Installation und Betrieb

### Schlechtes Handbuch

Die Dokumentation zum Kern der NetCommunity, von Cassiopeia als „Modul Core“ bezeichnet, die in der Version vom 1. September 1999 vorlag, war stellenweise exorbitant veraltet: Hier einige Beispiele:

- Im Kapitel „Installation“, Unterkapitel „JDBC – Treiber“ (Kapitel 5.1.4, Seite 14) wird ein JDBC - Treiber für MySQL empfohlen. Die URL, die hier angegeben ist, ist seit 1998 unverändert. Daraus ist zu schließen, daß der Treiber nicht mehr weiter entwickelt wird. Am wichtigsten aber ist, daß Cassiopeia anscheinend ignoriert, daß die kommerzielle Nutzung des Treibers eine Registrierung erfordert, verbunden mit dem Kauf einer Lizenz. Ein Hinweis auf eventuell entstehende Folgekosten fehlt in der Dokumentation, obwohl dies nicht nur finanzielle, sondern auch rechtliche Schritte zur Folge haben kann.  
Der beschriebene Treiber kam wegen seiner restriktiven Lizenz, des fehlenden Supports und der eventuell entstehenden Folgekosten nicht zur Benutzung in Frage. Wie sich später erst zeigte wäre die Benutzung diese Treibers jedoch verbindlich gewesen. Siehe auch: „Änderungen an config-files“
- Im Kapitel „Installation“, Unterkapitel „Registrieren der Servlets“ (Kapitel 5.2.2, Seite 18) wird beschrieben, wie die Registrierung durchgeführt wird. Die aktuelle Version von JServ 1.1 wird leider nicht behandelt.
- Im Kapitel „Installation“, Unterkapitel „Starten der NetCommunity“ (Kapitel 5.3, Seite 24 ff.) wird (noch einmal) beschrieben, was alles notwendig ist, um NetCommunity zu starten. Hier wird nur die schon damals veraltete Version JServ 0.9.11 behandelt, sogar über die damals aktuelle Version 1.0 wird hier nichts gesagt, sondern nur eine URL angegeben, deren Dokumente noch dazu in englischer Sprache abgefaßt sind.

Das Handbuch, daß leider nicht als gedruckte Version existiert, hat noch weitere Schwachstellen, auf die hier nicht näher eingegangen wird, da deren Betrachtung für das Projekt nicht weiter wichtig ist.

### Änderungen an config-files

- Um NetCommunity überhaupt starten zu können, bedurfte es eines detektivischen Spürsinn: Es war notwendig, das Core – Config – File von Hand anzupassen, denn die hier benutzten Voreinstellungen forderten eigentlich, daß der oben beschriebene, veraltete JDBC - Treiber hätte benutzt werden sollen, was aber aus den bereits aufgezeigten Gründen unmöglich zu verantworten war.  
Im Grunde wäre das Verändern des config-files per Hand kein Problem gewesen, wenn in der Dokumentation darauf hingewiesen worden wäre. Aufgrund des fehlenden Hinweises wurde jedoch viel Zeit mit der Suche nach dieser potentiellen Störquelle verschwendet. Erst das Ändern einiger Einträge in diesem File machte einen Start von NetCommunity möglich, auch, wenn Änderungen an diesem File eigentlich nicht gewollt sind. So steht in der ersten Zeile der betreffenden Datei:  
### DO NOT EDIT THIS FILE! ###

### Laufstabilität

Um auch alternative Programme zu testen wurde die Software ‚Sporum‘ von smallpig.net installiert. Die Software besteht aus CGI – Scripten und stellt ähnliche Funktionen wie NetCommunity bereit. Nach der gelungenen Installation von Sporum war die vorher funktionierende NetCommunity allerdings nicht mehr zur Mitarbeit zu bewegen. Eine intensive Fehlersuche, stundenlanges Logfile – lesen, Mail-Wechsel und Telefonate mit dem Cassiopeia-Support brachten doch nicht die erhoffte Lösung. NetCommunity funktionierte nicht mehr ordnungsgemäß – eine komplette Neuinstallation des Rechners war angesagt.

## **Generelle Betrachtung der Software**

### **Konzept und Ziele der Entwickler**

Das Konzept, welches NetCommunity zu Grunde liegt, wird von Cassiopeia selbst wie folgt beschrieben: „*NetCommunity ist ein Baukastensystem zur Personalisierung und Kundenbindung.*“<sup>11</sup>. Der Betreiber der von NetCommunity wird im Vorwort auch als „*Werber*“<sup>12</sup> bezeichnet. Damit ist NetCommunity eindeutig ein Instrument, das auf Marketing ausgerichtet ist. Ziel des Softwarepaketes es, kontinuierlich Daten über den Benutzer und seine Gewohnheiten zu sammeln, um daraus ein detailliertes Profil zu erstellen. Mit dieser Prämisse kann NetCommunity zwar ohne weiteres die Funktionalität eines „Diskussionsbereichs“ erfüllen, ist aber in keiner Weise darauf ausgerichtet oder gar spezialisiert, sondern eher überdimensioniert.

### **Realisierung als Servlets**

Die Realisierung der Software als Servlets ist sinnvoll, da große Mengen Daten verarbeitet werden. Servlets sind hier die schnellste und eleganteste Methode und daher auch besser als CGI - Scripte.

### **Anlegen von Benutzern**

Ein großes Problem für Marketingstrategen besteht darin, möglichst viele Profile zu bekommen. Bei NetCommunity ist es deshalb ausdrücklich erwünscht, daß sich Benutzer selber, d.h. ohne Eingriff des Administrators anmelden können:

Unter [http://esso.swl.fh-heilbronn.de/servlets/NetCommunity?service=register\\_form](http://esso.swl.fh-heilbronn.de/servlets/NetCommunity?service=register_form) könnte sich also ein beliebiger Benutzer eintragen. Im täglichen Betrieb in einer Produktionsumgebung sind daher Maßnahmen zu treffen, damit sich Benutzer nicht ohne einen Administrator anmelden können.

### **Anlegen von Benutzergruppen**

Bei NetCommunity werden Benutzergruppen durch BOOL'sche Ausdrücke gebildet. Dabei werden die einzelnen, bereits vordefinierten Felder der Benutzerdatenbank<sup>13</sup> mit selbst definierten Zusatzfeldern<sup>14</sup> benutzt.

Für die Übergangszeit bis zur Realisierung der LDAP – Anbindung wird empfohlen, ein eigens definiertes Zusatzfeld mit der Bezeichnung der Abteilung anzulegen, damit diese Information hinterher zur Bildung von Benutzergruppen verwendet werden kann.

### **LDAP – Anbindung**

Eine Anbindung an ein bestehendes LDAP – Directory wird laut Cassiopeia mit neuen Modulen im Juli des Jahres stattfinden.

### **postinstallativer administrativer Aufwand**

Der postinstallative Aufwand besteht aus der Wartung des Webservers, des Servlets-Runners und der generellen Sicherung des Systems. Hinzu kommt das Anlegen von Diskussionsgebieten. Die Arbeit an NetCommunity muß jedoch nicht vom Systemadministrator gemacht werden.

### **Abschließende Betrachtung**

Andere Programme, die sich auf die Funktionalität eines „Diskussionsforums“ beschränken, erfüllen hier sicherlich einfacher und billiger Ihren Zweck. Besonders das Anlegen von Benutzergruppen ist in der vorliegenden Version 8.3 umständlich gelöst, was aber durch das zu Grunde liegende Konzept bedingt ist. Dieses Problem löst sich mit der Verfügbarkeit einer LDAP – Anbindung.

## Abschließende Bewertung

	Sehr gut	Gut	Normal	Schlecht	Nicht bewertet
<b>Funktionalität</b>					
Angemessenheit				X	
Richtigkeit			X		
Sicherheit			X		

	Sehr gut	Gut	Normal	Schlecht	Nicht bewertet
<b>Zuverlässigkeit</b>					
Reife	X				
Fehlertoleranz			X		
Wiederherstellbarkeit			X		

	Sehr gut	Gut	Normal	Schlecht	Nicht bewertet
<b>Benutzbarkeit</b>					
Verständlichkeit					X
Erlernbarkeit					X
Bedienbarkeit					X

	Sehr gut	Gut	Normal	Schlecht	Nicht bewertet
<b>Effizienz</b>					
Zeitverhalten	X				
Verbrauchsverhalten			X		

	Sehr gut	Gut	Normal	Schlecht	Nicht bewertet
<b>Übertragbarkeit</b>					
Anpaßbarkeit	X				
Installierbarkeit				X	

## Erklärungen zu den einzelnen Punkten

### Angemessenheit:

Das vorliegende Programmpaket ist für den Gebrauch als Diskussionsforum extrem überdimensioniert. Der Einsatz von NetCommunity (des Moduls ‚Discussion‘) ausschließlich als Diskussionsforum steht im krassen Gegensatz zu der strategische Ausrichtung auf „1:1 Marketing“ und „Kundenbindung“<sup>15</sup>. Das Modul ‚Discussion‘ erfüllt jedoch einzeln betrachtet seinen Zweck und weist die für ein Forum üblichen Funktionen auf.

### Richtigkeit:

Im Programmpaket wurden keine Fehler gefunden, weder orthographische, noch funktionelle.

### Sicherheit:

Bedingt durch das Ziel der Entwickler ist der Umfang der Software sehr groß. Daher ist die Realisierung als Servlets aus Performancegründen geboten. Dies bedingt natürlich wiederum den Einsatz einer Servlet-Engine, deren Sicherheit im Normalfall vom jeweiligen Systemadministrator abhängt. Generell ist die Sicherheit von den benutzten Komponenten abhängig, deshalb kann hier kein allgemeingültiges Urteil ergehen.



**Reife:**

Das Programmpaket liegt mittlerweile in der Revision 8.3 vor. Das Handbuch macht deutlich, daß (im Gegensatz zu anderen Softwareherstellern) keine Versionsnummern übersprungen wurden. Die hier gemachten Aussagen beziehen sich jeweils nur auf das Modul ‚Core‘ und ‚Discussion‘. Beide Elemente sind durchdacht und dem Ziel nach erfolgreich realisiert. Grund zur Beanstandung geben hier ausschließlich die vorkonfigurierten Fehlermeldungen auf englisch, die man erst von Hand ins Deutsche übersetzen muß.

**Fehlertoleranz:**

Die evaluierte Software erwies sich im Test genügend tolerant gegenüber Bedienungsfehlern. Probleme bereitete allerdings die Installation einer anderen Software. Es kam zu nicht behebbaren Fehlern, NetCommunity verweigerte den Start, der komplette Rechner mußte neu installiert werden.

**Wiederherstellbarkeit:**

Die Wiederherstellbarkeit der Software hängt sehr stark von den eingesetzten Komponenten ab, daher kann hier nur eine Aussage für den Prototypen getroffen werden. Ein Vollbackup der Datenbank und des Webservers (Server-Root und Document-Root) mit Bordmitteln und die anschließende Wiederherstellung brachten keine Probleme. Ebenso sollte der Einsatz einer professionellen Backup – Lösung (wie z.B. Legato NetWorker) zu einer guten Wiederherstellbarkeit beitragen. Die Test-Community wurde allerdings mehrmals stark von der Installation einer anderen Software aus dem Tritt gebracht, so das eine Restaurierung ohne Backup nicht mehr möglich war.

**Verständlichkeit, Erlernbarkeit, Bedienbarkeit**

Diese 3 Punkte wurden nicht bewertet, hängen sie doch zu stark von der Gestaltung der Oberflächen ab. Sie sind aber für die Akzeptanz der Software unter den Benutzern extrem wichtig, daher wird empfohlen, Spezialisten wie Grafikdesigner oder Softwareergonomen mit der Entwicklung der Templates beauftragen.

**Zeitverhalten**

Durch die Realisierung des Programms als Servlets ist das Zeitverhalten besser als z.B. bei CGI – Skripten. Bei einer großen Zahl von Benutzern kann das Zeitverhalten über verschiedene Mechanismen optimiert werden. Hier sind die Anzahl der ständig bestehenden Datenbankverbindungen und der Durchsatz des Gesamtsystems zu nennen.

**Verbrauchsverhalten**

Das System verbrauchte die Systemressourcen nicht übermäßig, Hauptspeicherverbrauch und Festplattenbelegung hängen jedoch stark von der Anzahl der Benutzer ab.

**Anpaßbarkeit**

Das gesamte „look and feel“ der Software ist über Templates zu verändern. Diese Templates können sehr frei gestaltet werden, müssen jedoch einige wenige Elemente zwingend beinhalten.

**Installierbarkeit**

Aus den bereits mehrmals erläuterten Gründen ist die Installation von NetCommunity nicht einfach. Hat man jedoch einmal die Tücken des Handbuches erkannt und sich mit den Schwachstellen der Konfiguration vertraut gemacht, dauert die Installation nicht länger als einen Vormittag.

## Anhänge

### ***Beschreibung des Testrechners***

Prozessor: Pentium 2  
Takt: 400 MHz  
RAM: 128 MB  
Festplatte: 6 GB EIDE  
Netzwerkkarte: 3Com EtherNIC

### ***Beschreibung des Prototypen***

#### Benutzergruppen

Gruppe 1: Vorstand  
Gruppe 2: Management  
Gruppe 3: Angestellte

#### Nicknames

Vor1, Vor2, Vor3 alle in Gruppe Vorstand  
Man1, Man2, Man3 alle in Gruppe Management  
Ang1, Ang2, Ang3 alle in Gruppe Angestellte

#### Definition der Benutzergruppen

Vorstand: nick[0]='V'  
Management: nick[0]='M'  
Angestellte: nick[0]='A'

#### Webserver

Aus Sicherheitsgründen ist der Zugang zum Webserver auf die Audi AG und das Softwarelabor an der FH Heilbronn beschränkt (.audi.de .swl.fh-heilbronn.de). Hier ist auch darauf zu achten, daß

### ***Literatur***

NetCommunity - Modul Core:  
Dokumentation für Installation, Gestaltung und Betrieb, Stand: 1.9.1999

NetCommunity - Modul Discussion:  
Dokumentation für Installation, Gestaltung und Betrieb, Stand: 1.9.1999

## Endnoten

---

<sup>1</sup> <http://www.cassiopeia.de>

<sup>2</sup> <http://www.redhat.com>, <http://www.linuxiso.org>

<sup>3</sup> <http://www.apache.org>

<sup>4</sup> <http://www.ibm.com/java/jdk/118/linux/>

<sup>5</sup> <http://www.rpm.org>

<sup>6</sup> <http://www.sun.com>

<sup>7</sup> <http://www.mysql.com>

<sup>8</sup> <http://www.gjt.org/javadoc/org/gjt/mm/mysql/package-summary.html>

<sup>9</sup> <http://www.pathname.com/fhs>

<sup>10</sup> <http://java.apache.org>

<sup>11</sup> Dokumentation zum Modul Core, Kapitel 2, Einführung, S. 5

<sup>12</sup> vgl. 11

<sup>13</sup> Dokumentation zum Modul Core, Kapitel 5, Unterkapitel 5.2.3.1: Standardfelder, S. 19

<sup>14</sup> Dokumentation zum Modul Core, Kapitel 5, Unterkapitel 5.2.3.2: Beliebige Zusatzfelder, S. 23

<sup>15</sup> vgl. 10