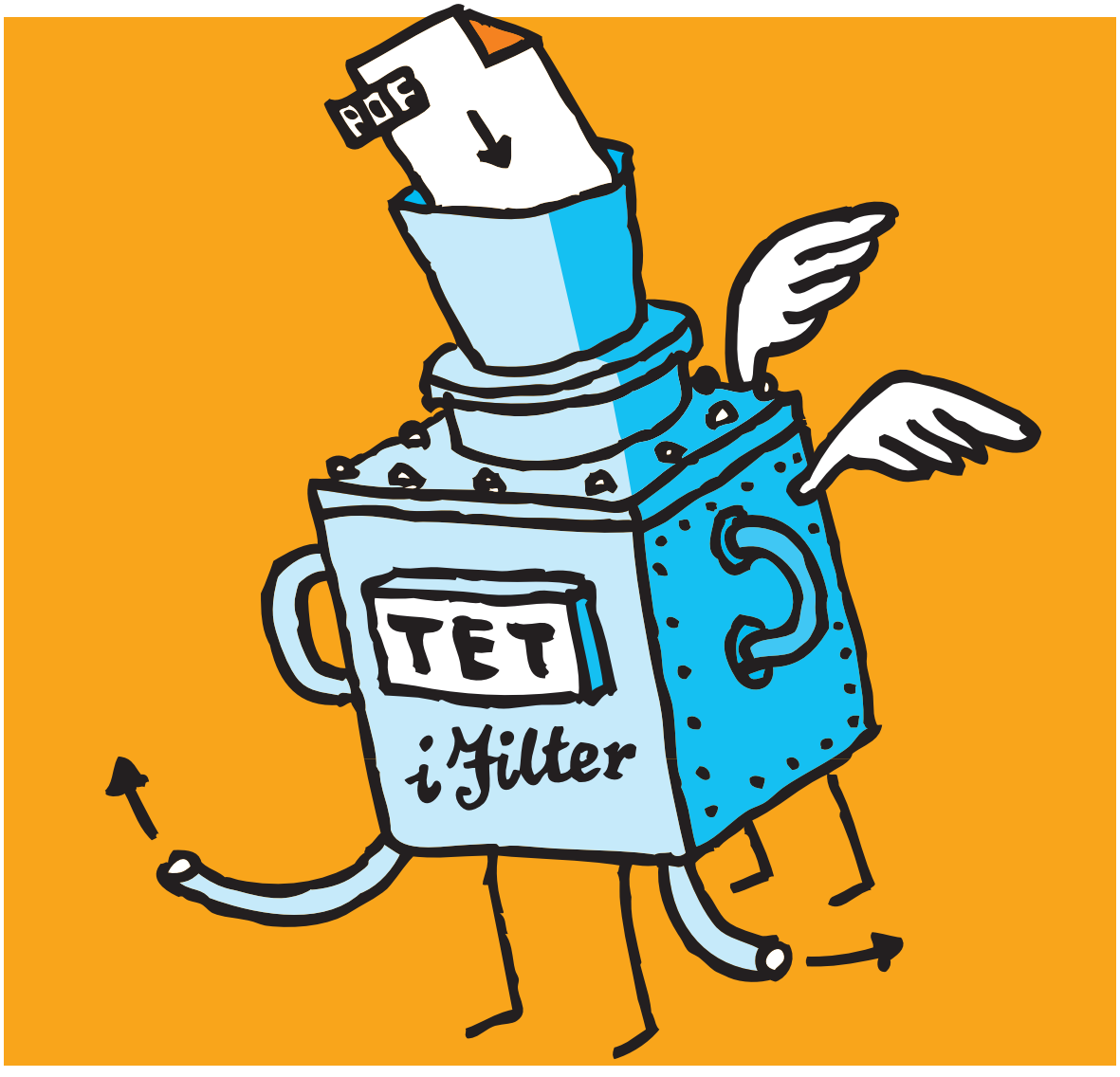
PDFlib GmbH

TET PDF IFilter

Version 5.0r1

Enterprise PDF Search für Windows



PDFlib GmbH
Franziska-Bilek-Weg 9, D-80339 München
www.pdflib.com
Tel. +49 • 89 • 452 33 84-0
Fax +49 • 89 • 452 33 84-99

Bei Fragen können Sie die PDFlib-Mailing-Liste abonnieren und sich deren Archiv ansehen unter: groups.yahoo.com/neo/groups/pdflib/info.

Vertriebsinformationen: sales@pdflib.com
Support für Inhaber einer kommerziellen PDFlib-Lizenz: support@pdflib.com (geben Sie bitte immer Ihre Lizenznummer an)

Inhaltsverzeichnis

0 Installation von TET PDF IFilter

TET wird als MSI-Paket für Windows ausgeliefert. Alle TET PDF IFilter-Pakete enthalten eine signierte IFilter-DLL sowie Hilfsdateien, Dokumentation und Beispiele. Zum Ausführen der MSI-Installationsroutine benötigen Sie Administratorrechte. Die Installationsroutine installiert und registriert TET PDF IFilter. Zusätzliche Schritte für bestimmte Suchumgebungen (z.B. Windows Search, SharePoint) sowie benutzerdefinierte Konfigurationen sind in diesem Handbuch gesondert aufgeführt.

32-Bit- und 64-Bit-Versionen. TET PDF IFilter ist für 32-Bit- und 64-Bit-Versionen verfügbar. Beide Versionen werden in separaten Paketen geliefert und können bei Bedarf auf dem selben System installiert werden. Die 64-Bit-Version ist eine native 64-Bit-Implementierung, die nur auf 64-Bit-Systemen installiert und ausgeführt werden kann. Die 32-Bit-Version lässt sich hingegen sowohl auf 32-Bit- als auch auf 64-Bit-Systemen ausführen.

Update auf eine neuere Version von TET PDF IFilter. Sie können eine ältere Version von TET PDF IFilter auf dem System zuerst deinstallieren, bevor Sie die neue Version installieren. Die Installationspakete enthalten immer die Vollversion des Produkts und setzen keine bestehende Installation voraus.

Anwendung des Lizenzschlüssels für TET PDF IFilter. Zum produktiven Einsatz von TET PDF IFilter benötigen Sie einen gültigen Lizenzschlüssel. Wenn Sie eine Lizenz für TET PDF IFilter erworben haben, müssen Sie zur Verarbeitung von großen Dokumenten den Lizenzschlüssel eingeben. Normalerweise geben Sie den Lizenzschlüssel bei der Installation von TET PDF IFilter in der MSI-Installationsroutine an. Alternativ können Sie den Lizenzschlüssel auch nach der Installation direkt in der Registry eintragen (siehe »Manuelle Installation«, Seite 6). 32-Bit- und 64-Bit-Versionen können die selben Lizenzschlüssel verwenden.

Mit dem Lizenzschlüssel 0 (Null) lässt sich die Evaluierungsversion auf einem Server oder die kostenlose Desktop-Version für den nicht kommerziellen Einsatz installieren.

Einschränkungen der Evaluierungsversion. TET PDF IFilter kann auch ohne kommerzielle Lizenz als voll funktionsfähige Evaluierungsversion verwendet werden. Ohne gültigen Lizenzschlüssel unterstützt TET PDF IFilter alle Funktionen, verarbeitet aber nur PDF-Dokumente mit bis zu 10 Seiten und einer Größe bis zu 1 MB. Evaluierungsversionen von TET PDF IFilter dürfen nicht im produktiven Einsatz, sondern nur für die Evaluierung des Produkts verwendet werden. Zum produktiven Einsatz von TET PDF IFilter benötigen Sie einen gültigen Lizenzschlüssel.

Kostenlose Desktop-Version für die nicht kommerzielle Nutzung. TET PDF IFilter für Desktop-Systeme, d.h. Windows XP/Vista/7/8/10, sind für den persönlichen Bedarf, d.h. die nicht kommerzielle Nutzung kostenlos verfügbar. Jegliche kommerzielle Verwendung auf Desktop-Systemen erfordert jedoch einen kommerziellen Lizenzschlüssel.

TET PDF IFilter für Windows Server erfordert stets eine kommerzielle Lizenz.

Unterstützte IFilter-Clients. TET PDF IFilter implementiert die IFilter-Schnittstelle von Microsoft, die von zahlreichen Produkten zur Volltextindizierung unterstützt wird. Im

vorliegenden Handbuch werden diese IFilter-Clients genannt. TET PDF IFilter wurde mit folgenden Produkten getestet, kann aber auch mit anderen Produkten von Microsoft oder Drittanbietern funktionieren, die die IFilter-Schnittstelle unterstützen:

- SharePoint 2013 und SharePoint Foundation 2013
- SharePoint Server 2010, SharePoint Foundation 2010 mit Search Server oder Search Server Express (beachten Sie, dass SharePoint Foundation 2010 als Standalone-Produkt lediglich die Indizierung der integrierten Dateitypen unterstützt, was PDF nicht mit einschließt)
- Windows SharePoint Services 3.0, SharePoint Portal Services 2003, Office SharePoint Server 2007, SharePoint Server 2010, FAST Search Server 2010 für SharePoint
- SQL Server 2005, 2008, 2012, 2014, 2016
- Search Server 2008 und Search Server Express 2008, Search Server 2010 und Search Server Express 2010
- Exchange Server 2007 und 2010
- Site Server
- Windows Search ist in den Explorer von Windows Vista/7/8/10 integriert

TET PDF IFilter ist als 32-Bit- und 64-Bit-Version verfügbar. Die 64-Bit-Version von IFilter läuft nur zusammen mit den 64-Bit Versionen der obigen Produkte.

Erforderliche Schritte nach der Installation. Für einige IFilter-Clients müssen nach der Installation von TET PDF IFilter eventuell weitere Schritte durchgeführt werden. Für weitere Informationen hierzu siehe die entsprechenden Abschnitte in Kapitel 1, »Erste Schritte«, Seite 9.

Manuelle Installation. Obwohl das Installationsprogramm aller notwendigen Schritte zum Einsatz von TET PDF IFilter durchführt, können in Einzelfällen weitere manuelle Schritte wie folgt erforderlich sein.

Um den Lizenzschlüssel manuell hinzuzufügen, tragen Sie ihn im folgenden Registry-Wert ein:

```
HKEY_LOCAL_MACHINE\SOFTWARE\PDFlib\TET PDF IFilter5\license
```

Um die TET PDF IFilter-DLL im System zu registrieren (damit sie von den IFilter-Clients gefunden werden kann) führen Sie den folgenden Befehl in der Kommandozeile aus (passen Sie bei Bedarf das Installationsverzeichnis an):

```
regsvr32 "C:\Programme\PDFlib\TET PDF IFilter 5.0 64-bit\bin\TETPDFIFilter.dll"
```

Stellen Sie sicher, diesen Befehl über eine Eingabeaufforderung mit Administratorrechten auszuführen (klicken Sie auf *Start*, geben Sie *Eingabeaufforderung* ein, rechtsklicken Sie auf *Eingabeaufforderung* und klicken Sie auf *Als Administrator ausführen)*.

Wenn TET PDF IFilter bereits von einem IFilter-Client verwendet wurde, müssen Sie alle zugehörigen Dienste, die auf TET PDF IFilter zugreifen, vor der erneuten Registrierung der DLL stoppen. Stellen Sie zudem sicher, dass die Ereignisprotokollierung von Windows geschlossen wurde.

Ausführen privilegierter Befehle. Für den Zugriff auf die Registry benötigen Sie Administratorrechte (z.B. für die Anwendungen *regsvr32* und *registerpropdesc*). Mit Administratorrechten können Sie die Eingabeaufforderung folgendermaßen aufrufen: klicken Sie auf *Start*, geben Sie im Suchfeld *cmd.exe* an, rechtsklicken Sie auf den Eintrag *Ein-*

gabeaufforderung und wählen Sie *Als Administrator* ausführen. Dies aktiviert die Eingabeaufforderung im Administratormodus.

1 Erste Schritte

Dieses Kapitel beschreibt die ersten Schritte, die zur Konfiguration und Verwendung einiger Such- und Retrieval-Produkte (IFilter-Clients) erforderlich sind, die von TET PDF IFilter unterstützt werden. Diese Beschreibung soll Sie dabei unterstützen, TET PDF IFilter schnell in Betrieb zu nehmen. Erweiterte Konfigurationsaspekte werden in Kapitel 3, »Indizierung von Metadaten«, Seite 41 erläutert. Die vollständige Installation von TET PDF IFilter enthält eine Reihe von Beispiel-PDFs, die zum Testen der Installation und für eine schnelle Übersicht über die leistungsstarke Metadaten-Funktionalität von TET PDF IFilter dienen.

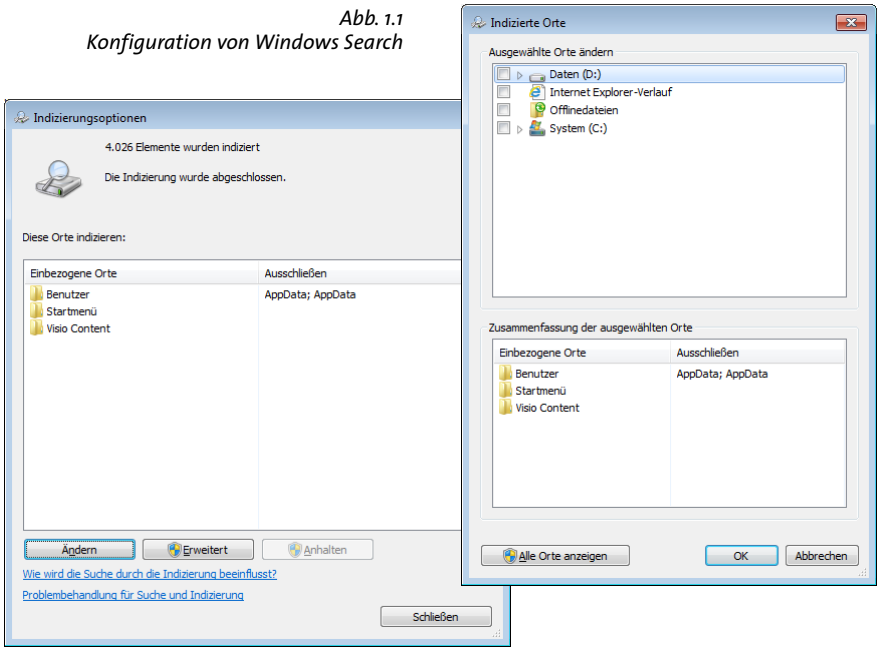
1.1 Windows Search

Systemvoraussetzungen. Windows Desktop Search (WDS) 3 und das Nachfolgeprodukt Windows Search ersetzt den veralteten Indexing Service. Es bietet Vorteile bei Leistung und Erweiterbarkeit und unterstützt ein Property-System für Metadaten, das von TET PDF IFilter unterstützt wird. Windows Search 4 ist für Windows Vista/7/8/10 und Windows Server 2008/2012 verfügbar.

Vorbereitung und Konfiguration. Windows Search indiziert Dokumente standardmäßig nur in Bibliotheken und Offlinedateien. Sie können WDS jedoch folgendermaßen anweisen, Dokumente an anderen Speicherorten (einschließlich Netzlaufwerken) zu indizieren:

- Klicken Sie auf *Start*, *Systemsteuerung*, *Indizierungsoptionen* und dann auf *Ändern*.

Abb. 1.1
Konfiguration von Windows Search



- Wählen Sie die gewünschten Orte unter *Ausgewählte Orte ändern* und klicken Sie auf *OK*.
- Klicken sie auf *Erweitert* und bei *Index löschen und neu erstellen* auf *Neu erstellen*, um die Dokumente sofort zu indizieren.

Starten/Beenden von Windows Search. Mit folgenden Methoden lässt sich der Suchdienst (d.h. die Indizierung, die wiederum TET PDF IFilter aufruft) manuell starten und beenden:

- Geben Sie zum Starten des Suchdienstes folgendes Kommando in einer Eingabeaufforderung mit Administratorberechtigung ein:

  ```
  net start wsearch
  ```

  Geben Sie zum Beenden des Suchdienstes folgendes Kommando ein:

  ```
  net stop wsearch
  ```

- Zur Steuerung des Dienstes in der Systemsteuerung:
  Klicken Sie auf *Start, Systemsteuerung, Verwaltung, Computerverwaltung* und navigieren Sie zu *Dienste und Anwendungen, Dienste*. Navigieren Sie in der Ansicht der verfügbaren Dienste zu *Windows Search* und doppelklicken Sie darauf. Im angezeigten Dialog haben Sie unter anderem die Möglichkeit, den Dienst zu *Starten* bzw. zu *Beenden*.
- Um den Katalog erneut aufzubauen:
  Klicken Sie auf *Start, Systemsteuerung, Indizierungsoptionen, Erweitert, Neu erstellen*.
  Alle Dokumente werden erneut indiziert.

Beachten Sie, dass WDS den Indizierungsdienst in bestimmten Situationen automatisch startet.

Einfache Textsuche. Zur Textsuche können Sie mehrere Methoden verwenden:

- WDS zeigt das Deskbar-Suchfeld im Startmenü an. Geben Sie die Suchbegriffe in das Suchfeld ein und drücken Sie die *Enter*-Taste.
- Öffnen Sie das Suchfenster mit *Windows-F*.
- Geben Sie im Windows Explorer den Suchbegriff in das Suchfeld rechts oben ein.

WDS zeigt jedes Mal eine Trefferliste aller Dokumente an, die den gesuchten Begriff enthalten. Ist ein PDF-Viewer installiert (z.B. Adobe Acrobat oder Adobe Reader) werden die Inhalte des ausgewählten Dokuments im Ergebnisfenster angezeigt.

Erweiterte Textsuche. Zusätzlich zur einfachen Textsuche können Sie verschiedene Suchfilter zur Eingrenzung Ihrer Suche hinzufügen. Tabelle 1.1 enthält einige Beispiele. Eine vollständige Listevon Syntaxelementen für die erweiterte Suchabfrage finden Sie unter

http://windows.microsoft.com/en-us/windows7/advanced-tips-for-searching-in-windows

Für die Abfrage von Metadaten siehe Abschnitt 4.1, »Metadaten in Windows Search«, Seite 53.

Tabelle 1.1 Beispielsyntax für die Suchabfrage mit Windows Search 3.0 und höher

Beispiel für Suchbegriff	*Beschreibung*
Hol	*Dokumentinhalte oder -eigenschaften enthalten Wörter, die mit* `Hol` *beginnen*
Holmes UND Watson *Holmes + Watson* *Holmes Watson* *(Holmes Watson)*	*Dokumentinhalte enthalten sowohl* `Holmes` *als auch* `Watson`
"Sherlock Holmes"	*Dokumentinhalte enthalten den exakten Begriff* `Sherlock Holmes`
Holmes ODER Watson	*Dokumentinhalte enthalten entweder* `Holmes` *oder* `Watson`
Holmes NICHT Mowgli *Holmes -Mowgli*	*Dokumentinhalte enthalten den Begriff* `Holmes`*, aber nicht den Begriff* `Mowgli`

Programmgesteuerte Indexabfrage. Zusätzlich zur interaktiven Suche können Sie den WDS-Index auch programmgesteuert abfragen. Der Zugriff auf den Suchindex kann über die AQS-Schnittstelle *(Advanced Query Syntax)* sowie SQL-Syntaxerweiterungen erreicht werden, die den Suchindex durch eine Datenbank-ähnliche Programmierschnittstelle präsentieren. Für Informationen zur Abfrage von Metadaten mit SQL siehe »SQL-Abfragen für Metadaten-Properties«, Seite 56.

Einschränkungen. Bei unseren Tests stießen wir auf die folgenden internen Einschränkungen von WDS:

- Folgende Einschränkung bezieht sich nur auf Windows XP/Vista: der für ein Dokument erzeugte Katalogeintrag kann eine Größe von ca. 1 MB nicht überschreiten. Wird für ein Dokument ein größerer Katalogeintrag erzeugt, wird der verbleibende Text ignoriert. Bei einer typischen Größe von ca. 3 KB pro Textseite reicht dies für mehr als 300 Seiten Text. Es gibt jedoch keine feste Grenze, deshalb können die Ergebnisse variieren. Als einzige bekannte Lösung für diese Einschränkung bei WDS muss das Dokument in kleinere Teile zerlegt werden.
- PDF-Dokumente werden nicht indiziert, so lange sie in Acrobat geöffnet sind. Wir empfehlen, Acrobat vor der Erzeugung eines Suchindexes zu schließen.
- Die Suche nach benutzerdefinierten Metadaten-Eigenschaften funktioniert nicht interaktiv, sondern nur bei programmgesteuerten Suchabfragen (z.B. mit PowerShell).

1.2 SharePoint und FAST Search

1.2.1 Systemvoraussetzungen

TET PDF IFilter läuft mit folgenden SharePoint-Konfigurationen:

- SharePoint Server 2013 und SharePoint Foundation 2013; Hotfix KB2883000 oder 8. Juli 2014 Kumulatives Update für SharePoint Server 2013 ist erforderlich.
- SharePoint Server 2010, SharePoint Foundation 2010 mit Search Server oder Search Server Express (beachten Sie, dass SharePoint Foundation 2010 als Standalone-Produkt lediglich die Indizierung der integrierten Dateitypen unterstützt, was PDF nicht mit einschließt)
- FAST Search Server 2010 für SharePoint
- Microsoft Office SharePoint Server 2007
- SharePoint Portal Server 2003
- Windows SharePoint Services 3.0

Weitere Informationen zu SharePoint finden Sie unter *msdn.microsoft.com/en-us/sharepoint/default.aspx*

1.2.2 Installation für SharePoint 2013

Gehen Sie folgendermaßen vor, um TET PDF IFilter für SharePoint 2013 zu konfigurieren:

- Installieren Sie TET PDF IFilter über die MSI-Installationsroutine.
- Öffnen Sie das Fenster *SharePoint 2013 Management Shell* und geben Sie die folgenden Kommandos ein, um TET PDF IFilter für die PDF-Indizierung vorzubereiten:

```
$ssa = Get-SPEnterpriseSearchServiceApplication
Set-SPEnterpriseSearchFileFormatState -SearchApplication $ssa -Identity pdf ↵
     -UseIFilter $true -Enable $true
```

- Sie können mit folgendem Aufruf prüfen, ob das vorige Kommando erfolgreich war:

```
Get-SPEnterpriseSearchFileFormat -SearchApplication $ssa -Identity pdf
```

  Die Ausgabe dieses Kommandos sollte in etwa so aussehen (der Eintrag *UseIFilter* sollte den Wert *True* haben):

```
Identity   : pdf
Name       : PDF
MimeType   : application/pdf
Extension  : .pdf
BuiltIn    : True
Enabled    : True
UseIFilter : True
```

- Starten Sie jetzt den Suchdienst von SharePoint 2013 erneut:

```
net stop OSearch15
net stop SPSearchHostController
net start SPSearchHostController
```

  Der Start des Dienstes *SPSearchHostController* bewirkt implizit auch den Start des Dienstes *OSearch15*.

1.2.3 Installation für SharePoint 2010 und frühere Versionen

Installation und Neustart. Gehen Sie zur Konfiguration von TET PDF IFilter mit SharePoint 2010 und früheren Versionen folgendermaßen vor; für SharePoint Foundation 2010 mit Search Server oder Search Server Express konfigurieren Sie das PDF-Symbol wie unten angegeben und TET PDF IFilter selbst wie beschrieben in Abschnitt 1.3, »Search Server«, Seite 15:

- Installieren Sie TET PDF IFilter über die MSI-Installationsroutine.
- Wenn SharePoint während der Installation von TET PDF IFilter läuft, muss der SharePoint-Suchdienst erst neu gestartet werden, bevor der neu installierte PDF-IFilter vom SharePoint Crawler erkannt wird. Wenn Sie ihn nicht erneut starten, erhalten Sie folgende Fehlermeldung vom Crawler:

  ```
  The filtering process could not load the item.
  This is possibly caused by an unrecognized item format or item corruption.
  ```

  Den SharePoint-Suchdienst für SharePoint 2010 starten Sie mit folgendem Kommando erneut:

  ```
  net stop osearch14
  net start osearch14
  ```

  Den SharePoint-Suchdienst für SharePoint 2007 und frühere Versionen starten Sie mit folgendem Kommando erneut:

  ```
  net stop osearch
  net start osearch
  ```

- Öffnen Sie die SharePoint-Verwaltungsseite: *Start, Alle Programme, Microsoft Office Server, SharePoint 3.0 Zentraladministration*.
- Dies öffnet die Webseite *Suchdienstverwaltung: SharedServices1*. Unter *Suchen* klicken Sie auf *Sucheinstellungen*. Klicken Sie auf *Dateitypen*.
- Klicken Sie auf der sich öffnenden Seite auf *Neuer Dateityp* und erstellen Sie die Dateinamenserweiterung *pdf*.
- Optional können Sie ein Symbol für PDF-Dateien erstellen, wie im nächsten Abschnitt beschrieben.

Hinzufügen eines Symbols für PDF-Dateien. Um PDF-Dokumente in der Ergebnisliste einer SharePoint-Abfrage visuell darzustellen, können Sie ein PDF-Symbol konfigurieren, das für PDF-Dokumente angezeigt wird. Dazu gehen Sie wie folgt vor:

- Laden Sie das PDF-Symbol von folgender Seite herunter:

  www.adobe.com/misc/linking.html#pdficon

  Speichern Sie es unter dem Namen *pdficon_small.gif*.

- Für SharePoint Server 2010 und SharePoint Foundation 2010 kopieren Sie die Symboldatei in das folgende Verzeichnis:

  ```
  C:\Programme\Common Files\Microsoft Shared\Web Server Extensions\14\TEMPLATE\IMAGES
  ```

  Für SharePoint Server 2007 und Windows SharePoint Services 3.0 kopieren Sie die Symboldatei in das folgende Verzeichnis:

  ```
  C:\Programme\Common Files\Microsoft Shared\Web Server Extensions\12\Template\Images
  ```

Für SharePoint Portal Server 2003 kopieren Sie die Symboldatei in das folgende Verzeichnis:

```
C:\Programme\Common Files\Microsoft Shared\Web Server Extensions\60\Template\Images
```

- Für SharePoint Server 2010 und SharePoint Foundation 2010 laden Sie die Konfigurationsdatei *Docicon.xml* in das folgende Verzeichnis in einem Texteditor:

  ```
  C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\14\TEMPLATE\XML
  ```

  Für SharePoint Server 2007 und Windows SharePoint Services 3.0 laden Sie die Konfigurationsdatei *Docicon.xml* in das folgende Verzeichnis in einem Texteditor:

  ```
  C:\Programme\Common Files\Microsoft Shared\Web server extensions\12\Template\Xml
  ```

  Für SharePoint Portal Server 2003 laden Sie die Konfigurationsdatei *Docicon.xml* in das folgende Verzeichnis in einem Texteditor:

  ```
  C:\Programme\Common Files\Microsoft Shared\Web server extensions\60\Template\Xml
  ```

- Fügen Sie die folgende Zeile dem Element *<ByExtension>* als Unterelement hinzu:

  ```
  <Mapping Key="pdf" Value="pdficon_small.gif"/>
  ```

- Starten Sie den IIS-Webserver erneut.

1.2.4 Einfache und erweiterte Textsuche

Für den Aufbau von Suchabfragen in SharePoint stehen Ihnen mehrere Methoden zur Verfügung:

- Stichwortsuche
- SQL-Abfragen
- In URLs kodierte Abfragen

Für weitere Information siehe

msdn.microsoft.com/en-us/library/ms497338.aspx

Für die Abfrage von Metadaten siehe Abschnitt 4.2, »Metadaten in SharePoint und Search Server«, Seite 59.

1.3 Search Server

Systemvoraussetzungen. TET PDF IFilter läuft mit folgenden Versionen von Search Server:

- Search Server 2008 und Search Server 2008 Express
- Search Server 2010 und Search Server 2010 Express

Vorbereitung und Konfiguration. Die Konfiguration von Search Server funktioniert ähnlich wie die Konfiguration von SharePoint. Gehen Sie folgendermaßen vor, um TET PDF IFilter für Search Server zu konfigurieren:

- Installieren Sie TET PDF IFilter über die MSI-Installationsroutine.
- Führen Sie nach der Installation die unten beschriebenen Schritte durch.
- Öffnen Sie die Search Server-Verwaltungsseite: *Start, Alle Programme, Microsoft Search Server, Search Server 2008 Zentraladministration*.
- Dies öffnet die Webseite *Suchdienst verwalten*. Unter *Crawlereinstellungen* klicken Sie auf *Dateitypen*.
- Klicken Sie auf der sich öffnenden Seite auf *Neuer Dateityp* und erstellen Sie die Dateinamenserweiterung *pdf*.
- Optional können Sie ein Symbol für PDF-Dateien erstellen, wie für SharePoint in »Hinzufügen eines Symbols für PDF-Dateien«, Seite 13 beschrieben.

Erforderliche Schritte nach der Installation. Wenn Search Server während der Installation von TET PDF IFilter läuft, muss der Suchdienst erst neu gestartet werden, bevor der neu installierte PDF-IFilter vom Search Server Crawler erkannt wird. Wenn Sie ihn nicht erneut starten, erhalten Sie eine Fehlermeldung vom Crawler.

Search Server starten Sie mit folgendem Kommando erneut:

```
net stop spsearch
net start spsearch
```

Einfache und erweiterte Textsuche. Die Konfiguration der Abfragen funktioniert bei Search Server genau sowie bei SharePoint. Für weitere Informationen siehe *msdn.microsoft.com/en-us/library/aa981100.aspx*.

1.4 Exchange Server

Systemvoraussetzungen. TET PDF IFilter läuft mit Microsoft Exchange Server 2010. Ältere Versionen von Exchange Server wurden nicht getestet, können aber durchaus kompatibel sein.

Vorbereitung und Konfiguration. Gehen Sie folgendermaßen vor, um TET PDF IFilter für Exchange Server zu konfigurieren:

- Installieren Sie TET PDF IFilter über die MSI-Installationsroutine.
- Führen Sie nach der Installation die unten beschriebenen Schritte durch.

Erforderliche Schritte nach der Installation. TET PDF IFilter muss für Microsoft Exchange Server registriert werden, indem das folgende PowerShell-Skript mit Administratorrechten ausgeführt wird:

```
register_in_exchange_2010.ps1
```

Dieses Skript wird im Unterverzeichnis *IFilter clients\Exchange* des Installationsverzeichnisses von TET PDF IFilter installiert. Nachdem das Skript erfolgreich ausgeführt wurde, müssen Sie den Dienst *Microsoft Exchange Search Indexer* erneut starten. Führen Sie diesen Schritt entweder über die Systemsteuerung unter Dienste durch oder über eine Kommandozeile in der PowerShell:

```
stop-service MSExchangeSearch -Force
start-service MSExchangeSearch
```

Wenn eine neuere Version von TET PDF IFilter installiert wird, muss das Skript zur Registrierung erneut ausgeführt werden, da sich das Installationsverzeichnis von TET PDF IFilter bei jedem Update ändert.

Nachdem TET PDF IFilter registriert wurde, werden die PDF-Anhänge aller neuen Mails von Exchange indiziert. Um PDF-Anhänge bestehender Nachrichten zu indizieren, müssen alle Mailboxen erneut indiziert werden. Der folgende Artikel auf der MSDN-Website beschreibt die möglichen Prozeduren zum Neuaufbau eines Exchange-Volltextindex:
technet.microsoft.com/de-de/library/aa995966(v=EXCHG.80).aspx

1.5 SQL Server

Systemvoraussetzungen. TET PDF IFilter läuft mit folgenden Varianten von SQL Server:

- SQL Server 2005 Workgroup, Standard und Enterprise
- SQL Server 2005 Express
- SQL Server 2008, 2012, 2014, 2016

Weitere Informationen zu SQL Server finden Sie unter
msdn.microsoft.com/de-de/library/bb418498.aspx

Weitere Informationen zur Volltextsuche in SQL Server finden Sie unter
msdn.microsoft.com/de-de/library/mt590198(v=sql.1).aspx

Vorbereitung und Konfiguration. Um Ihnen die volle Kontrolle über den Einsatz von Filtern in SQL Server zu geben, registriert das Installationsprogramm TET PDF IFilter in keiner von SQL Server automatisch. Daher müssen Sie TET PDF IFilter manuell für alle Instanzen von SQL Server separat registrieren.

Mit den folgenden Schritten wird SQL Server angewiesen, auf systemweit installierte IFilter zuzugreifen:

- Installieren Sie TET PDF IFilter über die MSI-Installationsroutine.
- Öffnen Sie SQL Server Management Studio und führen Sie die folgenden Anweisungen aus, um dieser Instanz von SQL Server die systemweiten Dokumentfilter verfügbar zu machen (für weitere Informationen siehe *msdn.microsoft.com/en-us/library/dd207002%28v=sql.120%29.aspx*:

```
exec sp_fulltext_service 'load_os_resources', 1;
GO
exec sp_fulltext_service 'update_languages'
GO
exec sp_fulltext_service 'restart_all_fdhosts'
GO
```

Testen der Konfiguration. Sie können die Konfigurationsergebnisse prüfen, um sicherzustellen, dass TET PDF IFilter für eine Instanz von SQL Server verfügbar ist. Verwenden Sie dazu folgende Anweisungen:

```
SELECT document_type, path FROM sys.fulltext_document_types WHERE document_type = '.pdf'
```

Eine Ausgabe ähnlich der folgenden bedeutet, dass TET PDF IFilter für die Instanz erfolgreich konfiguriert wurde (die genaue Pfadangabe hängt von Ihrem Installationspfad ab):

```
.pdf    C:\Programme\PDFlib\TET PDF IFilter 5.0 64-bit\bin\TETPDFIFilter.dll
```

Vorbereiten einer Datenbanktabelle für die Volltextindizierung von PDF. Mit Hilfe von TET PDF IFilter erzeugt SQL Server den Volltextindex für PDF-Dokumente in einer Spalte vom Typ *varbinary(max)*. Da der Dokumenttyp in dieser Situation nicht verfügbar ist, muss die Dateinamenserweiterung in einer separaten Tabellenspalte, der sogenannten Typ-Spalte *(type column)* gespeichert werden. Die Typ-Spalte kann ein beliebiger zeichenbasierten Datentyp sein. Wir verwenden *VARCHAR(4)* und speichern die Dateinamenserweiterung *pdf*.

Die folgenden Anweisungen erzeugen eine *DocumentTable* mit einem Beispiel-PDF in der Spalte *data*, dem Dateinamen in der Spalte *name* und dem zugehörigen Typ in der Spalte *extension*:

```
CREATE DATABASE TestDatabase
GO
USE TestDatabase
GO
CREATE TABLE DocumentTable
(pk INT NOT NULL IDENTITY CONSTRAINT DocumentTablePK PRIMARY KEY,
data VARBINARY(MAX), name VARCHAR(100), extension VARCHAR(4))
GO
INSERT INTO DocumentTable(data, name, extension) SELECT *,'The_Hound_of_the_
Baskervilles.pdf','pdf' FROM OPENROWSET(BULK 'C:\Program Files\PDFlib\TET PDF IFilter 5.0
64-bit\PDF samples\The_Hound_of_the_Baskervilles.pdf', SINGLE_BLOB) AS Document
GO
```

Nun können Sie den Volltextindex erzeugen:

```
sp_fulltext_database 'enable'
GO
CREATE FULLTEXT CATALOG TestCatalog AS DEFAULT
GO
CREATE FULLTEXT INDEX ON DocumentTable (data TYPE COLUMN extension)
KEY INDEX DocumentTablePK
GO
```

Löschen und Neuaufbau des Volltextindex. Mit folgenden Anweisungen können Sie den Volltextindex löschen:

```
USE TestDatabase
GO
DROP FULLTEXT INDEX ON DocumentTable
GO
```

Mit folgenden Anweisungen können Sie den Volltextindex neu aufbauen:

```
USE TestDatabase
CREATE FULLTEXT INDEX ON DocumentTable (data TYPE COLUMN extension)
KEY INDEX DocumentTablePK
GO
```

Einfache und erweiterte Textsuche. Mit folgender Anweisung können Sie einzelne Wörter im Volltextindex abfragen:

```
SELECT name FROM DocumentTable WHERE CONTAINS(*, 'Watson')
GO
```

Wenn Sie nach einem Satz suchen, der aus mehreren Wörtern besteht, schließen Sie ihn in doppelte Anführungszeichen ein:

```
SELECT name FROM DocumentTable WHERE CONTAINS(*,'"Arthur Conan Doyle"')
GO
```

Ein Beispielskript für die Ausführung dieser Schritte mit den mitgelieferten PDF-Beispielen wird mit TET PDF IFilter installiert. Weitere Informationen zum Prädikat *CONTAINS* in Transact-SQL finden Sie unter

msdn.microsoft.com/de-de/library/ms187787(SQL.100).aspx

Für die Abfrage von Metadaten siehe Abschnitt 4.3, »Metadaten in SQL Server«, Seite 64.

2 Indizierung von PDF-Inhalten

2.1 PDF-Dokumentdomänen

PDF-Dokumente können außer den Seiteninhalten noch an vielen anderen Stellen Text enthalten, z.B. in Anmerkungen oder Lesezeichen. Sie können auch Metadaten im XMP-Format oder als klassische Dokument-Infofelder enthalten. Die Stellen, an denen in einem PDF-Dokument Text vorkommt, werden als PDF-Dokumentdomänen bezeichnet. In der Liste unten werden alle PDF-Dokumentdomänen aufgeführt und erklärt, wie man den zugehörigen Text in Acrobat anzeigt. Die Liste enthält auch die Standardaktionen von TET PDF IFilter für alle Dokumentdomänen. TET PDF IFilter indiziert Text aus allen relevanten Quellen. Deshalb erhält man auch Suchtreffer für Stellen, die auf den ersten Blick keinen Text zu enthalten scheinen. Da Suchtreffer in IFilter-Clients normalerweise nicht markiert werden, muss man wissen, wie man Suchbegriffe in durchsuchten Dokumenten auffinden kann. Denken Sie daran, dass der gesuchte Text an einer anderen Stelle als dem eigentlichen Seiteninhalt auftreten kann. Wenn Sie Probleme haben, den Suchb egriff in einem PDF-Dokument zu lokalisieren, für das TET PDF IFilter einen Suchtreffer meldet, ziehen Sie die Liste unten zu Rate.

Beachten Sie folgendes:

- Das Durchsuchen von »Mehreren PDFs« bei Acrobat bezieht sich auf folgende Art von Suche: *Bearbeiten, Erweiterte Suche*. Klicken Sie auf *Mehr Optionen anzeigen* (sofern vorhanden). Wählen Sie unter *Suchen in:* das gewünschte Verzeichnis mit PDF-Dokumenten.
- Manche Beschreibungen beziehen sich auf die Sammlungen von Property-Sets *documentXMP, imageXMP, shell, pdf* und *internal*. Diese lassen sich über die XML-Konfigurationsdatei aktivieren (siehe Abschnitt 3.3, »Vordefinierte Metadaten-Properties«, Seite 45). Die Sammlungen der Property-Sets *shell* und *internal* sind standardmäßig aktiviert, *pdf*, *documentXMP* und *imageXMP* deaktiviert. Für weitere Informationen zu Sammlungen von Property-Sets siehe Abschnitt 3.3, »Vordefinierte Metadaten-Properties«, Seite 45.
- Die Notation *@indexNestedPdf* bezieht sich auf ein Attribut in der XML-Konfigurationsdatei (siehe Abschnitt 6.2, »XML-Elemente und -Attribute«, Seite 75).

Im folgenden finden Sie Informationen zum Durchsuchen von PDF-Domänen und wie man diese mit Acrobat X/XI/DC durchsucht. Dies ist wichtig, um Suchtreffer in Acrobat lokalisieren zu können.

Text auf der Seite. Seiteninhalte sind die Hauptquelle für Text in PDF. Der Text auf einer Seite wird mit Hilfe von Fonts dargestellt und mit einer der vielen Encoding-Techniken von PDF kodiert.

- Anzeige in Acrobat: Seiteninhalte sind immer sichtbar
- Durchsuchen einer einzelnen PDF-Datei mit Acrobat X/XI/DC: *Bearbeiten, Suchen* oder *Bearbeiten, Erweiterte Suche*. TET PDF IFilter ist in der Lage, auch Text in Dokumenten zu verarbeiten, für den Acrobat keine korrekte Glyphenzuordnung zu Unicode-Werten liefern kann. In diesem Fall können Sie das auf TET basierende TET Plugin verwenden. Es verfügt über einen eigenen Suchdialog *Zusatzmodule, PDFlib*

TET Plugin..., TET Suchen, ist jedoch nicht als vollwertige Suchfunktionalität konzipiert.
- Durchsuchen mehrerer PDF-Dateien mit Acrobat X/XI/DC: *Bearbeiten, Erweiterte Suche*. Klicken Sie auf *Mehr Optionen anzeigen*. Wählen Sie unter *Suchen in:* das gewünschte Verzeichnis mit PDF-Dokumenten.
- TET PDF IFilter: Seiteninhalte werden standardmäßig indiziert. In bestimmten Situationen kann es sinnvoll sein, die Indizierung von Seiteninhalten mit *@indexPageContents=false* zu unterbinden.

Vordefinierte Dokument-Infofelder. Prinzipiell handelt es sich bei Dokument-Infofeldern um Schlüssel-/Wertpaare.
- Anzeige in Acrobat X/XI/DC: *Datei, Eigenschaften...*
- Durchsuchen einer einzelnen PDF-Datei mit Acrobat X/XI/DC: nicht verfügbar
- Durchsuchen mehrerer PDF-Dateien mit Acrobat X/XI/DC: Klicken Sie auf *Bearbeiten, Erweiterte Suche* und dann im unteren Bereich des Fensters auf *Mehr Optionen anzeigen*. Wählen Sie unter *Suchen in:* das gewünschte Verzeichnis mit PDF-Dokumenten und unter *Folgende Zusatzkriterien verwenden:* aktivieren Sie eins der folgenden: *Erstellungsdatum, Änderungsdatum, Verfasser, Titel, Thema, Stichwörter*.
- TET PDF IFilter: vordefinierte Dokument-Infofelder werden indiziert, wenn die Property-Set-Sammlung *shell* aktiviert ist.

Benutzerdefinierte Dokument-Infofelder. Sie können zusätzliche Dokument-Infofelder nach Ihren eigenen Wünschen definieren.
- Anzeige in Acrobat X/XI/DC: *Datei, Eigenschaften..., Benutzerdefiniert* (im kostenlosen Adobe Reader nicht verfügbar)
- Durchsuchen mit Acrobat X/XI/DC: nicht verfügbar
- TET PDF IFilter: benutzerdefinierte Dokument-Infofelder werden indiziert, wenn auf diesem Dokument-Infofeld basierende benutzerdefinierte Properties definiert sind.

XMP-Metadaten auf Dokumentebene. XMP-Metadaten bestehen aus einem XML-Stream mit erweiterten Metadaten.
- Anzeige in Acrobat X/XI/DC: *Datei, Eigenschaften...* und in der Registerkarte *Beschreibung* klicken Sie auf *Zusätzliche Metadaten...* (im kostenlosen Adobe Reader nicht verfügbar)
- Durchsuchen einer einzelnen PDF-Datei mit Acrobat X/XI/DC: nicht verfügbar
- Durchsuchen mehrerer PDF-Dateien mit Acrobat X/XI/DC: Klicken Sie auf *Bearbeiten, Erweiterte Suche* und dann im unteren Bereich des Fensters auf *Mehr Optionen anzeigen*. Wählen Sie unter *Suchen in:* das gewünschte Verzeichnis mit PDF-Dokumenten und unter *Folgende Zusatzkriterien verwenden:* aktivieren Sie *XMP-Metadaten* (im kostenlosen Adobe Reader nicht verfügbar).
- TET PDF IFilter: XMP-Dokumentmetadaten werden indiziert, wenn die Property-Set-Sammlung *documentXMP* aktiviert ist oder wenn benutzerdefinierte Properties auf der Basis von XMP-Dokumentdaten definiert sind.

XMP-Metadaten auf Bildebene. XMP-Metadaten können an Dokumentbestandteile wie Bilder, Seiten, Fonts usw. angehängt werden. XMP ist jedoch außer auf Dokumentebene meist nur noch auf Bildebene vorhanden.

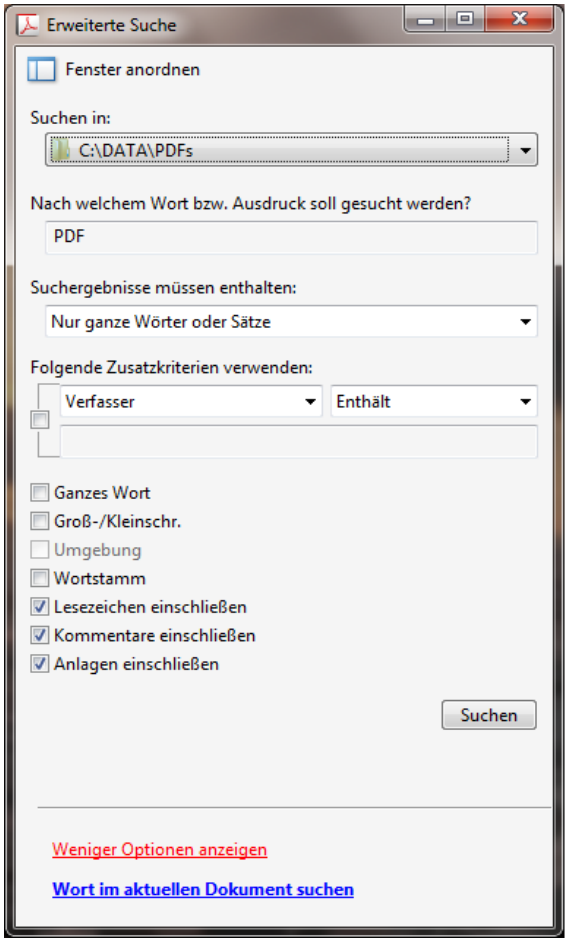


Abb. 2.1
Die erweiterte Suche von Acrobat

- Anzeige in Acrobat X: *Werkzeuge, Inhalt, Objekt bearbeiten*, rechtsklicken Sie auf das Bild und wählen Sie *Metadaten anzeigen...* (im kostenlosen Adobe Reader nicht verfügbar)
- Anzeige in Acrobat XI/DC: *Anzeige, Ein-/Ausblenden, Navigationsfenster, Inhalt*, navigieren Sie zum gewünschten Bild, rechtsklicken Sie darauf und wählen Sie *Metadaten anzeigen...* (im kostenlosen Adobe Reader nicht verfügbar).
- Durchsuchen einer einzelnen PDF-Datei mit Acrobat X/XI/DC: nicht verfügbar
- TET PDF IFilter: XMP-Metadaten auf Bildebene werden indiziert, wenn die Property-Set-Sammlung *imageXMP* aktiviert ist.

Text in Formularfeldern. Formularfelder werden über der Seite liegend angezeigt. Technisch gesehen sind sie nicht Teil des Seiteninhalts, sondern werden durch separate Datenstrukturen dargestellt.

- Anzeige in Acrobat X/XI: *Werkzeuge, Formulare, Bearbeiten* (im kostenlosen Adobe Reader nicht verfügbar). Anzeige in Acrobat DC: *Werkzeuge, Formulare vorbereiten* (im kostenlosen Adobe Reader nicht verfügbar)
- Durchsuchen mit Acrobat X/XI: nicht verfügbar

- TET PDF IFilter: Formularfelder werden indiziert, wenn benutzerdefinierte Properties auf Basis des pCOS-Pseudo-Objekts *fields* definiert sind. Wenn die Hauptseiteninhalte (d.h. die Formularfeldüberschriften) nicht erforderlich sind, weil sie konstant sind, können Sie die Indizierung der Seiteninhalte mit der Seitenoption *skipengines={text image}* deaktivieren.
  Der folgende Ausschnitt zeigt die relevanten Bestandteile der XML-Konfigurationsdatei für diese Situation (siehe Kapitel 6, »XML-Konfigurationsdatei«, Seite 73):

```
<n:Tet>
    <n:TetOptions></n:TetOptions>
    <n:DocOptions></n:DocOptions>
    <n:PageOptions>skipengines={text image}</n:PageOptions>
</n:Tet>

<n:Filtering metadataHandling="propertyAndText"/>

<n:Metadata>
    <n:PropertySet guid="E9CDA960-D09A-43bc-AAAA-BBBBBBBBBBBB">
        <n:Property friendlyName="Formfield" identifier="2">
            <n:Source pdfObject="fields[*]/V"/>
        </n:Property>
    </n:PropertySet>
</n:Metadata>
```

Text in Kommentaren (Anmerkungen). Ähnlich wie Formularfelder werden Anmerkungen (Notizen, Kommentare usw.) über der Seite liegend und durch separate Datenstrukturen dargestellt. Die relevanten Textinhalte einer Anmerkung sind abhängig von ihrem Typ. Bei Weblinks kann der relevante Teil die URL sein, während für andere Anmerkungstypen die sichtbaren Textinhalte von Bedeutung sein können.

- Anzeige in Acrobat X/XI: *Kommentar, Kommentarliste*. Anzeige in Acrobat DC: *Werkzeuge, Kommentar, Kommentarliste*
- Durchsuchen einer einzelnen PDF-Datei mit Acrobat X/XI/DC: *Bearbeiten, Suche*, aktivieren Sie *Kommentare einfügen* oder verwenden Sie die Schaltfläche *Kommentare suchen* in der Kommentarleiste.
- Durchsuchen mehrerer PDF-Dateien mit Acrobat X/XI/DC: Klicken Sie auf *Bearbeiten, Erweiterte Suche* und dann im unteren Bereich des Fensters auf *Mehr Optionen anzeigen*. Wählen Sie unter *Suchen in:* das gewünschte Verzeichnis mit PDF-Dokumenten und unter *Folgende Zusatzkriterien verwenden:* aktivieren Sie *Kommentare einschließen*.
- TET PDF IFilter: Kommentare werden indiziert, wenn die Property-Set-Sammlung *pdf* aktiviert ist

Text in Lesezeichen. Lesezeichen sind nicht direkt seitenbezogen, auch wenn sie eine Aktion enthalten können, die auf eine bestimmte Seite springt. Lesezeichen können hierarchisch verschachtelt werden.

- Anzeige in Acrobat X/XI/DC: *Anzeige, Ein-/Ausblenden, Navigationsfenster, Lesezeichen*
- Durchsuchen einer einzelnen PDF-Datei mit Acrobat X/XI/DC: *Bearbeiten, Erweiterte Suche,* aktivieren Sie *Lesezeichen einfügen*
- Durchsuchen mehrerer PDF-Datei mit Acrobat X/XI/DC: *Bearbeiten, Erweiterte Suche,* aktivieren Sie *Lesezeichen einfügen*
- Durchsuchen mehrerer PDF-Dateien mit Acrobat X/XI: Klicken Sie auf *Bearbeiten, Erweiterte Suche* und dann im unteren Bereich des Fensters auf *Mehr Optionen an-*

zeigen. Wählen Sie unter *Suchen in:* das gewünschte Verzeichnis mit PDF-Dokumenten und unter *Folgende Zusatzkriterien verwenden:* aktivieren Sie *Lesezeichen einschließen* (im kostenlosen Adobe Reader nicht verfügbar)
- TET PDF IFilter: Kommentare werden indiziert, wenn die Property-Set-Sammlung *pdf* aktiviert ist

Dateianhänge. PDF-Dokumente können Dateianhänge auf Seiten- oder Dokumentebene enthalten, die selbst wiederum PDF-Dokumente sind.
- Anzeige in Acrobat X/XI/DC: *Anzeige, Ein-/Ausblenden, Navigationsfenster, Anlagen*
- Durchsuchen mit Acrobat X/XI/DC: Klicken Sie auf *Bearbeiten, Erweiterte Suche* und aktivieren Sie *Anlagen einschließen* (im kostenlosen Adobe Reader nicht verfügbar). Verschachtelte Dateianhänge werden nicht rekursiv durchsucht.
- TET PDF IFilter: PDF-Anhänge werden rekursiv indiziert, wenn *@indexNestedPdf=true*

PDF-Pakete und -Portfolios. Bei PDF-Paketen und PDF-Portfolios handelt es sich um Dateianhänge mit zusätzlichen Eigenschaften.
- Anzeige in Acrobat X/XI/DC: Acrobat zeigt das Deckblatt der Pakete/Portfolios und ihre einzelnen Bestandteile mit einer speziellen Benutzeroberfläche für PDF-Pakete an.
- Durchsuchen eines einzelnen PDF-Pakets mit Acrobat X/XI/DC: *Bearbeiten, Gesamtes Portfolio durchsuchen*
- Durchsuchen mehrerer PDF-Pakete mit Acrobat X/XI/DC: nicht verfügbar
- TET PDF IFilter: PDF-Dokumente in Paketen/Portfolios werden rekursiv indiziert, wenn *@indexNestedPdf=true*.

PDF-Standards und andere PDF-Merkmale. Diese Domäne enthält keinen expliziten Text, sondern wird als Container für verschiedene interne Eigenschaften eines PDF-Dokuments verwendet, z.B. Status von PDF/X und PDF/A oder Tagged PDF.
- Anzeige in Acrobat X/XI/DC: *Anzeige, Ein-/Ausblenden, Navigationsfenster, Standards* (nur für standardkonforme PDF-Dokumente verfügbar)
- Durchsuchen mit Acrobat X/XI/DC: nicht verfügbar
- TET PDF IFilter: Kommentare werden indiziert, wenn die Property-Set-Sammlung *pdf* aktiviert ist.

Tagged PDF. TET rekonstruiert die Layout-Struktur und -Hierarchie direkt aus den Seiteninhalten, ohne den Strukturbaum zu verwenden, der in Tagged PDF vorhanden ist. Seiteninhalte, die zum Verständnis des Dokuments nicht erforderlich sind, sondern lediglich zu Layoutzwecken oder als Dekoration erzeugt werden, können in Tagged PDF als Artefakte ausgezeichnet werden. Die häufigste Verwendung von Artefakten findet sich bei laufenden Kopf- und Fußzeilen sowie Seitenzahlen und Kapitelüberschriften. Die häufigste Verwendung von Artefakten findet sich bei laufenden Kopf- und Fußzeilen sowie Seitenzahlen und Kapitelüberschriften. Je nach Anwendungsfall kann es wünschenswert sein, als Seiteninhalte ausgezeichnete Artefakte zu verarbeiten oder auch nicht.
- Anzeige in Acrobat XI/DC: *Anzeige, Ein-/Ausblenden, Navigationsfenster, Tags*. Klicken Sie im Menü *Tags* auf *Suchen...* und wählen Sie *Artefakte*. Als Artefakte ausgezeichneter Text, Bilder und Vektorgrafiken werden markiert.
  Alternativ können Sie folgendes aktivieren: *Werkzeuge, Barrierefreiheit, TouchUp-*

Leserichtung. Das Tool markiert die mit Tags versehenen Inhalte mit schattierten Rechtecken. Bei den nicht markierten Inhalten handelt es sich um Artefakte.

- Ignorieren von Artefakten bei der Suche in Acrobat X/XI/DC: nicht verfügbar
- Ignorieren von Artefakten bei TET: übergeben Sie die Seitenoption *ignoreartifacts*.

Ebenen. Mit Hilfe von Ebenen (der technische Begriff ist *optional content)* können Seiteninhalte sichtbar oder unsichtbar gemacht werden. Je nach Anwendungsfall kann es wünschenswert sein, Seiteninhalte auf unsichtbaren Ebenen zu verarbeiten oder auch nicht.

- Anzeige in Acrobat XI/DC: Anzeige, *Ein-/Ausblenden, Navigationsfenster, Ebenen*. Für sichtbare Ebenen wird vor dem Namen ein Augensymbol eingeblendet. Mit einem Klick auf dieses Symbol lässt sich die Sichtbarkeit steuern.
- Suche in Acrobat X/XI/DC: Acrobat sucht den Inhalt aller Ebenen. Wird ein Suchergebnis auf einer unsichtbaren Ebene gefunden, bietet Acrobat an, die Ebene sichtbar zu machen.
- Verarbeitung von Ebenen mit TET PDF IFilter: mit der Seitenoption *layers* lässt sich die Inhaltsextraktion auf sichtbare oder unsichtbare Ebenen einschränken. Alternativ können alle Ebenen verarbeitet werden, was nur sinnvoll ist, wenn diese sich nicht überlappen.

2.2 Automatische Spracherkennung

Korrekte Spracherkennung. Die Inhaltsindizierung kann durch sprachspezifische Nachbearbeitung der indizierten Inhalte erheblich verbessert werden. Folgende Aspekte sind stark sprachspezifisch, wobei Details zwischen den verschiedenen IFilter-Clients variieren können:

- Die sogenannte Wörtertrennung zerlegt den Text in einzelne Wörter.
- Mit Hilfe der Wortstammerkennung wird die Stammform indizierter Wörter extrahiert. Dadurch werden Suchtreffer erzielt, selbst wenn der Suchbegriff in leicht abgewandelter (flektierter) Form im indizierten Dokument vorkommt.
- Thesaurus-basierte Suche
- Listen mit Stoppwörtern enthalten für Suchabfragen irrelevante Wörter, die nicht in den Index aufgenommen werden, wie *der/die/das, einer/eine, und, oder*.

Automatische Spracherkennung. Die oben genannte Funktionalität kann nur implementiert werden, wenn die natürliche Sprache des Textes bekannt ist. Standardmäßig verwenden IFilter-Clients für sprachspezifische Verarbeitung die Spracheinstellung des Computers. Das Indizieren von japanischen Dokumenten auf einem englischen System wird daher nur zu einem minderwertigen Index führen und bestehende Zieldokumente können daher eventuell nicht korrekt durchsucht werden. Das Zuweisen der richtigen Sprache ist daher für Dokumente in ostasiatischen Sprachen besonders wichtig.

Um die sprachspezifische Verarbeitung während der Indizierung zu verbessern, erkennt TET PDF IFilter die natürliche Sprache von Dokumentinhalt und -Properties vom Typ *string* automatisch. Dabei kommen zwei Methoden zum Einsatz:

- In einigen Fällen ist zur Erkennung der Sprache die Prüfung des Schriftsystems ausreichend. Japanische Hiragana-Zeichen werden zum Beispiel ausschließlich in Texten in japanischer Sprache verwendet.
- Viele Sprachen verwenden jedoch das gleiche Schriftsystem, so zum Beispiel viele westeuropäische Sprachen die lateinische Schrift. In diesen Fällen bestimmt TET PDF IFilter anhand statistischer Analysen die korrekte Sprache.

Da die automatische Spracherkennung jeden Textabschnitt getrennt analysiert, kann ein Dokument beliebige Sprachmischungen enthalten. Jedem Segment wird die zugehörige Sprache folgendermaßen zugewiesen:

- Die automatische Schrifterkennung zerlegt den Text in einzelne Abschnitte mit dem selben Schriftsystem.
- Falls die Sprache anhand des Schriftsystems nicht eindeutig erkannt werden kann, wird der Text mit Hilfe der automatischen Spracherkennung analysiert. Dem gesamten Abschnitt wird auf diese Weise die wahrscheinlichste Sprache zugeordnet, er wird aber nicht weiter in kleinere Einheiten pro Sprache zerlegt. Zum Beispiel wird einem Abschnitt mit lateinischem Text in Deutsch, Englisch und Französisch abhängig von der tatsächlichen Verteilung der Sprachen eine dieser Sprachen für den gesamten Abschnitt zugeordnet.

Manuelle Sprachzuweisung. In TET PDF IFilter kann die automatische Spracherkennung für bestimmte Anwendungen konfiguriert werden. Für diese Funktion werden LCIDs (Locale Identifiers) verwendet. LCIDs geben die natürliche Sprache an und unterscheiden auch zwischen verschiedenen nationalen und regionalen Varianten (z.B. UK Englisch und US Englisch). Tabelle 2.1 führt einige weit verbreitete LCID-Werte auf.

Eine vollständige Liste aller LCIDs finden Sie in unter

msdn.microsoft.com/en-us/library/ms776294(VS.85).aspx

Hinweis *TET PDF IFilter führt über die sprachspezifische Verarbeitung hinaus keine weitere Spracherkennung durch. Die LCID-Information muss von den IFilter-Clients ausgewertet werden. Während manche IFilter-Clients (z.B. SharePoint, SQL Server) ausgefeilte LCID-Verarbeitung anbieten, wird die LCID von anderen IFilter-Clients hingegen vollständig ignoriert.*

Tabelle 2.1 Verbreitete LCID-Werte mit entsprechender Primär- und Sekundärsprache

LCID	Primärsprache	Sekundärsprache (Land)
0x0000	Neutrale Spracheinstellung (locale)	Neutrale Subsprache
0x0401	Arabisch (ar)	Saudiarabien (SA)
0x0404	Chinesisch (zh)	Traditionell (Hant)
0x0407	Deutsch (de)	Deutschland (DE)
0x0409	Englisch (en)	Vereinigte Staaten (US)
0x040c	Französisch (fr)	Frankreich (FR)
0x0410	Italienisch (it)	Italien (IT)
0x0411	Japanisch (ja)	Japan (JP)
0x0413	Holländisch (nl)	Niederlande (NL)
0x0419	Russisch (ru)	Russland (RU)
0x0804	Chinesisch (zh)	Vereinfacht (Hans)
0x0c0a	Spanisch (es)	Spanien (ES)
0x0800	Standard-Spracheinstellung des Computers	
0x1000	Nicht angegebene benutzerdefinierte Spracheinstellung	Nicht angegebene benutzerdefinierte Subsprache

XML-Konfiguration für LCIDs. LCIDs zum Überschreiben oder Ergänzen der automatischen LCID-Erkennung können im Element *LocaleId* der XML-Konfigurationsdatei angegeben werden:

```
<LocaleId default="1031" detection="auto" latin="1031" cyrillic="1026" chinese="4100"/>
```

Das Attribut *detection* kann die Werte *auto*, *disabled* und *script* haben. Wenn *detection=disabled*, werden alle anderen Attribute außer *default* ignoriert. Standardwert: *auto*. Mit der Einstellung *script* wird die Skript-Analyse gestartet, aber die statistische Analyse deaktiviert.

Mit dem Attribut *default* lässt sich eine globale LCID-Einstellung für alle Texte setzen, für die *detection=disabled*. Wenn dieses Attribut fehlt, wird die Spracheinstellung des Computers verwendet.

Für alle Attribute außer *detection* kann ein numerischer Wert in dezimaler oder hexadezimaler Syntax angegeben werden. Hexadezimale Werte müssen mit *0x* beginnen. Tabelle 2.2 führt die unterstützten Skript-Attribute mit ihren Standardwerten auf. LCIDs für Text in allen anderen Skripten wird automatisch zugewiesen.

Tabelle 2.2 Attribute zur Angabe von skriptspezifischen LCIDs

Attribut	***Standardwert***
default	*0x0800 Aktuelle Spracheinstellung des Computers (wird für alle Textabschnitte verwendet, sofern* `detection=disabled`)
latin	*0x0409 Englisch (US)*
cyrillic	*0x0419 Russisch (RU)*
arabic	*0x0401 Arabisch (SA)*
chinese	*0x0804 Chinesisch (Volksrepublik China)*

2.3 PDF-Versionen und geschützte Dokumente

PDF-Versionen. TET PDF IFilter akzeptiert alle PDF-Versionen bis PDF 1.7 Extension Level 8, dem Dateiformat von Acrobat DC. Dies umfasst verschiedene PDF-basierte ISO-Standards, z.B. PDF/A, PDF/E ud PDF/X. Beachten Sie, dass ISO 32 000-1 technisch äquivalent ist zu PDF 1.7 und daher auch unterstützt wird, genauso wie die in ISO 32 000-2 spezifizierte Verschlüsselungsmethode.

Geschützte PDF-Dokumente. TET PDF IFilter indiziert Text und Metadaten aus allen Dokumenten, sofern sie geöffnet werden können. Dies umfasst die folgenden Arten von PDF-Dokumenten:

- Unverschlüsselte Dokumente;
- Mit einem Master-Kennwort verschlüsselte Dokumente, die kein Benutzerkennwort benötigen. Der Status der Acrobat-Sicherheitseinstellungen *Kopieren von Inhalt Zulässig/Nicht Zulässig* wirkt sich auf Dokumente in dieser Gruppe nicht aus.

Auf den ersten Blick mag die zweite Kategorie wie ein Verstoß gegen die Absicht des Autors erscheinen, das Dokument zu schützen. Dies ist jedoch nicht der Fall, da TET PDF IFilter kein Mittel zum eigentlichen Kopieren des Textes bietet; es unterstützt lediglich die Suchmaschine bei der Indizierung des Dokuments und beim anschließenden Auffinden des Dokuments während der Suche. Sobald das Dokument bei der Suche identifiziert und in Acrobat geöffnet wird, gelten beim Kopieren von Inhalten noch immer die gleichen Einschränkungen, die für das Dokument festgelegt wurden.

Verschlüsselte PDF-Dokumente, die nicht geöffnet werden konnten, werden protokolliert. Diese Kategorie umfasst die folgenden Fälle:

- Mit Benutzerkennwort geschützte Dokumente, die in Acrobat nur unter Angabe des entsprechenden Kennworts geöffnet werden können.
- Verschlüsselte PDF-Anhänge in ansonsten unverschlüsselten Dokumenten.
- Dokumente, die mit einem benutzerspezifischen Sicherheitszertifikat verschlüsselt wurden.

Beschädigte PDF-Dokumente. PDF-Dokumente können beschädigte Datenstrukturen enthalten, entweder wegen fehlerhafter PDF-Erzeugung oder aufgrund einer versehentlichen Änderung (z.B. durch fehlerhafte Netzwerkübertragung). TET PDF IFilter erkennt beschädigte PDF-Dokumente und versucht sie zu reparieren, um Text und Metadaten erfolgreich extrahieren zu können. Dieser Reparaturmodus wird beim Indizieren automatisch angewendet. Wenn dies nicht ausreicht, verarbeitet TET PDF IFilter das Dokument mit einem gründlicheren Reparaturmodus. Da dieser mehr Zeit benötigt, wird er nur in den Fällen angewendet, in denen der automatische Reparaturmodus nicht erfolgreich ist.

Wenn ein Dokument zwar erfolgreich geöffnet werden kann, aber eine oder mehrere beschädigte Seiten enthält, werden diese ignoriert und die verbleibenden Seiten verarbeitet. Für jede ignorierte Seite wird ein Eintrag in der Ereignisanzeige erzeugt.

2.4 Unicode-Nachbearbeitung

Der TET-Kern der die zugrunde liegende Engine zur PDF-Textextraktion von TET PDF IFilter implementiert, bietet umfangreiche Steuerungsmechanismen für die Unicode-Nachbearbeitung. Diese werden im TET-Handbuch ausführlich behandelt, das im TET PDF IFilter-Paket enthalten ist. Die wichtigsten Funktionen sind unten zusammengefasst.

Die Funktionen zur Unicode-Nachbearbeitung werden mit Hilfe von TET-Dokumentoptionen gesteuert. In TET PDF IFilter können diese Optionen mit dem Element *DocOptions* der XML-Konfigurationsdatei übergeben werden, z.B.

```
<Tet>
        <DocOptions>decompose={canonical=_all}</DocOptions>
        <PageOptions/>
        <TetOptions/>
</Tet>
```

2.4.1 Unicode-Folding

Beim Folding werden ein oder mehrere Unicode-Zeichen verarbeitet und eine bestimmte Aktion für jedes der Zeichen ausgeführt. Folgende Aktionen sind verfügbar:

- Erhalten des Zeichens;
- Entfernen des Zeichens;
- Ersetzen des Zeichens durch ein anderes (festes) Zeichen.

Foldings sind nicht verkettet: die Ausgabe eines Foldings wird nicht weiter durch die verfügbaren Foldings verarbeitet. Foldings betreffen nur Unicode-Textausgabe, jedoch nicht die Glyphen, die von der Struktur *TET_char_info* oder den <*Glyph*>-Elementen in TETML zurückgegeben werden. Wenn z.B. bestimmte Unicode-Zeichen beim Folding entfernt werden, werden die zugeordneten Glyphen der ursprünglichen Zeichen trotzdem ausgegeben.

Um die Lesbarkeit zu verbessern, werden in den Beispielen der folgenden Tabellen isolierte Unteroptionen der Optionsliste *fold* aufgelistet. Beachten Sie, dass diese Unteroptionen bei Anwendung mehrerer Foldings in einer großen Optionsliste zusammengefasst werden müssen; die Option *fold* darf nicht mehrfach übergeben werden. Das folgende Beispiel ist falsch:

```
fold={ {[:blank:] U+0020} } fold={ {_dehyphenation remove} }        FALSCH!
```

Die folgende Optionsliste zeigt die korrekte Syntax für mehrfache Foldings:

```
fold={ {[:blank:] U+0020 } {_dehyphenation remove} }
```

Beispiele für Foldings. Tabelle 2.3 zeigt Beispiele für die Option *fold* für verschiedene Anwendungen von Foldings. Die Beispieloptionen müssen in der Dokument-Optionsliste übergeben werden. TET kann Foldings auf eine ausgewählte Teilmenge aller Unicode-Zeichen anwenden. Diese werden Unicode-Mengen genannt; für die entsprechende Syntax siehe das TET-Handbuch.

Tabelle 2.3 Beispiele für die Option `fold`

Beschreibung und Optionsliste	***vor dem Folding***	***nach dem Folding***
Entfernen aller Zeichen in einer Unicode-Menge		
In der Ausgabe bleiben nur die in ISO 8859-1 (Latin-1) definierten Zeichen erhalten, d.h. alle Zeichen außerhalb des Unicode-Blocks Latin-1 werden entfernt: `fold={{[^U+0020-U+00FF] remove}}`	Ą U+0104	*n/a*
Entfernen aller nicht alphabetischen Zeichen (z.B. Satzzeichen, Zahlen): `fold={{[:Alphabetic=No:] remove}}`	7 U+0037 A U+0041	*n/a* A U+0041
Entfernen aller Zeichen außer Zahlen: `fold={{[^[:General_Category=Decimal_Number:]] remove}}`	7 U+0037 A U+0041	7 U+0037 *n/a*
Entfernen aller unbekannten Zeichen, also PUA-Zeichen und Zeichen, für die kein geeigneter Unicode-Wert ermittelt werden konnte (die übrigen Standard-Foldings werden erneut aktiviert): `fold={{[:Private_Use:] remove} {[U+FFFD] remove} default}`	U+FFFF	*n/a*
Entfernen aller Gedankenstriche und Satzzeichen: `fold={{[:General_Category=Dash_Punctuation:] remove}}`	- U+002D	*n/a*
Entfernen aller Bidi-Steuerzeichen: `fold={{[:Bidi_Control:] remove}}`	U+200E	*n/a*
Entfernen aller Variantenselektoren für Standard oder Ideographic Variation Sequences (IVS) unter Beibehaltung aller internen Foldings: `fold={{[[\uFE00-\uFE0F][\U000E0100-\U000E01EF]] remove} default}`	≨ U+2268 FE 65 U+FE00	≨ U+2268
Ersetzen aller Zeichen in einer Unicode-Menge durch ein anderes Zeichen		
Leerzeichen-Folding: alle Varianten von Unicode-Leerzeichen werden auf U+0020 abgebildet: `fold={{[:blank:] U+0020}}`	U+00A0	U+0020
Folding für Gedanken-/Bindestriche: alle Varianten von Unicode-Gedanken-/Bindestrichen werden auf U+002D abgebildet: `fold={{[:Dash:] U+002D}}`	‐ U+2011	- U+002D
Ersetzen aller unbelegten Zeichen (d.h. Unicode-Codepunkten, denen kein Zeichen zugewiesen ist) durch U+FFFD: `fold={{[:Unassigned:] U+FFFD}}`	U+03A2	? U+FFFD
Verarbeitung spezieller Zeichen		
Erhalten aller Trennstriche am Zeilenende unter Beibehaltung der Standard-Foldings. Da diese Zeichen intern in TET identifiziert werden (denn sie haben keine bestimmte Unicode-Property) wird das Folding für den Bereich mit Hilfe des Schlüsselworts `_dehyphenation` *identifiziert:* `fold={{_dehyphenation preserve}}`	- U+002D	- U+002D
Erhalten von arabischen Tatweel-Zeichen (die standardmäßig entfernt werden): `fold={{[U+0640] preserve}}`	ـ U+0640	ـ U+0640
Ersetzen verschiedener Interpunktionszeichen durch die entsprechenden ASCII-Zeichen: `fold={ {[U+2018] U+0027} {[U+2019] U+0027} {[U+201C] U+0022} {[U+201D] U+0022}}`	“ U+201C	" U+0022

2.4.2 Unicode-Dekomposition

Dekompositionen ersetzen ein Zeichen durch eine äquivalente Folge von einem oder mehreren anderen Zeichen. Ein Unicode-Zeichen wird als (entweder kompatibel oder kanonisch) äquivalent zu einem anderen Zeichen oder einer Folge von Zeichen bezeichnet, wenn sie die gleiche Bedeutung haben, aber aus historischen Gründen (meist im Zusammenhang mit Round Tripping bei Legacy-Encodings) unterschiedlich in Unicode kodiert werden. Dekompositionen zerstören Information. Dies ist nützlich, wenn Sie nicht am Unterschied zwischen dem ursprünglichen Zeichen und seinem Äquivalent interessiert sind. Wenn Sie jedoch an diesem Unterschied interessiert sind, sollten Sie die entsprechende Dekomposition nicht anwenden. Für weitere Informationen zu Unicode-Dekompositionen siehe
www.unicode.org/versions/Unicode8.0.0/ch02.pdf (Abschnitt 2.12) und
www.unicode.org/versions/Unicode8.0.0/ch03.pdf (Abschnitt 3.7)

Hinweis *Der Begriff »Dekomposition« wird hier verwendet, wie im Unicode-Standard definiert, obwohl viele Dekompositionen ein Zeichen gar nicht in Einzelteile zerlegen, sondern es in ein anderes Zeichen konvertieren.*

Kanonische Dekomposition. Wenn Zeichen oder Zeichenfolgen kanonisch äquivalent sind, dann stellen sie exakt das gleiche abstrakte Zeichen dar und sollten deshalb immer das gleiche Aussehen und Verhalten zeigen. Typische Beispiele sind vorkombinierte Zeichen (z.B. Ä U+00C4) im Gegensatz zu kombinierenden Sequenzen (z.B. A U+0041 ¨ U+0308): beide Darstellungen sind kanonisch äquivalent. Durch den Wechsel von einer Darstellungsform zur anderen wird keine Information entfernt. Bei kanonischer Dekomposition wird eine Darstellung durch eine andere ersetzt, die als kanonische Darstellung bezeichnet wird.

In den Unicode-Codetabellen[1] (nicht jedoch in den Zeichentabellen) sind kanonische Zuordnungen mit dem Symbol IDENTICAL TO ≡ U+2261 versehen. Der Dekompositionsname <*canonical*> wird implizit angenommen. Tabelle 2.4 enthält verschiedene Beispiele.

Die folgenden Dokumentoptionen ordnen alle kanonischen Äquivalente ihren äquivalenten Entsprechungen zu:

```
decompose={canonical=_all}
```

Kompatibilitätsdekomposition. Wenn Zeichen kompatibel äquivalent sind, dann stellen sie zwar das gleiche abstrakte Zeichen dar, können sich aber in Aussehen und Verhalten unterscheiden. Typische Beispiele sind isolierte Formen arabischer Zeichen (z.B. س U+0633) im Gegensatz zu kontextspezifischen Formen (z.B. س U+FEB2 , سـ U+FEB4 , ـس U+FEB3). Kompatibel äquivalente Zeichen unterscheiden sich in der Formatierung. Das Entfernen dieser Formatierung bedeutet zwar einen Informationsverlust, kann aber die Verarbeitung bei bestimmten Anwendungen (z.B. Suchläufen) vereinfachen. Kompatibilitätsdekomposition entfernt die Formatierungsinformationen.

1. Siehe www.unicode.org/charts/

Tabelle 2.4 Kanonische Dekomposition: Unteroption für die Option decompose *(kanonisch äquivalente Zeichen sind in den Unicode-Codetabellen mit dem Symbol IDENTICAL TO* ≡ U+2261 *gekennzeichnet)*

Dekomposition	*Beschreibung*	*vor der Dekomposition*	*nach der Dekomposition*
canonical[1]	*Kanonische Dekomposition*	À U+00C0	A ◌̀ U+0041 U+0300
		林 U+F9F4	林 U+6797
		Ω U+2126	Ω U+03A9
		ば U+3070	は ◌゙ U+306F U+3099
		אַ U+FB2F	א ◌ַ U+05D0 U+05B8

1. Damit bestimmte Zeichen erhalten bleiben, wird diese Dekomposition standardmäßig nicht auf alle Zeichen angewendet; für weitere Informationen siehe das TET-Handbuch.

In den Unicode-Codetabellen sind kompatible Zuordnungen mit dem Symbol ALMOST EQUAL TO ≈ U+2248 versehen, gefolgt vom Dekompositionsnamen (oder »Tag«) in spitzen Klammern, z.B. <*noBreak*>. Wird kein Tag-Name angegeben, wird <*compat*> angenommen. Die Tag-Namen sind identisch mit den Optionsnamen in Tabelle 2.5. Wie einige Beispiele zeigen, kann durch die Dekomposition ein einzelnes Zeichen in eine Folge mehrerer Zeichen konvertiert werden.

Hinweis *Alle Einträge in Tabelle 2.5 beschreiben Kompatibilitätsdekompositionen, wobei das Tag* compat *nur »sonstige« Kompatibilitätsdekompositionen enthält, d.h. solche ohne einen bestimmten Namen.*

Hinweis *Beachten Sie, dass Glyphen in einem PDF-Dokument manchmal bereits auf eine zerlegte Sequenz statt auf die unzerlegten Unicode-Werte abgebildet werden. In diesem Fall hat die Option* decompose *keine Wirkung auf die Ausgabe.*

Beispiele für Dekompositionen. Dekompositionen können in TET mit der Dokumentoption *decompose* gesteuert werden. Eine Dekomposition kann auf bestimmte Unicode-Zeichen beschränkt werden. Die Teilmenge der Zeichen, auf die die Dekomposition angewendet wird, heißt Domäne. Tabelle 2.5 führt die Unteroptionen für alle Unicode-Dekompositionen mit Beispielen auf.

Die folgenden Beispiele für die Option *decompose* müssen in der Optionsliste von *TET_open_document( )* übergeben werden. Die Namen der Dekompositionen in der Optionsliste *decompose* sind Tabelle 2.5 entnommen.

Deaktivieren aller Dekompositionen:

```
decompose={none}
```

Erhalten breiter Zeichen (double-byte oder zenkaku) und hankaku (schmaler) Zeichen:

```
decompose={wide=_none narrow=_none}
```

Tabelle 2.5 Kanonische Dekomposition: Unteroptionen für die Option decompose *(kanonisch äquivalente Zeichen werden in den Unicode-Codetabellen mit dem Symbol ALMOST EQUAL TO* ≈ U+2248 *gekennzeichnet)*

Dekomposition	*Beschreibung*	*vor der Dekomposition*	*nach der Dekomposition (in logischer Anordnung)*
circle	*Eingekreiste Zeichen*	㉑ U+3251	2 1 U+0032 U+0031
compat[1]	*Sonstige Kompatibilitätsdekompositionen, z.B. gängige Ligaturen*	ﬁ U+FB01	f i U+0066 U+0069
final	*Finale Präsentationsformen, besonders Arabisch*	ﺱ U+FEB2	س U+0633
font	*Fontvarianten, z.B. mathematische Mengenzeichen, hebräische Ligaturen*	ℂ U+2102	C U+0043
fraction[1]	*Gemeine Brüche*	¼ U+00BC	1 ⁄ 4 U+0031 U+2044 U+0034
initial	*Initiale Präsentationsformen, besonders Arabisch*	ﺳ U+FEB3	س U+0633
isolated	*Isolierte Präsentationsformen, besonders Arabisch*	ﺴﺮ U+FD0E	س ر U+0633 U+0631
medial	*Mittlere Präsentationsformen, besonders Arabisch*	ﺴ U+FEB4	س U+0633
narrow	*Schmale (hankaku) Kompatibilitätszeichen*	ｦ U+FF66	ヲ U+30F2
nobreak	*Umbruchgeschützte Leerzeichen*	U+00A0	U+0020
none	*Deaktivieren aller nicht explizit in der Optionsliste* decompose *angegebenen Dekompositionen*	*(alle Zeichen bleiben unverändert)*	
small	*Kleine Formen für die Kompatibilität zu CNS 11643*	﹐ U+FE50	, U+002C
square	*eckige CJK-Fontvarianten*	㌔ U+3314	キ 口 U+30AD U+30ED
sub[1]	*Tiefgestellte Zeichen*	₁ U+2081	1 U+0031
super[1]	*Hochgestellte Zeichen*	ª U+00AA ™ U+2122	a U+0061 T M U+0054 U+004D
vertical	*Vertikale Layout-Präsentationsformen*	ﹷ U+FE37	{ U+007B
wide	*Breite (zenkaku) Kompatibilitätsformen*	￡ U+FFE1	£ U+00A3

1. Damit bestimmte Zeichen erhalten bleiben, wird diese Dekomposition standardmäßig nicht auf alle Zeichen angewendet; für weitere Informationen siehe das TET-Handbuch.

Zuordnen aller kanonischen Äquivalente auf ihre Entsprechungen:

```
decompose={canonical=_all}
```

Mit der folgenden Optionsliste lässt sich die Dekomposition *circle* aktivieren, alle anderen Dekompositionen werden deaktiviert:

```
decompose={none circle=_all}
```

Im Gegensatz dazu lassen sich mit der folgenden Optionsliste alle Dekompositionen aktivieren (da das Weglassen aller anderen Optionen das Standardverhalten aktiviert):

```
decompose={circle=_all}
```

2.4.3 Unicode-Normalisierung

Der Unicode-Standard definiert vier Normalformen, die auf den Konzepten der kanonischen bzw. kompatiblen Äquivalenz beruhen. Alle Normalformen platzieren kombinierende Zeichen in einer definierten Reihenfolge und wenden Zerlegung (Dekomposition) und Zusammensetzung (Komposition) auf unterschiedliche Weise an:

- Normalform C (NFC): kanonische Dekomposition gefolgt von kanonischer Komposition
- Normalform D (NFD): kanonische Dekomposition
- Normalform KC (NFKC): Kompatibilitätsdekomposition gefolgt von kanonischer Komposition
- Normalform KD (NFKD): Kompatibilitätsdekomposition

Die Normalformen sind im Unicode Standard Annex #15 »Unicode Normalization Forms« spezifiziert (siehe *www.unicode.org/versions/Unicode5.2.0/ch03.pdf#G21796* und *www.unicode.org/reports/tr15/)*.

TET PDF IFilter unterstützt alle vier Normalformen. Die Unicode-Normalisierung lässt sich mit der Dokumentoption *normalize* steuern, z.B.

```
normalize=nfc
```

TET PDF IFilter wendet standardmäßig keine Normalisierung an. Wegen der möglichen Wechselwirkung zwischen den Optionen *decompose* und *normalize* werden die Standard-Dekompositionen deaktiviert, wenn die Option *normalize* auf einen anderen Wert als *none* gesetzt wird.

Die Wahl der Normalform hängt von den Anforderungen der Anwendung ab. Beispielsweise erwarten einige Datenbanken Text im Format NFC, was auch das gängige Format für Unicode-Text im Web ist. Tabelle 2.6 zeigt die Auswirkung der Normalisierung auf verschiedene Zeichen.

Tabelle 2.6 Beispiele für Unicode-Normalformen

vor der Normalisierung	***NFC***	***NFD***	***NFKC***	***NFKD***
Ä U+00C4	Ä U+00C4	A ¨ U+0041 U+0308	Ä U+00C4	A ¨ U+0041 U+0308
A ¨ U+0041 U+0308	Ä U+00C4	A ¨ U+0041 U+0308	Ä U+00C4	A ¨ U+0041 U+0308
¨ A U+0308 U+0041	¨ A U+0308 U+0041	¨ A U+0308 U+0041	¨ A U+0308 U+0041	¨ A U+0308 U+0041
fi U+FB01	fi U+FB01	fi U+FB01	f i U+0066 U+0069	f i U+0066 U+0069
3 5 U+0033 U+2075	3 5 U+0033 U+2075	3 5 U+0033 U+2075	3 5 U+0033 U+0035	3 5 U+0033 U+0035
Å U+212B	Å U+00C5	A ° U+0041 U+030A	Å U+00C5	A ° U+0041 U+030A

Tabelle 2.6 Beispiele für Unicode-Normalformen

vor der Normalisierung	*NFC*	*NFD*	*NFKC*	*NFKD*
™ U+2122	™ U+2122	™ U+2122	T M U+0054 U+004D	T M U+0054 U+004D
Ⅳ U+2163	Ⅳ U+2163	Ⅳ U+2163	I V U+0049 U+0056	I V U+0049 U+0056
רּ U+FB48	ג · U+05E8 U+05BC	ג · U+05E8 U+05BC	ג · U+05E8 U+05BC	ג · U+05E8 U+05BC
가 U+AC00	가 U+AC00	ᄀ ᅡ U+1100 U+1161	가 U+AC00	ᄀ ᅡ U+1100 U+1161
ぢ U+3062	ぢ U+3062	ち ゙ U+3061 U+3099	ぢ U+3062	ち ゙ U+3061 U+3099
㉙ U+32C9	㉙ U+32C9	㉙ U+32C9	1 0 月 U+0031 U+0030 U+6708	1 0 月 U+0031 U+0030 U+6708

2.5 Benutzerdefinierte Tabellen für das Mapping von Glyphen

Obwohl in TET PDF IFilter viele Workarounds implementiert sind, kann es in seltenen Fällen vorkommen, dass Text aus PDF-Dokumenten nicht korrekt extrahiert werden kann, wenn wichtige Informationen für die Unicode-Zuordnung im Dokument fehlen. Wenn Sie viele Dokumente mit ähnlichen Eigenschaften haben (beispielsweise mit der gleichen Software und den gleichen Fonts erstellt), können Sie Zusatztabellen für die Unicode-Zuordnung übergeben, um Text aus PDF-Dokumenten zu extrahieren, der sonst nicht indiziert werden könnte.

TET PDF IFilter unterstützt verschiedene Tabellenformate, die in der Dokumentation für PDFlib TET ausführlich beschrieben werden. Sie können solche Mapping-Tabellen auch mit dem frei verfügbaren PDFlib FontReporter und dem PDFlib TET Plugin für Adobe Acrobat verwenden. Tabellen für die Unicode-Zuordnung müssen in der Konfigurationsdatei mit geeigneten Dokumentoptionen konfiguriert werden und können im Verzeichnis *resource* des Installationsverzeichnisses von TET PDF IFilter platziert werden.

XML-Konfiguration für TET-Optionen. Optionslisten zur Steuerung des TET-Kerns (z.B. bei Verwendung von benutzerdefinierten Tabellen zum Glyphen-Mapping) müssen entsprechend der in der TET-Produktdokumentation beschriebenen Syntax für Optionslisten aufgebaut werden und können an TET PDF IFilter in den Elementen *DocOptions*, *PageOptions* und *TetOptions* der XML-Konfigurationsdatei übergeben werden, die alle Unterelemente des Elements *Tet* sind:

```
<Tet>
        <DocOptions>glyphmapping {{fontname=T* glyphlist={tex}}}</DocOptions>
        <PageOptions/>
        <TetOptions>searchpath={C:/glyphlists}</TetOptions>
</Tet>
```

3 Indizierung von Metadaten

3.1 Metadaten-Quellen in PDF

Die meisten PDF-Dokumente enthalten Dokument-Infofelder wie *Verfasser* oder *Titel*. Zusätzlich können PDF-Dokumente auch XMP-Metadaten enthalten. TET PDF IFilter unterstützt die Indizierung von verschiedenen Arten von Metadaten in PDF-Dokumenten.

Standard- und benutzerdefinierte Dokument-Infofelder. Dokument-Infofelder stellen die einfache, herkömmliche Art von PDF-Metadaten dar. In Acrobat können sie über den Menüeintrag *Datei, Eigenschaften...* angezeigt werden. Folgende Dokument-Infofelder sind in PDF vordefiniert:

```
Titel, Verfasser, Thema, Stichwörter, Erstellt am, Erstellt von, Geändert am, Trapped
```

Zusätzlich können der Menge an Dokument-Infofeldern benutzerdefinierte Einträge hinzugefügt werden. Sie lassen sich in Acrobat (jedoch nicht in Adobe Reader) über *Datei, Eigenschaften..., Benutzerdefiniert* anzeigen.

Sowohl Standard- als auch benutzerdefinierte Dokument-Infofelder können über die pCOS-Pfade in TET PDF IFilter adressiert werden (siehe unten).

XMP-Properties auf Dokument- und Bildebene. XMP *(Extensible Metadata Platform[1])* ist ein Framework für Metadaten. XMP ist zum Beispiel erforderlich für die Konformität zu PDF/A und wird von einer wachsenden Anzahl von Anwendungen unterstützt. XMP-Metadaten sind in Schemas organisiert, die eine Reihe von Properties (Eigenschaften) enthalten. Properties werden über das Präfix des Namensraums und den Namen der Property adressiert.

XMP-Metadaten sind in der Regel dem gesamten Dokument zugeordnet. In PDF können sie jedoch auch einzelnen Seiten, Bildern oder anderen Objekten zugeordnet werden. In der Praxis wird diese Funktionalität hauptsächlich für Rasterbilder verwendet. Ein Digitalfoto kann zum Beispiel den Namen des Fotografen, Copyright-Hinweise, Szenendetails und andere Informationen enthalten. Eine wesentliche Quelle für XMP-Metadaten auf Bildebene ist Adobe Photoshop. Wenn Sie mit Photoshop Bilder erzeugen oder diese in Adobe-Workflows verwenden, werden die XMP-Metadaten normalerweise dem PDF-Dokument hinzugefügt und können mit TET PDF IFilter indiziert werden.

Die XMP-Spezifikation enthält eine Beschreibung aller vordefinierten XMP-Schemas und -Properties. Für weitere Informationen zu XMP siehe

www.adobe.com/devnet/xmp

XMP-Properties auf Dokument- oder Bildebene (sowie anderen Objekten zugeordnetes XMP) können in zwei Schritten adressiert werden: ein pCOS-Pfad identifiziert das relevante PDF-Objekt und der aus zwei Teilen bestehende Name der XMP-Property adressiert das XMP-Zielschema und die Property.

XMP-Metadaten können in Acrobat mit benutzerdefinierten XMP-Panels angezeigt und bearbeitet werden. Dabei handelt es sich um eine Benutzeroberfläche für XMP-

1. Siehe www.adobe.com/products/xmp

Metadaten. Beispiele für XMP-Panels werden mit TET PDF IFilter installiert; weitere Informationen hierzu sowie Installationsanweisungen finden Sie unter

www.pdflib.com/knowledge-base/xmp-metadata/xmp-panels/

Erweiterte pCOS-Pfade. Die pCOS-Schnittstelle *(PDFlib Comprehensive Object Syntax)* bietet eine einfache und elegante Lösung für das Abrufen von beliebigen Informationen aus allen Bereichen eines PDF-Dokuments, mit Ausnahme der Seiteninhalte, wie zum Beispiel Seitengröße, Metadaten, interaktive Elemente usw. Beispiele für die Verwendung von pCOS-Pfaden sowie eine Syntaxbeschreibung finden Sie in der pCOS Pfadreferenz, die als separates Dokument erhältlich ist. Zusätzliche Beispiele finden Sie im pCOS Cookbook unter *www.pdflib.com/pcos-cookbook/*.

Mit pCOS lassen sich in TET PDF IFilter Informationen über ein Dokument adressieren. pCOS-Pfade stellen bestimmte Elemente von PDF-Dokumenten wie Lesezeichen, Fontnamen oder Seitengrößen dar. Sie können auch Dokument-Infofelder oder XMP-Metadaten adressieren (jedoch keine Properties innerhalb eines XMP-Streams). pCOS-API-Funktionen stehen in TET PDF IFilter zwar nicht zur Verfügung, sie können pCOS-Pfade aber als Ausdrücke in der XML-Konfigurationsdatei übergeben, um Informationen zu einem Dokument zu indizieren.

In Tabelle 3.1 werden pCOS-Pfade für oft verwendete PDF-Objekte aufgeführt. Viele pCOS-Objekte werden durch Arrays dargestellt, die einen Array-Index in eckigen Klammern erfordern, z.B. bedeutet *pages[0]* die erste Seite (Seitenzahlen beginnen bei 0). Zusätzlich zu den von TET (der Basis von TET PDF IFilter) unterstützten pCOS-Pfaden unterstützt der IFilter folgende Syntaxerweiterungen für pCOS-Pfade: Statt eines Arrays oder eines Dictionary-Index können Sie das Wildcard-Zeichen »*« (einzelner Stern) verwenden. Damit durchläuft TET PDF IFilter alle möglichen Werte des Array-Index und schließt alle Objektwerte in die Indizierung mit ein. Innerhalb eines einzelnen pCOS-Pfades kann eine beliebige Anzahl von Wildcards verwendet werden.

Tabelle 3.1 pCOS-Pfade für verschiedene PDF-Objekte

pCOS-Pfad	Typ	Beschreibung
length:pages	*Zahl*	*Anzahl der Seiten im Dokument*
/Info/Title	*String*	*Standard-Dokument-Infofeld Titel*
/Info/ArticleNumber	*String*	*benutzerdefiniertes Dokument-Infofeld* `ArticleNumber` *(Dokument-Infofelder können beliebige Namen verwenden)*
/Root/Metadata	*Stream*	*XMP-Stream mit den Dokument-Metadaten*
images[*]/Metadata	*Stream*	*XMP-Metadaten-Streams für alle Bilder im Dokument*
fonts[*]/name	*String*	*Fontname*
length:fonts	*String*	*Anzahl der Fonts in einem Dokument*
length:images	*String*	*Anzahl der Bilder in einem Dokument*
fonts[*]/embedded	*boolean*	*Status der Font-Einbettung*
pages[*]/width	*Zahl*	*Breite des sichtbaren Bereichs der Seite*
pages[*]/annots[*]/A/URI	*String*	*Ziel-URL des Weblinks auf allen Seiten*
fields[*]/V	*String*	*Textinhalte (Wert) aller Formularfelder auf allen Seiten im Dokument*

XML-Konfiguration für Metadaten-Quellen. Quellen für Metadaten-Properties können im Element *Source* (Unterelement von *Property)* der XML-Konfigurationsdatei angegeben werden. Das Attribut *pdfObject* enthält einen erweiterten pCOS-Pfad für ein PDF-Objekt, das Attribut *xmpName* enthält das Schema-Präfix und den Namen einer XMP-Property:

```
<Source pdfObject="/Info/ArticleNumber"/>
<Source xmpName="acme:number"/>
```

3.2 Struktur von Metadaten

Metadaten sind hierarchisch angeordnet:

- *Properties* bilden die grundlegenden Bausteine für Metadaten. Unter Windows und der IFilter-Schnittstelle werden Properties (Eigenschaften) durch einen eindeutigen numerischen Wert identifiziert (siehe unten).
- *Property-Sets* sind Gruppen von Properties, die in der Regel eine gewisse logische Beziehung zueinander haben. Alle Properties eines Property-Sets tragen die gleiche GUID (siehe unten). Property-Sets können in der XML-Konfigurationsdatei definiert werden.
- *Sammlungen von Property-Sets* bestehen aus einer Gruppe von Property-Sets. In TET PDF IFilter sind verschiedene vordefinierte Sammlungen von Property-Sets implementiert. Damit lassen sich mehrere Property-Sets gemeinsam aktivieren oder deaktivieren. Es ist nicht erforderlich, weitere Sammlungen von Property-Sets zu konfigurieren.

Identifikation von Properties und GUIDs. Properties werden in der IFilter-Schnittstelle durch eine aus zwei Teilen bestehende ID identifiziert:

- Den ersten Teil bildet der *Globally Unique Identifier* GUID (manchmal auch *Universally Unique Identifier, UUID* , genannt), ein eindeutiger 128-Bit-Identifier, dessen Wert für alle Properties eines Property-Sets identisch ist. Für weitere Informationen zu GUIDs siehe

  www.itu.int/ITU-T/studygroups/com17/oid/X.667-E.pdf

  Zur Erzeugung von GUIDs stehen mehrere Tools zur Verfügung; Sie können auch Online-Dienste verwenden, siehe z.B. unter

  www.itu.int/ITU-T/asn1/uuid.html

  Eine GUID sieht zum Beispiel folgendermaßen aus:
  7a737220-0cd0-11dd-bd75-0002a5d5c51b.
- Durch den zweiten Teil der ID wird die Property innerhalb des Property-Sets eindeutig identifiziert. Er kann aus einer positiven Ganzzahl namens *identifier* oder kurz *ID* bestehen. Property-Identifier in einem Property-Set müssen mit dem Wert 2 beginnen, sind aber ansonsten beliebig. Sie werden von allen IFilter-Clients unterstützt. Alternativ kann der zweite Teil auch ein Name in Klartext sein. Die Verwendung von Namen statt ID ist veraltet und wird von manchen IFilter-Clients, wie z.B. Windows Search, nicht mehr unterstützt. Allerdings macht es die Konfiguration bequemer, so kann diese Methode beispielsweise noch im IFilter-Client SharePoint verwendet werden. Für Informationen zur Aktivierung von GUID+Namen siehe »XML-Konfiguration für die Verarbeitung von GUID+Name von Properties«, Seite 47.

Die Kombination aus GUID+ID oder GUID+Name ist erforderlich, um die Abfrage von Metadaten-Properties in Suchprodukten zu konfigurieren. Für weitere Informationen zu Metadaten-Properties siehe Abschnitt 3.4, »Benutzerdefinierte Metadaten-Properties«, Seite 46.

3.3 Vordefinierte Metadaten-Properties

In TET PDF IFilter sind folgende Sammlungen von Property-Sets integriert:

- *Shell-Properties* sind in Windows gebräuchlich und haben benutzerfreundliche Namen. TET PDF IFilter füllt alle Shell-Properties, die Äquivalente in PDF-Dokumente haben. Häufig verwendete Beispiele sind *System.Author*, *System.Title* und *System.-Document.DateCreated*. Für eine Liste und Beschreibung von Shell-Properties siehe

  msdn.microsoft.com/en-us/library/windows/desktop/dd561977(v=vs.85).aspx

- *PDF-Eigenschaften* gibt es nur für PDF-Dokumente. Sie werden mit Hilfe von pCOS-Pfaden gefüllt. Beispiele hierfür sind die PDF-Versionsnummer, Lesezeichentext oder Seitengröße.
- *XMP-Properties auf Dokumentebene* umfassen alle vordefinierten Dokument-Properties in der XMP-Spezifikation 2005, siehe

  www.adobe.com/devnet/xmp

- *XMP-Properties auf Bildebene* umfassen XMP-Properties, die an die Bilder im Dokument angefügt sind. Diese werden ebenfalls aus der XMP-Spezifikation 2005 abgeleitet.
- *Interne Eigenschaften* sind Hilfseigenschaften für Entwicklung und Fehlersuche, die nicht für den produktiven Einsatz bestimmt sind. Sie enthalten unter anderem die Software-Version sowie den Zeitpunkt der Indizierung.

Eine komplette Liste aller vordefinierten Properties zusammen mit ihren GUIDs und benutzerfreundlichen Namen finden Sie in Anhang A, »Vordefinierte Metadaten-Properties«, Seite 81.

Die Properties aus diesen Sammlungen müssen nicht separat konfiguriert werden. Da jedoch nicht alle Sammlungen standardmäßig aktiviert werden, müssen Sie diese bei Bedarf aktivieren. Vordefinierte Properties können Sie überschreiben, indem Sie die zugehörige GUID+ID-Kombination einer benutzerdefinierten Property zuweisen.

XML-Konfiguration zur Aktivierung von Property-Set-Sammlungen. Vordefinierte Sammlungen von Property-Sets können Sie mit den zugehörigen Attributen im Element *Metadata/PropertySetCollection* aktivieren:

```
<PropertySetCollection
        shell="true"
        pdf="true"
        documentXmp="true"
        imageXmp="true"
        internal="true"/>
```

Die Property-Set-Sammlungen *Shell* und *Intern* sind standardmäßig aktiviert, *PDF*, *documentXMP* und *imageXMP* sind deaktiviert. Benutzerdefinierte Properties werden implizit aktiviert, wenn ein oder mehrere benutzerdefinierte Properties angegeben werden.

Mit dem Attribut *metdataHandling* des Elements *Filtering* kann die Verarbeitung aller Standard- und benutzerdefinierten Metadaten-Properties vollständig deaktiviert werden:

```
<Filtering metadataHandling="ignore">
```

3.4 Benutzerdefinierte Metadaten-Properties

Benutzerdefinierte Metadaten-Properties können zusätzlich zu den vordefinierten Properties für besondere Bedürfnisse z.B. innerhalb eines Unternehmens, einer Organisation oder Branche festgelegt werden. TET PDF IFilter bietet alle Möglichkeiten zur Steuerung benutzerdefinierter Properties: Sie können diese in der Konfigurationsdatei angeben, so dass sie von TET PDF IFilter erzeugt und von der Suchmaschine indiziert werden.

Planen von benutzerdefinierten Metadaten-Properties. Beachten Sie beim Anlegen von benutzerdefinierten Properties folgende Punkte (für weitere Informationen zu GUIDs, IDs und benutzerfreundlichen Namen siehe »Identifikation von Properties und GUIDs«, Seite 44):

- Properties können zu Property-Sets zusammengefasst werden. Jede Property benötigt einen eindeutigen 128-Bit-Identifier namens GUID.
- Die Property *identifier* ist eine eindeutige Ganzzahl (Integer), die die Property innerhalb ihrem Property-Set identifiziert. Die Identifier in einem Property-Set beginnen mit dem Wert 2. Bei einigen IFilter-Clients können die Identifier durch benutzerfreundliche Namen ersetzt werden. Vordefinierte Properties können Sie überschreiben, indem Sie die zugehörige GUID+ID-Kombination einer benutzerdefinierten Property zuweisen.
- Der *friendly name* für eine Property ist optional, falls ein Identifier verfügbar ist, und andernfalls erforderlich. Er kann ein beliebiger Name sein, der innerhalb der Konfigurationsdatei eindeutig sein muss. Einige IFilter-Clients akzeptieren ihn anstatt des Identifiers, benutzerfreundliche Namen werden jedoch nicht von allen IFilter-Clients unterstützt.
- Property *source:* Properties können aus den Dokument-Metadaten oder den allgemeinen PDF-Informationen gemäß Abschnitt 3.1, »Metadaten-Quellen in PDF«, Seite 41 entnommen werden.
- Der *Datentyp* der Property: *Int32* (32-Bit-Integer), *Double* (doppelt genaue Fließkommazahl), *Boolean* (true/false), *DateTime* (siehe unten) und *String*.
- Die Regel *precedence*: Bei mehr als einer Datenquelle für die Property können Sie festlegen, ob die erste nicht leere Datenquelle Vorrang hat (d.h. nachfolgende Quellen werden ignoriert) oder ob die Daten aus allen nicht leeren Quellen gesammelt werden.
- Sie können angeben, ob die Property als *vector* ausgegeben wird, d.h. mehrere Werte werden in einer Array-Struktur an die IFilter-Schnittstelle übergeben und nicht als fester Wert (siehe Abschnitt 3.5, »Properties mit mehreren Werten«, Seite 49).
- Ein *Präfix,* das dem Property-Namen vorangestellt wird, falls Properties als Teil des Volltexts indiziert werden (siehe Abschnitt 3.6, »Indizieren von Metadaten-Properties als Text«, Seite 50).

Property-Typ DateTime. Zur Angabe eines bestimmten Zeitpunkts können Sie den Datentyp *DateTime* für Properties verwenden. Ob wohl die Eingabe je nach Quelle in unterschiedlichen Formaten vorliegen kann, erfolgt die Ausgabe für Properties vom Typ *DateTime* immer in dem für die IFilter-Schnittstelle erforderlichen Format:

- Handelt es sich bei der Datenquelle um einen pCOS-Pfad, z.B. für ein Standard- oder benutzerdefiniertes Dokument-Infofeld wie */Info/CustomDate,* muss der Wert für ein Datum im PDF-Standardformat angegeben werden, wie er in ISO 32000-1, Abschnitt

7.9.7 spezifiziert ist. Das Format eines PDF-Strings für das Datum ist *D:YYYYMMDDHHmmSSOHH'mm*. Beachten Sie, dass das PDF-Datumsformat (anders als das unten beschriebene XMP-Format) keine Sekundenbruchteile unterstützt. Zum Beispiel:

```
D:201109231858                  GMT
D:201109231058-08'00'           gleicher Zeitpunkt in U.S. Pacific Standard Time
D:201109231958+01'00'           gleicher Zeitpunkt in Central European Time
```

- Handelt es sich bei der Datenquelle um eine XMP-Property (z.B. *xmp:ModifyDate)*, muss der Wert für den grundlegenden Wertetyp *Date* wie in der XMP-Referenz 2005 angegeben werden, was dem in ISO 8601[1] spezifizierten Format entspricht. Das Format eines XMP-Strings für das Datum ist *YYYY-MM-DDThh:mm:ss.sTZD,* wobei *TZD* die Zeitzone (Time Zone Designator) bezeichnet *(Z* oder *+hh:mm* oder *-hh:mm).* Beachten Sie, dass manche Bestandteile optional sind: XMP-Datumsangaben unterstützen sechs Granularitätsstufen mit zunehmender Genauigkeit, während das PDF-Datumsformat nur eine einzige Granularitätsstufe kennt. Zum Beispiel:

```
2011-09-23T18:58:30Z            GMT
2011-09-23T13:58:30-05:00       gleicher Zeitpunkt in U.S. Eastern Standard Time
2011-09-23T13:58:30+01:00       gleicher Zeitpunkt in Central European Time
```

TET PDF IFilter normalisiert alle *DateTime*-Properties gemäß Spezifikation der IFilter-Schnittstelle zu UTC. Daher kann die Suche nach *DateTime*-Properties immer bezüglich der lokalen Zeitzone erfolgen; die Zeitzone, in der das PDF-Dokument erstellt wurde, spielt keine Rolle.

XML-Konfiguration für benutzerdefinierte Properties. Im Element *PropertySet* können ein oder mehrere benutzerdefinierte Properties angegeben werden, wobei jedes Element *Property* eine Property in einem Property-Set beschreibt:

```
<PropertySet guid="33333333-5354-4c72-992C-5DF2AA4E7CBA">
        <Property friendlyName="MailTo" identifier="2" type="String">
            <Source xmpName="acme:mailto"/>
        </Property>
</PropertySet>
```

Der selben Windows-Property können mehrere PDF-Quellen zugeordnet werden. Ist ein Element *Property* vorhanden, wird die Verarbeitung der angegebenen Property automatisch aktiviert. Mit dem Attribut *metdataHandling* des Elements *Filtering* kann die Verarbeitung aller Standard- und benutzerdefinierten Metadaten-Properties jedoch vollständig deaktiviert werden:

```
<Filtering metadataHandling="ignore">
```

XML-Konfiguration für die Verarbeitung von GUID+Name von Properties. TET PDF IFilter verwendet GUID+ID, um Properties an der IFilter-Schnittstelle zu identifizieren, sofern das Attribut *identifier* vorhanden ist. Benutzerdefinierte Properties, die kein Attribut *identifier*, sondern nur das Attribut *friendlyName* haben, werden stattdessen durch GUID+Name identifiziert. Um die Verwendung von GUID+Name auch für vordefinierte Properties zu aktivieren (sowie zur globalen Verwendung von GUID+Name), können Sie das Attribut *useIdentifier* des Elements *Filtering* verwenden:

1. Siehe www.w3.org/TR/NOTE-datetime

```
<Filtering useIdentifier="false">
```

Die für die vordefinierten Properties verwendeten Namen lauten ähnlich wie die Präfixe für Properties in Tabelle 3.2, außer dass das führende *TET_* mit Unterstrich '_' weggelassen wird (z.B. *System_Author, pdfversion, dc_contributor, photoshop_DateCreated)*.

Die Verwendung von GUID+Name statt GUID+ID für Properties ist in manchen Umgebungen, vor allem SharePoint, komfortabler.

3.5 Properties mit mehreren Werten

Metadaten-Properties können einen oder mehrere Werte haben. Properties mit einem Wert bestehen aus einem festen Wert, der das Dokument als Ganzes beschreibt. Beispiele für Properties mit einem einzigen Wert sind das Erstellungsdatum (Quellen: *xmp:CreateDate* und */Info/CreationDate)* sowie der eindeutige Dokument-Identifier *(dc:identifier)*.

Properties mit mehreren Werten können im Dokument mehrfach vorkommen. Beispiele hierfür sind die Liste aller Dokumentverfasser oder die Schlüsselwörter. Das mehrfache Auftreten einer Property kann verschiedene Gründe haben:

- Die Quelle der Property ist ein XMP-Container und kann daher mehrere Einträge enthalten, z.B. *dc:creator* ist in XMP vom Typ *Seq*.
- Die Property wird von einem pCOS-Pfad mit mehreren Werten über Wildcards befüllt, wobei die Wildcards eine beliebige Anzahl von Einzeleinträgen umfassen, z.B. *bookmarks[*]/Title*.
- Die Property wird aus mehr als einer Quelle befüllt und das Attribut *precedence* der Property hat den Wert *try-all*, z.B. *pdf:Keywords* und */Info/Keywords*. Die normale Methode des *precedence=first-wins* verarbeitet jedoch nur die erste nicht leere Quelle der Property .

Vektorbasierte Verarbeitung von Properties. TET PDF IFilter verarbeitet alle relevanten Quellen in der Property-Definition (abhängig vom Attribut *precedence*) und gibt so viele nicht leere Property-Werte zurück wie möglich. Mit anderen Worten, jeder Wert wird als einzelne Einheit an den IFilter-Client zurückgegeben. Property-Werte werden in beliebiger Reihenfolge zurückgegeben.

Alternativ können Properties mit mehreren Werten als Vektoren an den IFilter-Client übergeben werden. Dabei wird ein einzelnes Array-Element ausgegeben, das einen oder mehrere Werte enthält. Für die vektorbasierte Verarbeitung von Properties gibt es folgende Besonderheiten:

- SharePoint unterstützt Properties mit mehreren Werten nur, wenn sie vektorbasiert verarbeitet werden.
- Manche IFilter-Clients wie Windows Search unterstützen vektorbasierte Abfragen, wobei in einer einzigen Abfrage nach einem oder mehreren Werten in einer Vektor-Property mit mehreren Werten gesucht werden kann.

Beachten Sie, dass es zwei unterschiedliche Konzepte gibt: *multivalued* (mit mehreren Werten) bezieht sich auf die Quelle der Property, während *vector processing* (Vektorverarbeitung) sich auf die Art bezieht, wie Property-Werte an den IFilter-Client übergeben werden. Vektorbasierte Verarbeitung kann auf Properties mit mehreren Werten angewendet werden, selbst wenn sie nur einen Wert enthalten.

Einige der vordefinierten Properties sind mehrwertig (siehe Anhang A, »Vordefinierte Metadaten-Properties«, Seite 81).

XML-Konfiguration für vektorbasierte Properties. Vektorbasierte Verarbeitung für benutzerdefinierte Properties ässt sich mit dem Attribut *emitAsVector* des Elements *Property* aktivieren:

```
<Property friendlyName="MailTo" type="String" precedence="try-all" emitAsVector="true">
        <Source xmpName="acme:mailto"/>
        <Source xmpName="gov:mailto"/>
</Property>
```

3.6 Indizieren von Metadaten-Properties als Text

Die meisten Textsuchmaschinen unterstützen die Abfrage von Properties. Allerdings ist die Abfrage von Properties bei Retrieval-Produkten, die nur Volltextsuche unterstützen, wie z.B. SQL Server, nicht möglich. Auf der anderen Seite mag die Suche nach Properties unerwünscht sein, wenn Sie nach allen Treffern dieses Suchbegriffs suchen möchten, unabhängig davon, ob er im Dokumentinhalt oder einer Property auftritt. In beiden Fällen können Sie TET PDF IFilter anweisen, alle Properties in die Volltextsuche mit aufzunehmen. Um die Property vom tatsächlichen Inhalt des Dokuments zu unterscheiden, kann TET PDF IFilter den Property-Werten optional einen String voranstellen, um Properties leichter als solche zu identifizieren, falls die Suchmaschine das Abfragen von Properties nicht direkt unterstützt. Für die in Anhang A, »Vordefinierte Metadaten-Properties«, Seite 81 aufgeführten vordefinierten Properties werden feste Präfixe verwendet.

Zwar erlaubt die Indizierung von Properties als Text die Suche nach Properties in Umgebungen, die dies sonst nicht unterstützen, die Suche nach Properties ist in dem Fall jedoch eingeschränkt. Boolesche und andere Ausdrücke beispielsweise sind für Property-Werte nicht verfügbar.

XML-Konfiguration für die Indizierung von Metadaten als Text. Um Metadaten-Properties als Text zu indizieren, setzen Sie das Attribut *metadataHandling* des Elements *Filtering* auf *propertyAndText* (um die Properties mit dem Haupttext transparent zu verschmelzen) oder auf *propertyAndPrefixedText* (um die Properties mit einem Präfix zu identifizieren):

```
<Filtering metadataHandling="PropertyAndPrefixedText">
```

Das optionale Präfix, das benutzerdefinierten Properties beim Filtern des Dokuments vorangestellt wird, kann für benutzerdefinierte Properties im Attribut *textIndexPrefix* des Elements *Property* angegeben werden:

```
<Property friendlyName="Title" identifier="7" textIndexPrefix="TITLE_">
...
</Property>
```

Die Präfixe, die vordefinierten Properties vorangestellt werden, sind nach folgendem Schema aufgebaut:

```
TET_<property name>_
```

Punktzeichen '.' im Property-Namen werden durch Unterstriche ersetzt '_' (für Beispiele hierzu siehe Tabelle 3.2).

Tabelle 3.2 Präfixe für die Indizierung von Metadaten als Text

Sammlung von Property-Sets	Beispiel für Property-Namen	Präfixe für die Indizierung von Metadaten als Text
Shell	`System.Author`	`TET_System_Author_`
TET	`PDFlib.TETPDFIFilter.pdfversion`	`TET_pdfversion_`
XMP	`dc:contributor`	`TET_dc_contributor_`
Image	`photoshop:DateCreated`	`TET_photoshop_DateCreated_`

Szenario 1: Transparentes Verschmelzen von Metadaten-Properties mit dem Haupttext. Wenn Metadaten-Properties hinreichend unterscheidbaren Text zur Identifizierung des Zieldokuments enthalten, genügt es, die Properties im Volltextindex und diesen in die Standard-Volltextabfragen aufzunehmen. Wenn Sie zum Beispiel eine bestimmte Artikelnummer abfragen, spielt es keine Rolle, ob die Nummer im Haupttext des Dokuments oder in einer Metadaten-Property auftritt, solange die gesuchte Artikelnummer nur in einem bestimmten Dokument vorhanden ist. Es ist also egal, ob der Text im Haupttext oder in einer Metadaten-Property auftritt, Sie müssen lediglich die Indizierung von Properties als Volltext aktivieren.

Verwenden Sie zum transparenten Verschmelzen der Properties mit dem Haupttext folgende XML-Konfiguration:

```
<Filtering metadataHandling="propertyAndText">
```

Szenario 2: Trennen der Metadaten vom Haupttext. In anderen Fällen kann es relevant sein, ob der Text im Hauptdokument oder in einer Metadaten-Property auftritt. Zum Beispiel macht es einen großen Unterschied, ob Sie nach Dokumenten suchen, die von *Doyle* verfasst wurden oder in denen der Begriff *Doyle* im Haupttext auftaucht. In diesem Szenario müssen Sie nicht nur die Indizierung von Properties als Volltext ermöglichen, sondern auch geeignete Präfixe für jede Property hinzufügen, durch die zwischen Text im Inhalt des Hauptdokuments und Text in den Metadaten-Properties unterschieden werden kann.

Der Wert der vordefinierten Property *System.Author* wird dem Präfix *TET_System_Author_* vorangestellt. Sie können zum Beispiel die Suche nach der Property *System.Author=Doyle* mit der Volltextsuche nach *TET_System_Author_Doyle* emulieren. Da *System.Author* eine vordefinierte Property ist, erfordert die zugehörige XML-Konfiguration keine Einträge für Properties, sondern muss lediglich das Indizieren von Properties als Textpräfix aktivieren:

```
<Filtering metadataHandling="propertyAndPrefixedText">
```

Um die Suche nach Properties für Dokumente mit der Artikelnummer *XY123456* mit der Volltextsuche nach *ArticleNumber_XY123456* zu emulieren, verwenden Sie folgende XML-Konfiguration:

```
<Filtering metadataHandling="propertyAndPrefixedText">

<PropertySet guid="404e8a40-2e85-11dd-97f6-0002a5d5c51b">
        <Property identifier="2" textIndexPrefix="ArticleNumber_">
                <Source pdfObject="/Info/ArticleNumber"/>
        </Property>
<PropertySet>
```

3.7 Ignorieren von Seiteninhalten zugunsten von Metadaten

In einigen Fällen möchten Benutzer eventuell ausschließlich nach Metadaten-Properties und nicht nach Seiteninhalten suchen, d.h. Seiteninhalte bei der Indizierung vollständig ignorieren und die Suche ausschließlich auf Metadaten-Properties basieren. Dies kann in folgenden Fällen sinnvoll sein:

- Bessere Steuerung von Benutzerabfragen: um ein exaktes, auf Metadaten-Properties basierendes Ergebnis zu erhalten, kann auf das Sichten langer Ergebnislisten verzichtet werden.
- Die gesuchten Dokumente enthalten in etwa die gleichen Wörter, jedoch in verschiedenen Kombinationen, z.B. Rechnungen oder andere Transaktionsdokumente.
- Die tatsächlichen Seiteninhalte sind für die Suche nicht wirklich relevant, da sie im Voraus bekannt sind. Allerdings ist die Art der montierten Seiten für ein bestimmtes Dokument von Interesse, z.B. Versicherungsgeschäfte, die mit einer variablen Anzahl von Verträgen verbunden sind: nicht der genaue Vertragstext ist für die Suche relevant, sondern Anzahl und Art der zusammengestellten Vertragsunterlagen.
- Die gesuchten Dokumente enthalten überhaupt keinen durchsuchbaren Text, z.B. ohne OCR gescannte Dokumente.
- Die Dokumente enthalten Informationen, die für den Index nicht relevant sind, z.B. lange Finanzdokumente, die nur aus Zahlen bestehen oder technische Pläne ohne Text (bzw. Text nur in den Bildunterschriften der Zeichnungen).
- Leistungsoptimierung in einem der oben genannten Fälle: keine Zeit auf die Indizierung von Dokumenten zu verschwenden, wenn die Inhalte wenig hilfreich für die Suche sind.

XML-Konfiguration für die Deaktivierung der Indizierung von Seiteninhalten. Um die Indizierung von Seiteninhalten komplett zu unterbinden, setzen Sie das Attribut *indexPageContents* des Elements *Filtering* auf *false:*

```
<Filtering indexPageContents="false" metadataHandling="property">
```

4 Metadaten-Verarbeitung in IFilter-Clients

4.1 Metadaten in Windows Search

Erzeugen einer Property-Beschreibungsdatei. Windows Search greift auf das Property-System von Windows zu, das Beschreibungen von vordefinierten und benutzerdefinierten Metadaten-Properties enthält. Um benutzerdefinierte Metadaten-Objekte mit Windows Search zu verwenden, müssen Sie eine Property-Beschreibungsdatei (oft als *.propdesc* bezeichnet) erstellen, die die Namen, Datentypen und GUIDs der Objekte sowie andere Eigenschaftsattribute angibt. Die Eigenschaftsbeschreibungen müssen den zugehörigen Property-Beschreibungen in der XML-Konfigurationsdatei entsprechen. Property-Beschreibungen müssen als XML-Dateien gemäß der unter folgender URL angegebenen Syntax erstellt werden:

msdn.microsoft.com/en-us/library/bb773879(VS.85).aspx

Mehrere Beispiele für Property-Beschreibungsdateien werden mit TET PDF IFilter installiert. Das folgende Beispielfragment zeigt einige Property-Beschreibungen:

```
<propertyDescription name="PDFlib.TETPDFIFilter.fontcount"
  formatID="{5eac0060-1ba4-11dd-92c4-0002a5d5c51b}" propID="2">
  <typeInfo type="Int32" isInnate="true" isViewable="true" isQueryable="true"/>
  <labelInfo label="fontcount" sortDescription="LowestHighest"/>
  <searchInfo inInvertedIndex="true" isColumn="true" columnIndexType="OnDisk"/>
</propertyDescription>
<propertyDescription name="PDFlib.TETPDFIFilter.weblink"
  formatID="{5eac0060-1ba4-11dd-92c4-0002a5d5c51b}" propID="6">
  <typeInfo type="String" isInnate="true" multipleValues="true" isViewable="true"
    isQueryable="true"/>
  <labelInfo label="weblink" sortDescription="AToZ"/>
  <searchInfo inInvertedIndex="true" isColumn="true" columnIndexType="OnDisk"/>
</propertyDescription>
```

Beachten Sie, dass das Attribut *multipleValues="true"* bei Properties mit mehreren Werten entscheidend ist (siehe Abschnitt 3.5, »Properties mit mehreren Werten«, Seite 49).

Vordefinierte Properties. Die Property-Beschreibungsdatei *predefined_properties.propdesc* für alle vordefinierten Properties (siehe Anhang A, »Vordefinierte Metadaten-Properties«) wird mit TET PDF IFilter installiert. Obwohl die Property-Definitionen in TET PDF IFilter integriert sind, müssen Sie die gewünschten Sammlungen von Property-Sets in der XML-Konfigurationsdatei aktivieren (siehe Abschnitt 3.3, »Vordefinierte Metadaten-Properties«, Seite 45) und die Property-Beschreibungen registrieren (siehe »Registrieren von Metadaten-Properties in Windows«, Seite 54), bevor sie in Abfragen verwendet werden können.

XML-Konfiguration für Windows Search. Tabelle 4.1 führt Anforderungen und Empfehlungen für die XML-Konfiguration von TET PDF IFilter mit Windows Search auf.

Tabelle 4.1 XML-Konfiguration für Windows Search

Element	Attribut	Anforderungen und Empfehlung
Filtering	**useIdentifier**	*Muss auf* `true` *gesetzt sein, da Windows Search nur GUID+ID zur Identifizierung von Properties unterstützt, nicht jedoch GUID+Name (siehe »XML-Konfiguration für die Verarbeitung von GUID+Name von Properties«, Seite 47).*
Property	**identifier**	*Erforderlich, da Windows Search nur GUID+ID zur Identifizierung von Properties unterstützt, nicht jedoch GUID+Name (siehe »XML-Konfiguration für die Verarbeitung von GUID+Name von Properties«, Seite 47).*
PropertySet-Collection	**shell**	*Sollte auf* `true` *gesetzt sein, um die Windows Search bekannten Shell-Properties zu unterstützen (beachten Sie, dass* `true` *der Standardwert ist).*

Registrieren von Metadaten-Properties in Windows. Um Properties mit Windows Search abzufragen, müssen die zugehörigen Property-Beschreibungen im Property-System von Windows registriert sein. Sie können Property-Beschreibungen mit dem Kommandozeilen-Tool *registerpropdesc.exe* registrieren, das mit TET PDF IFilter installiert wird. Beachten Sie, dass das Tool nur funktioniert, wenn Windows Search auf dem Computer installiert ist. Übergeben Sie den Dateinamen einer Property-Beschreibung an das Tool, um sie beim Property-System von Windows zu registrieren (falls Leerzeichen im Pfad enthalten sind, müssen Sie diesen in doppelte Anführungszeichen einschließen):

```
registerpropdesc "predefined_properties.propdesc"
```

Beachten Sie auch »Ausführen privilegierter Befehle«, Seite 6. Das Property-System von Windows speichert die Namen der Property-Beschreibungsdateien in folgenden Registry-Schlüsseln (und aufsteigenden Nummern in der letzten Schlüsselkomponente):

```
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\PropertySystem\PropertySchema\0000
```

Das Tool *registerpropdesc* gibt eine Fehlermeldung und einen *HRESULT*-Fehlerwert aus, falls die Property-Datei nicht erfolgreich registriert werden konnte (z.B., weil eine Property doppelt auftaucht). Im Fehlerfall können Sie der Ereignisanzeige der Anwendung weitere Informationen entnehmen:

- *Start, Einstellungen, Systemsteuerung, Verwaltung, Ereignisanzeige*
- Klicken Sie im linken Bereich auf *Windows-Protokolle, Anwendung*.
- Gab es ein Problem mit der Registrierung der Property, wird ein Eintrag mit der Quelle *Microsoft-Windows-propsys* angezeigt. Doppelklicken Sie auf die Zeile mit dem entsprechenden Fehler und analysieren Sie die Fehlermeldung (z.B. *Duplizierte Property ausgelassen)*.

Alternativ können Sie den von *registerpropdesc.exe* ausgegebenen *HRESULT*-Wert zur Fehleranalyse heranziehen. Eine Liste aller *HRESULT*-Werte und -Beschreibungen finden Sie unter

msdn.microsoft.com/en-us/library/cc231198.aspx

Das frei verfügbare Kommandozeilen-Tool *prop.exe* bietet eine ähnliche Funktionalität wie *registerpropdesc* sowie zusätzliche Funktionen für den Umgang mit dem Property-System von Windows:

prop.codeplex.com

Anforderungen für das Registrieren von Property-Beschreibungen. Beachten Sie folgende wichtige Anforderungen, wenn Sie Property-Beschreibungen für Windows Search registrieren:

- Damit nach benutzerdefinierten Properties gesucht werden kann, müssen Sie die Property-Beschreibungen registrieren, den Suchdienst beenden und erneut starten und den Neuaufbau des Katalogs mit folgender Option von Windows Search erzwingen:

```
registerpropdesc "predefined_properties.propdesc"
net stop wsearch
net start wsearch
...Katalog neu erstellen (siehe »Starten/Beenden von Windows Search«, Seite 10)...
```

- Registrierte Properties dürfen nicht die gleiche GUID+ID oder den gleichen Namen haben wie Windows-Properties (einschließlich der Shell-Properties in Anhang A, »Vordefinierte Metadaten-Properties«). Eine Liste aller Windows-Properties finden Sie unter

  msdn.microsoft.com/en-us/library/windows/desktop/dd561977(v=vs.85).aspx

- Um zur Aktivierung des Tools *registerpropdesc* in den Registry-Bereich *HKEY_LOCAL_MACHINE* zu schreiben, benötigen Sie die entsprechenden Rechte.
- Da in der Registry nur die Dateinamen (nicht jedoch die Inhalte) der Property-Beschreibungen gespeichert werden, muss die *propdesc*-Datei an der registrierten Stelle verbleiben, damit die Property-Beschreibungen für die Suche verfügbar sind.
- Für die *propdesc*-Datei benötigen alle Benutzer Leserechte.
- Sie können mehrere Property-Dateien gleichzeitig registrieren. Zwei Property-Beschreibungsdateien dürfen jedoch keine Beschreibungen der selben Property enthalten (durch GUID+ID beschrieben).

Mit der Option *-u* lässt sich eine zuvor registrierte Property-Beschreibung wieder entfernen:

```
registerpropdesc -u "c:\property\acme.propdesc"
```

Suche nach Metadaten-Properties. Sobald Sie benutzerdefinierte Metadaten-Properties konfiguriert haben, können Sie nach Properties suchen. Tabelle 4.2 enthält Beispiele für die Abfrage von Properties. Die Namen aller vordefinierten Properties für Windows Search finden Sie in Anhang A, »Vordefinierte Metadaten-Properties«.

Shell-Properties können gemäß der Beispiele in Tabelle 4.2 abgefragt werden. Die Advanced Query Syntax (AQS) für interaktive Suchvorgänge unterstützt benutzerdefinierte Properties ab Windows 7. In Windows XP/Vista müssen Sie Metadaten-Properties als Text indizieren (siehe Abschnitt 3.6, »Indizieren von Metadaten-Properties als Text«, Seite 50), um benutzerdefinierte Metadaten in interaktiven Suchvorgängen einzuschließen, oder SQL-basierte Suchläufe verwenden (siehe unten).

Tabelle 4.2 Beispielsyntax für die Metadaten-Abfrage mit Windows Search 3.0 und höher

Beispiel für Suchbegriff	***Beschreibung***
`author:Doyle`	*Verfasser enthält* `Doyle`
`author:"Conan Doyle"`	*Verfasser enthält die Wörter* `Conan Doyle`
`author:Doy`	*Verfasser beginnt mit* `Doy`

Tabelle 4.2 Beispielsyntax für die Metadaten-Abfrage mit Windows Search 3.0 und höher

Beispiel für Suchbegriff	*Beschreibung*
`size: < 500000`	*Dokument ist kleiner als 500 000 Byte*
`date: <=4/7/12`	*Änderungsdatum liegt auf oder vor dem 7. April 2012 (Datumsformat abhängig von den Systemeinstellungen)*
`System.Document.DateCreated: = 09/23/2002`	*Erstellungsdatum ist der 23. September 2002 (Datumsformat abhängig von den Systemeinstellungen)*
`System.MIMEType: ="application/pdf"`	*führt alle PDF-Dokumente im Index auf*
`System.Document.PageCount: = 144`	*Dokument hat 144 Seiten*
`PDFlib.TETPDFIFilter.bookmark:Index`	*Dokument enthält ein Lesezeichen mit dem Text* `Index`

SQL-Abfragen für Metadaten-Properties. SQL-Abfragen können sowohl nach vordefinierten als auch nach benutzerdefinierten Properties suchen. Einige Beispiele finden Sie unten. Sie gehen davon aus, dass die Indizierung aller vordefinierten Properties in der XML-Konfigurationsdatei aktiviert wurde und dass die Property-Beschreibung *predefined_properties.propdesc* registriert wurde. Wir verwenden ADO *(ActiveX Data Objects)* und PowerShell-Skripte, um SQL-basierte Abfragen abzusetzen. Sie können die SQL-Anweisungen aber ebenfalls in jeder anderen ADO- oder ADO.NET-Umgebung verwenden. Eine Beschreibung der SQL-Syntaxerweiterungen für Windows finden Sie unter

msdn.microsoft.com/en-us/library/bb231256(VS.85).aspx

Nach Properties kann auf zwei Arten gesucht werden:

- Abfragen eines bestimmten Property-Werts: welche Dokumente haben *Doyle* als Verfasser?
- Abfragen des Werts einer bestimmten Property in ein oder mehreren Dateien: wer ist der Verfasser dieses Dokuments?

PowerShell für SQL-Abfragen. Mehrere Beispiele für PowerShell-Abfrageskripte werden mit TET PDF IFilter installiert. Mit dem folgenden PowerShell-Skript werden alle Lesezeichen für alle Dokumente aufgelistet:

```
$objConnection = New-Object -comobject ADODB.Connection
$objRecordset = New-Object -comobject ADODB.Recordset
$objConnection.Open("Provider=Search.CollatorDSO;Extended
Properties='Application=Windows';")

$objRecordSet.Open(
"SELECT System.ItemPathDisplay, `"PDFlib.TETPDFIFilter.bookmark`" FROM SYSTEMINDEX ",
$objConnection)

While ($objRecordset.EOF -ne $True) {
    $private:item = $objRecordset.Fields.Item("System.ItemPathDisplay")
    Write-Output $item.Value
    $item = $objRecordset.Fields.Item("PDFlib.TETPDFIFilter.bookmark")
    Write-Output $item.Value
    $objRecordset.MoveNext()
}
```

Das folgende PowerShell-Skript listet alle Dokumente auf, in denen mindestens ein Lesezeichen mit dem Text *alpha* vorkommt:

```
$objConnection = New-Object -comobject ADODB.Connection
$objRecordset = New-Object -comobject ADODB.Recordset
$objConnection.Open("Provider=Search.CollatorDSO;Extended
Properties='Application=Windows';")

$objRecordSet.Open("SELECT System.ItemPathDisplay FROM SYSTEMINDEX WHERE " +
        "`"PDFlib.TETPDFIFilter.bookmark`" = SOME ARRAY ['alpha']", $objConnection)

While ($objRecordset.EOF -ne $True) {
    $private:item = $objRecordset.Fields.Item("System.ItemPathDisplay")
    Write-Output $item.Value
    $objRecordset.MoveNext()
}
```

VBScript für SQL-Abfragen. Der folgende VBScript-Code listet alle Dokumente zusammen mit den Namen der Anwendung auf, von der das Dokument erstellt wurde. Leider können nur einige Shell-Properties mit VBScript-Abfragen verwendet werden; andere Properties können nicht abgefragt werden:

```
On Error Resume Next

Set objConnection = CreateObject("ADODB.Connection")
Set objRecordSet = CreateObject("ADODB.Recordset")

objConnection.Open "Provider=Search.CollatorDSO;Extended
Properties='Application=Windows';"

objRecordSet.Open "SELECT System.ItemPathDisplay, System.ApplicationName FROM
SYSTEMINDEX", objConnection

objRecordSet.MoveFirst

Do Until objRecordset.EOF
    Wscript.Echo objRecordset.Fields.Item("System.ItemPathDisplay")
    Wscript.Echo objRecordset.Fields.Item("System.ApplicationName")
    Wscript.Echo ""
    objRecordset.MoveNext
Loop
```

Komplexe Property-Abfragen mit SQL. Die folgenden Beispiele enthalten nur die relevante SQL-Anweisung und können in jeder SQL-Umgebung verwendet werden. Um diese Anweisungen in PowerShell-Skripten zu verwenden, müssen Sie die korrekten Anführungszeichen verwenden, z.B. *`"PDFlib.TETPDFIFilter.width`"* statt *"PDFlib.TETPDFIFilter.width"*. Viele Beispiele unten verwenden Array-Abfragen für Vektor-Properties (siehe Abschnitt 3.5, »Properties mit mehreren Werten«, Seite 49). Ausführliche Informationen zu Array-Abfragen finden Sie unter

msdn.microsoft.com/en-us/library/bb231264(VS.85).aspx

- Liste aller Dokumente, in denen *Doyle* in der Property Verfasser vorkommt:

```
SELECT System.ItemPathDisplay FROM SYSTEMINDEX WHERE CONTAINS("System.Author",
'Doyle')
```

- Liste aller Dokumente, in denen der Verfasser mit *Rudy* beginnt:

```
SELECT System.ItemPathDisplay FROM SYSTEMINDEX WHERE CONTAINS("System.Author",
'"Rudy*"')
```

- Liste aller PDF/A-1a-konformen Dokumente:

```
SELECT System.ItemPathDisplay FROM SYSTEMINDEX WHERE "PDFlib.TETPDFIFilter.pdfa" =
'PDF/A-1:2005'
```

- Liste aller Dokumente, in denen mindestens ein Lesezeichen mit *alph* beginnt:

```
SELECT System.ItemPathDisplay FROM SYSTEMINDEX WHERE "PDFlib.TETPDFIFilter.bookmark"
LIKE SOME ARRAY ['alph%']
```

- Liste aller Dokumente, die mindestens einen der Fonts *Bembo* und *TimesNewRoman* enthalten:

```
SELECT System.ItemPathDisplay FROM SYSTEMINDEX WHERE "PDFlib.TETPDFIFilter.font" =
SOME ARRAY ['Bembo', 'TimesNewRoman']
```

- List aller Dokumente, die sowohl den Font *Bembo* als auch den Font *Bembo-Bold* enthalten:

```
SELECT System.ItemPathDisplay FROM SYSTEMINDEX WHERE "PDFlib.TETPDFIFilter.font" =
SOME ARRAY ['Bembo'] AND "PDFlib.TETPDFIFilter.font" = SOME ARRAY ['Bembo-Bold']
```

- Liste aller Dokumente mit mindestens einer Seite von *width=595*. Beachten Sie die einfachen Anführungszeichen um 595, da *width* vom Typ *Double* ist; die Anführungszeichen sind beim Typ *Int32* nicht erforderlich:

```
SELECT System.ItemPathDisplay FROM SYSTEMINDEX WHERE "PDFlib.TETPDFIFilter.width" =
SOME ARRAY ['595']
```

- Liste aller Dokumente mit mindestens einer Seite von *width=200* und mindestens einer Seite von *height=150*:

```
SELECT System.ItemPathDisplay FROM SYSTEMINDEX WHERE "PDFlib.TETPDFIFilter.width" =
SOME ARRAY ['200'] AND "PDFlib.TETPDFIFilter.height" = SOME ARRAY ['150']
```

- Liste aller Dokumente mit mindestens einem Tennis-Bild (Photoshop-Kategorie *TEN=tennis*):

```
SELECT System.ItemPathDisplay FROM SYSTEMINDEX WHERE
"PDFlib.TETPDFIFilter.images.photoshop.SupplementalCategories" = SOME ARRAY ['TEN']
```

- Liste aller Dokumente mit einer PDF-Version höher als 1.6 (die PDF-Version wird von TET PDF IFilter als String zurückgegeben, der die Versionsnummer mal zehn enthält, z.B. 16 für PDF 1.6):

```
SELECT System.ItemPathDisplay FROM SYSTEMINDEX WHERE "PDFlib.TETPDFIFilter.pdfversion"
> '16'
```

4.2 Metadaten in SharePoint und Search Server

Die Verarbeitung von Metadaten-Eigenschaften bei Search Server funktioniert fast genauso wie bei SharePoint. In diesem Abschnitt finden Sie die Konfigurationsschritte für beide Produkte. Produktspezifische Unterschiede sind jeweils angegeben.

Sie können die Verarbeitung von Metadaten in SharePoint mit Hilfe des Verwaltungsnamensraums *Microsoft.Office.Server.Search.Administration* von Enterprise Search per Programm steuern. Zur Verarbeitung von Metadaten-Eigenschaften werden in SharePoint folgende Konzepte eingesetzt:

- *Durchforstete (gecrawlte) Eigenschaften* werden von TET PDF IFilter beim Indizieren (Crawlen) von PDF-Dokumenten erzeugt. Gecrawlte Eigenschaften können mehrere Werte haben. Sie werden über die Konfigurationsdatei von TET PDF IFilter gesteuert. SharePoint durchsucht gecrawlte Eigenschaften genauso wie andere Felder, dabei sind jedoch keine erweiterten Funktionen verfügbar.
- *Verwaltete Eigenschaften* können in Suchabfragen und in der erweiterten Suche verwendet und in den Suchergebnissen angezeigt werden.

Ausführliche Informationen zu verwalteten (managed) Eigenschaften in der Erweiterten Suche finden Sie unter

msdn.microsoft.com/en-us/library/bb428648.aspx
msdn.microsoft.com/en-us/library/bb608302.aspx

XML-Konfiguration für TET PDF IFilter mit SharePoint. Tabelle 4.3 führt Anforderungen und Empfehlungen für die XML-Konfiguration von TET PDF IFilter mit SharePoint auf.

Tabelle 4.3 XML-Konfiguration für SharePoint

Element	Attribut	Anforderungen und Empfehlung
Filtering	***useIdentifier***	*Auf* `false` *setzen, um Eigenschaften einschließlich der Vordefinierten über GUID+Name anzusprechen.*
Property	***emitAsVector***	*Muss für Eigenschaften mit mehreren Werten auf* `true` *gesetzt werden (siehe Abschnitt 3.5, »Properties mit mehreren Werten«, Seite 49).*
Property	***friendlyName***	*Dieses Attribut wird empfohlen, da die Verwendung von GUID+Name das Auffinden von Eigenschaften in der Liste erleichtert (siehe »Identifikation von Properties und GUIDs«, Seite 44).*

Konfigurieren von benutzerdefinierten Metadaten-Eigenschaften. So bereiten Sie benutzerdefinierte Properties für die Indizierung vor:

- Führen Sie eine komplette Durchforstung (Crawl) durch. Während der Durchforstung gibt TET PDF IFilter die Eigenschaften aus, die in der XML-Konfigurationsdatei konfiguriert sind und in ein oder mehreren Dokumenten gefunden wurden. Dies ist erforderlich, damit SharePoint die neu gefundenen Eigenschaften aufnehmen kann.
- SharePoint erzeugt für alle neuen Eigenschaftenkategorien synthetische Namen (z.B. *»Category 1«, »Category 2«).* Ein mit TET PDF IFilter installiertes PowerShell-Skript erzeugt eine Liste von Kategorienamen mit den zugehörigen GUIDs. Führen Sie das Skript folgendermaßen aus:

```
list_categories.ps1 <site URL>
```

In der resultierenden Ausgabe können Sie Kategorien über ihre GUID identifizieren; vergleichen Sie die GUID mit denen in *PropertySet/@guid* in der XML-Konfigurationsdatei für TET PDF IFilter (wenn Sie benutzerdefinierte Properties erstellt haben) oder mit den GUIDs der Property-Sets in Anhang A, »Vordefinierte Metadaten-Properties« (wenn Sie Properties aus den Sammlungen an Standard-Property-Sets verwenden).

- Ersetzen Sie die von SharePoint verwendeten synthetischen Kategorienamen durch benutzerfreundliche Namen, nachdem Sie die Kategorien anhand ihrer vom Skript *list_categories.ps1* zurückgegebenen GUIDs identifiziert haben:
  SharePoint: Klicken Sie auf *Start, SharePoint 3.0 Zentralverwaltung*. Klicken Sie in der linken Spalte auf *Verwaltung der gemeinsamen Dienste: SharedServices1* (oder ähnlich) und dort auf *Suchverwaltung, Eigenschaftenzuordnungen für Metadaten*. Klicken Sie wieder in der linken Spalte auf *Gecrawlte Eigenschaften*. Wählen Sie in der *Ansicht für gecrawlte Eigenschaften* einen Kategorienamen aus der Dropdown-Liste und klicken Sie auf *Kategorie bearbeiten*. Ändern Sie im Dialog *Kategorie bearbeiten* den Namen der Kategorie.
- SharePoint: Klicken Sie auf *Start, SharePoint 3.0 Zentralverwaltung*. Klicken Sie in der linken Spalte auf *Verwaltung der gemeinsamen Dienste: SharedServices* (oder ähnlich) und dort auf *Suchverwaltung, Verwaltete Eigenschaften*.
  Search Server: Klicken Sie auf *Start, Alle Programme, Microsoft Search Server, Search Server 2008 Verwaltung*. Die Seite *Eigenschaftenzuordnungen für Metadaten* wird geöffnet.
- Klicken Sie auf *Neue verwaltete Eigenschaft* (siehe Abbildung 4.1).
- Geben Sie für die Eigenschaft Namen, Beschreibung und Datentyp an und klicken Sie auf *Zuordnung hinzufügen* unten rechts neben dem Feld *Dieser verwalteten Eigenschaft zugeordnete, gecrawlte Eigenschaften*.

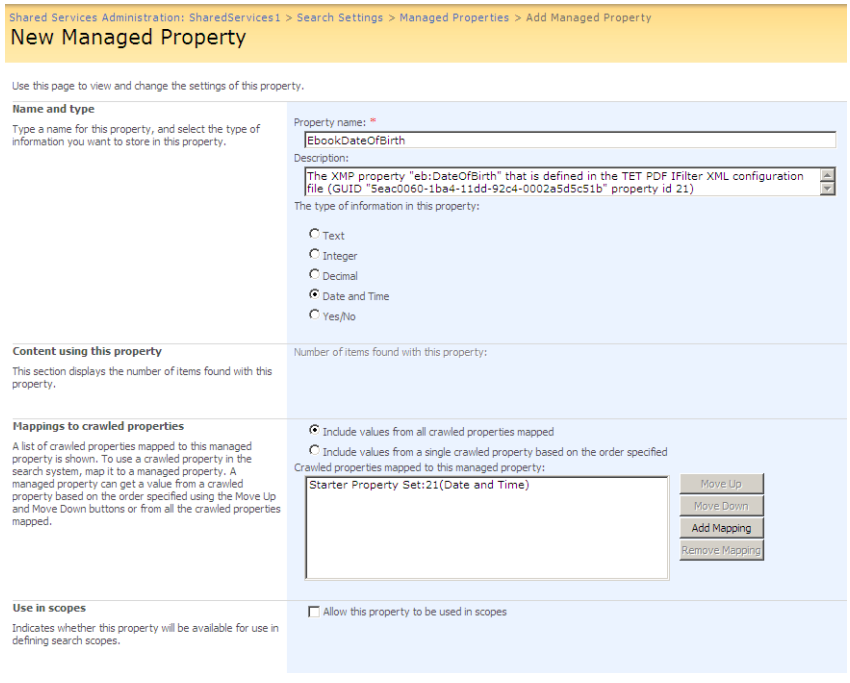


Abb. 4.1 Hinzufügen von verwalteten Eigenschaften in SharePoint

- Die Dropdown-Liste *Kategorie auswählen* filtert die Liste der gecrawlten Eigenschaften, die in der Listenbox *Gecrawlte Eigenschaft auswählen* angezeigt werden. Sie zeigt eine Liste von Eigenschaften zusammen mit ihren Kategorienamen und Eigenschaften-Identifier oder benutzerfreundlichen Namen. Die Anzeige des Identifiers oder benutzerfreundlichen Namens hängt von der Property-Definition in der XML-Konfigurationsdatei von TET PDF IFilter ab, siehe Anhang , »Konfigurieren von benutzerdefinierten Metadaten-Eigenschaften«. Wählen Sie eine neue gecrawlte Eigenschaft, vergeben Sie einen Namen für die verwaltete Eigenschaft und speichern sie.

Beispielausgabe des PowerShell-Skripts. Die folgende Auflistung zeigt den Ablauf des PowerShell-Skripts für die Seite *litwaredemo*:

```
PS C:\> & "C:\Program Files\PDFlib\TET PDF IFilter 5.0 64-bit\IFilter clients\Sharepoint\
   list_property_categories.ps1" http://litwaredemo

"Category 1" 007867f0-c59b-43fc-ab1e-8eee77057254
"Category 2" 1e3ee840-bc2b-476c-8237-2acd1a839b22
"Category 5" c60e822a-074f-4bd5-9889-6ebd372f2000
"Category 6" 17eb8447-fc9b-4d4d-81df-31e9aa770cbf
"Category 7" 5eac0060-1ba4-11dd-92c4-0002a5d5c51b
"Dublin Core" d92bb3ca-ce2b-4b9b-972a-5bf54b468171
...
```

Die erzeugte Liste zeigt, dass *»Category 7«* zu dem in *starter_advanced_search.xml* definierten Property-Set gehört. Benennen Sie es um in *»Starter Property Set«*, wie oben beschrieben.

Vorbereiten von SharePoint XML für Suchabfragen und verwaltete Eigenschaften. In diesem Schritt bereiten Sie das erforderliche XML vor, um verwaltete Eigenschaften zur Erweiterten Suche hinzuzufügen. Im folgenden Abschnitt wird anhand von Beispielen beschrieben, wie Sie dieses XML anwenden:

Erzeugen Sie für jede Eigenschaft ein Element *PropertyDef*, das dem Element *PropertyDefs* untergeordnet ist:

```
<PropertyDef Name="EbookDateOfBirth" DataType="datetime" DisplayName="Ebook Date of
Birth"/>
```

Das Attribut *Name* muss dem Namen der Eigenschaft entsprechen, das Attribut *DataType* beschreibt den Typ gemäß Tabelle 4.4 und *DisplayName* enthält einen beliebigen Namen, der auf der Benutzeroberfläche angezeigt wird.

Um die neue verwaltete Property für Suchabfragen von PDF-Dokumenten verfügbar zu machen, fügen Sie diese dem Element *ResultType* hinzu:

Tabelle 4.4 Eigenschaftsdatentypen für SharePoint

Datentyp in TET PDF IFilter	Datentypen für SharePoint
Int32	*integer*
Double	*decimal*
Boolean	*boolean*
DateTime	*datetime*
String	*text*

```
<ResultType DisplayName="PDF Documents" Name="pdfdocuments">
        <Query>FileExtension='pdf'</Query>
        <PropertyRef Name="Author"/>
        <PropertyRef Name="Description"/>
        <PropertyRef Name="FileName"/>
        <PropertyRef Name="Size"/>
        <PropertyRef Name="Path"/>
        <PropertyRef Name="Created"/>
        <PropertyRef Name="Write"/>
        <PropertyRef Name="CreatedBy"/>
        <PropertyRef Name="ModifiedBy"/>
        <PropertyRef Name="EbookDateOfBirth"/>
</ResultType>
```

Eine vollständige XML-Datei *(starter_advanced_search.xml)* für alle Properties im Starter-Set wird mit TET PDF IFilter installiert. Sie wurde mit dem Virtual PC Image von Microsoft zum Testen von SharePoint erstellt. Sie können das XML zu Test- und Evaluierungszwecken verwenden. Beachten Sie jedoch, dass XML zu Produktionszwecken auf der Grundlage des bestehenden XML und den neu konfigurierten Eigenschaften sorgfältig konstruiert werden muss.

Suche nach benutzerdefinierten Metadaten-Eigenschaften. Um erweiterte Metadatensuche zu implementieren, müssen Sie in TET PDF IFilter das Indizieren von Eigenschaften konfigurieren (siehe Abschnitt 3.4, »Benutzerdefinierte Metadaten-Properties«, Seite 46). TET PDF IFilter erzeugt dann eine gecrawlte Eigenschaft. Dann müssen Sie eine neue verwaltete Eigenschaft auf der Seite *Erweiterte Suche* wie folgt verfügbar machen:

- Melden Sie sich bei der SharePoint-Website an und navigieren Sie zu *Erweiterte Suche*. Um eine neue verwaltete Eigenschaft zu erstellen, klicken Sie in diesem Suchformular am unteren Rand der Seite auf *Eigenschaftseinschränkungen hinzufügen...*
- Klicken Sie rechts oben auf der Seite auf *Websiteaktionen*. Wählen Sie *Seite bearbeiten* aus dem Dropdown-Menü. Dadurch ändert sich das Aussehen der Seite und Sie erhalten zusätzliche Steuerelemente zur Modifizierung der Seite.
- Klicken Sie oben rechts unter *Erweitertes Suchfeld* auf *Bearbeiten*. Wählen Sie aus dem sich öffnenden Dropdown-Menü *Shared Web Part ändern*. Dadurch wird das Feld *Erweiterte Suche* mit einer gestrichelten Linie umrandet und auf der rechten Seite ein neues Feld mit dem gleichen Titel *Erweiterte Suche* angezeigt. Es enthält Einträge wie z.B. *Suchfeld*, *Scopes*, *Eigenschaften*.
- Klappen Sie die Kategorie *Eigenschaften* auf und suchen Sie nach dem Feld *Eigenschaften* mit dem XML-Inhalt. Klicken Sie in dieses Feld und dann auf die Schaltfläche mit drei Punkten und dem Tooltip *Klicken, um den Builder zu verwenden*, um das Texteingabefeld mit XML zu öffnen. Editieren Sie das XML wie im Beispiel unten angegeben; Sie können dazu auch einen XML-Editor verwenden.
- Klicken Sie nach dem Schließen des Texteingabefelds auf *OK* unten im Feld, in dem Sie editiert haben und klicken Sie auf *Einchecken um den Entwurf zu teilen* oben auf der Seite. Das Formular *Erweiterte Suche* wird wieder in normaler Ansicht angezeigt und wenn Sie im Dropdown-Menü *Ergebnistyp Alle Ergebnisse* auswählen, erscheint der neue Eigenschaftsname im Menü *(Eigenschaft auswählen)* unten auf der Seite unter *Eigenschaftseinschränkungen hinzufügen...*. Um das geänderte Formular für alle Benutzer verfügbar zu machen, müssen Sie eventuell zusätzlich zu *Einchecken, um den Entwurf zu teilen* oben im Formular *Erweiterte Suche* auch auf *Veröffentlichen* klicken.

Jetzt können Sie nach Dokumenten mit bestimmten Einträgen in der verwalteten Eigenschaft suchen. Abbildung 4.2 zeigt die *Erweiterte Suche* mit der benutzerdefinierten Eigenschaft *Ebook Date of Birth* im unteren Bereich der Seite.

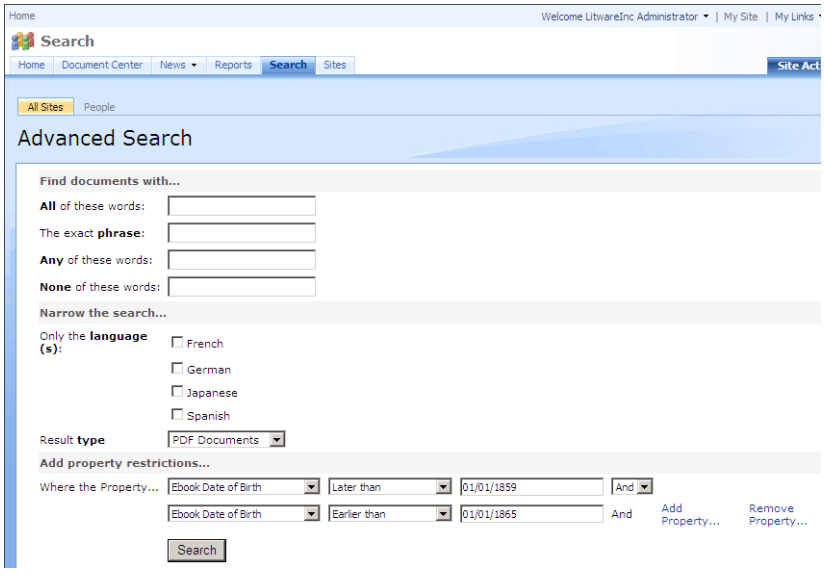


Abb. 4.2 Erweiterte Suche in SharePoint mit benutzerdefinierter Eigenschaft

Hinweis *Wenn Sie vermuten, dass einzelne Dokumente in der Liste der Suchergebnisse fehlen, hat SharePoint sie eventuell als Duplikat entfernt. Um das Entfernen von Duplikaten zu deaktivieren, klicken Sie auf den Link* `Duplikate anzeigen` *auf der Seite* `Suchergebnisse`.

Suche nach Metadaten-Eigenschaften. Tabelle 4.5 enthält Beispiele für die Abfrage von Eigenschaften. Eigenschaftsabfragen werden gemäß folgendem einfachen Schema aufgebaut:

```
<property name>:<value>
```

Die Syntaxbeschreibung für die Abfrage von Eigenschaften finden Sie unter *msdn.microsoft.com/en-us/library/office/ff394509%28v=office.14%29.aspx*

Tabelle 4.5 Beispiele für Metadaten-Abfrage mit SharePoint

Beispiel für Suchbegriff	*Beschreibung*
`author:Doyle`	*Verfasser enthält* `Doyle`
`author:Arthur author:Doyle`	*Verfasser enthält* `Arthur` *und* `Doyle`
`author:"Arthur Conan Doyle"`	*Verfasser enthält genau den Text* `Arthur Conan Doyle`

4.3 Metadaten in SQL Server

SQL Server unterstützt keine Indizierung von und Suche nach Metadaten-Properties. Alle Properties werden deshalb ignoriert. Um Metadaten-Abfragen zu implementieren, empfehlen wir, Metadaten-Properties als Text zu indizieren (siehe Abschnitt 3.6, »Indizieren von Metadaten-Properties als Text«, Seite 50).

XML-Konfiguration für SQL Server. Tabelle 4.6 führt Anforderungen und Empfehlungen für die XML-Konfiguration von TET PDF IFilter mit SQL Server auf.

Tabelle 4.6 XML-Konfiguration für SQL Server

Element	*Attribut*	*Anforderungen und Empfehlung*
Filtering	***metadataHandling***	*Setzen Sie dieses Attribut auf* `propertyAndText` *oder* `propertyAndPrefixedText`*, um das Indizieren von Properties als Text zu aktivieren (siehe Abschnitt 3.6, »Indizieren von Metadaten-Properties als Text«, Seite 50).*
Property	***textIndexPrefix***	*Setzen Sie das Präfix, wenn Sie explizit nach Properties suchen möchten.*

Suche nach Metadaten-Properties. Da es keine dedizierte Unterstützung für die Suche nach Properties bei SQL Server gibt, müssen Sie die Metadaten-Properties in einer Volltextabfrage abfragen. Sobald Sie die Indizierung von Properties als Text aktiviert haben, können Sie nach Dokumenten des Verfassers *Arthur Conan Doyle* wie folgt suchen:

```
SELECT name FROM DocumentTable WHERE CONTAINS(*,'"TET_System_Author_Arthur Conan Doyle"')
GO
```

Mit der folgenden Anweisung fragen Sie Dokumente ab, bei denen der Verfasser mit *Arthur* beginnt:

```
SELECT name FROM DocumentTable WHERE CONTAINS(*,'"System_Author_Arthur*"')
GO
```

5 Fehlerbehebung

5.1 TET PDF IFilter funktioniert nicht

Wenn TET PDF IFilter nicht funktioniert, prüfen Sie die folgenden Punkte.

Ist TET PDF IFilter korrekt registriert? Mit dem Kommandozeilen-Tool *FiltReg.exe* können Sie prüfen, ob TET PDF IFilter korrekt registriert ist. Das Programm führt alle Dateinamenserweiterungen auf, die mit IFiltern verbunden sind, sowie den Namen der zugehörigen IFilter-DLL. *FiltReg.exe* wird mit Microsoft Visual Studio installiert und ist auch im Windows SDK für Windows 7 enthalten. Für weitere Information siehe

msdn.microsoft.com/en-us/library/ms692537(VS.85).aspx

Hinweis *Für das Testen der 64-Bit-Variante von TET PDF IFilter ist die 64-Bit-Variante von* `FiltReg.exe` *erforderlich. Die 64-Bit-Variante dieses Tools wird nur installiert, wenn Sie Windows SDK auf einem 64-Bit-Computer installieren.*

Ist TET PDF IFilter korrekt registriert, gibt *FiltReg.exe* in etwa Folgendes aus:

```
Filters loaded by extension:
...
.pdf --> PDFlib TET PDF IFilter 32-bit (C:\Programme\PDFlib\TET PDF IFilter 5.0 64-
bit\bin\TETPDFIFilter.dll)
```

Ist TET PDF IFilter nicht korrekt registriert, müssen Sie es manuell registrieren (siehe »Manuelle Installation«, Seite 6).

Wenn Sie Windows Search installiert haben, können Sie die korrekte IFilter-Registrierung folgendermaßen prüfen: Klicken Sie auf *Start*, *Systemverwaltung*, *Indizierungsoptionen*, *Erweitert*, *Dateitypen*. Es wird eine lange Liste von Dateitypen mit den zugehörigen Filtern angezeigt. Blättern Sie in dieser Liste in der Spalte *Erweiterung* zu *pdf*. Der entsprechende Eintrag in der Spalte *Filterbeschreibung* sollte *PDFlib TET PDF IFilter 32-bit* (oder ggf. 64-bit) lauten.

TET PDF IFilter und Adobe Acrobat. Wenn Sie Adobe Reader oder Adobe Acrobat nach TET PDF IFilter installieren (oder den Reparaturmodus von Acrobat verwendet haben), werden dadurch alle Registry-Einträge von TET PDF IFilter überschrieben. Sie sollten dann das Installationsprogramm von TET PDF IFilter im Reparaturmodus erneut ausführen oder die DLL von TET PDF IFilter manuell registrieren, siehe »Manuelle Installation«, Seite 6.

Ist der Lizenzschlüssel verfügbar? Während TET PDF IFilter auf Windows XP/Vista/7/8/10 ohne kommerziellen Lizenzschlüssel verwendet werden kann, ist für Windows Server ein Lizenzschlüssel erforderlich. Wenn Sie auf einem Server-System arbeiten und die PDF-Indizierung nicht zu funktionieren scheint, kann das am fehlenden Lizenzschlüssel für TET PDF IFilter liegen. In diesem Fall läuft TET PDF IFilter im Evaluierungsmodus und ist daher auf kleine Dokumente beschränkt.

Dies können Sie in der Windows-Ereignisanzeige überprüfen (siehe »Ereignisprotokollierung der Anwendung«, Seite 70). Bei Problemen mit dem Lizenzschlüssel wird ein Eintrag mit der Quelle *TET PDF IFilter* und der Kategorie *TET Error* erzeugt. Doppelklicken

Sie auf die Zeile mit dem entsprechenden Fehler und analysieren Sie die Fehlermeldung. Der folgende Text bedeutet, dass kein gültiger Lizenzschlüssel gefunden werden konnte:

```
TET API Error in TetIFilter::Init: open_document_mem:
Invalid license key (error number 1986)
```

Wenn Sie diese Fehlermeldung erhalten, müssen Sie den Lizenzschlüssel in der Registry eintragen (siehe »Manuelle Installation«, Seite 6).

5.2 Probleme beim Einsatz von TET PDF IFilter

Wenn TET PDF IFilter nicht wie erwartet läuft, können die unten genannten Analysemethoden hilfreich sein.

Identifizieren problematischer Dokumente. Je nach IFilter-Client können die protokollierten Einträge die Namen der betroffenen Dateien enthalten oder auch nicht. Falls Dateinamen im Ereignisprotokoll sichtbar sind, sind sie unter Umständen dennoch nicht hilfreich. Windows Search gibt zum Beispiel gar keine Dateinamen aus. SharePoint wiederum lädt Dokumente über HTTP herunter und erzeugt eine temporäre lokale Kopie. Die Ereignisanzeige enthält die temporären lokalen Dateinamen, die aber mit den originalen URLs nichts zu tun haben. Um die betroffenen PDF-Dokumente leichter identifizieren zu können, enthalten die Einträge die Größe der Dateien in Bytes. Sie können die betroffenen Dokumente mit der Suchmaschine selbst schnell identifizieren.

- In Windows Search können Sie folgende Abfrage verwenden (im Beispiel beträgt die Größe der Datei 12345 Bytes):

  ```
  size: = 12345
  ```

- In SharePoint können Sie gescheiterte Filterversuche für eine Datei im Crawlprotokoll von SharePoint identifizieren (Gemeinsame Dienste von Office SharePoint verwalten: *Gemeinsame Dienste von Office SharePoint, Sucheinstellungen, Crawlprotokolle)*. Die hier aufgeführten Fehler bei PDF-Dokumenten entsprechen den von TET PDF IFilter ausgegebenen Fehlern in der Windows-Ereignisanzeige. Durch Vergleich der Dateigrößen im Crawlprotokoll und in der Ereignisanzeige können Sie die problematischen Dokumente identifizieren.

Beachten Sie auch das Konfigurationsattribut *Filtering/@errorIndicator*, das für problematische Dokumente im Index einen identifizierenden String ausgeben kann (siehe Abschnitt 6.2, »XML-Elemente und -Attribute«, Seite 75).

Gesperrte PDF-Dokumente können nicht indiziert werden. Wenn ein Dokument durch eine Anwendung gesperrt ist, kann TET PDF IFilter es nicht indizieren. Vor allem in Acrobat sind Dateien gesperrt, solange sie geöffnet sind. Obwohl der IFilter-Client das gesperrte Dokument später erneut versucht zu indizieren, ist der Index so lange unvollständig, bis das gesperrte Dokument freigegeben wird. Deshalb sollten Sie keine PDF-Dokumente während der Indizierung öffnen.

5.3 Keine oder unvollständige Indizierung von PDF-Dokumenten

Für SharePoint Server und Search Server bestehen Einschränkungen bei der Indizierung großer Dokumente. Da diese Einschränkungen in der Microsoft-Dokumentation nicht hinreichend erklärt sind, fassen die folgenden Abschnitte die Informationen aus verschieden Support-Artikeln und Blogs von Microsoft zusammen. Diese Angaben sind nur als Hinweise zu verstehen; wenden Sie sich im Zweifelsfall bitte an Microsoft.

5.3.1 Einschränkungen bei SharePoint 2010 und SharePoint 2013

Für SharePoint gibt es einige Einschränkungen beim Indizieren von Dokumenten. Weitere Information zu festen und konfigurierbaren Einschränkungen bei SharePoint 2013 finden Sie unter

https://technet.microsoft.com/de-de/library/cc262787%28v=office.15%29.aspx

Die folgenden SharePoint-Grenzwerte gelten für TET PDF IFilter:

- Die maximale Dateigröße *(MaxDownloadSize)* legt die maximale Größe von Dokumenten fest, die durchsucht und indiziert werden können. Der Standardwert für SharePoint 2013 ist 64 MB.
- Der maximale Zuwachsfaktor *(MaxGrowFactor)* legt den Faktor fest, mit dem der Wert *MaxDownloadSize* multipliziert wird, um das maximale Textvolumen für ein indiziertes Dokument zu ermitteln. Dieser Faktor ist erforderlich, weil der Text in der Datei komprimiert sein kann, was bei PDF-Dokumenten meist der Fall ist (Einheit: keine, Standardwert: 4).
- Die Größe des analysierten Inhalts gibt an, wie viele Zeichen eines Dokuments indiziert werden können. SharePoint 2013 hat eine Beschränkung von 2 Millionen Zeichen, die nicht geändert werden kann.

Ändern von maximaler Dateigröße und Zuwachsfaktor bei SharePoint 2010 und 2013.
Geben Sie folgendes Kommando in der *SharePoint 2013 Management Shell* ein (bei Verwendung mehrerer Suchdienste fügen Sie *-id <GUID of SSA>* an das erste Kommando an):

```
$ssa = Get-SPEnterpriseSearchServiceApplication
$ssa.SetProperty("MaxDownloadSize", ...new value...)
```

Eine ähnliche Sequenz kann für *MaxGrowFactor* angewendet werden. Die aktuellen Werte können Sie folgendermaßen prüfen:

```
$ssa = Get-SPEnterpriseSearchServiceApplication
$ssa.GetProperty("MaxDownloadSize")
```

5.3.2 Einschränkungen bei früheren Versionen von SharePoint

Der Standardwert von *MaxDownloadSize* ist 16 MB, der Standardwert von *MaxGrowFactor* ist 4 (siehe Abschnitt 5.3.1, »Einschränkungen bei SharePoint 2010 und SharePoint 2013«, Seite 68, für eine Erklärung dieser Werte), was maximal 64 MB an extrahiertem Text pro Dokument bedeutet. Je nach Produkt und Version sind die Registry-Einträge *MaxDownloadSize* und *MaxGrowFactor* in folgenden Schlüsseln in der Windows-Registry zu finden:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Office Server\<version>
        \Search\Applications\<GUID>\Gathering Manager

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Shared Tools\Web Server Extensions\<version>
        \Search\Applications\<GUID>\Gathering Manager

HKEY_LOCAL_MACHINE\Software\Microsoft\SPSSearch\Gathering Manager
```

Die *GUID* variiert je nach Installation.

Speichergröße von Abschnitten. Ein weiterer Grenzwert betrifft die Gesamtzahl der eindeutigen Wörter pro Dokument, die indiziert werden können. Der Wert *CB_ChunkBufferSizeInMegaBytes* legt den Platz fest, der für die Sammlung eindeutiger Wörter pro Dokument reserviert wird (Einheit: MB, Standardwert: 8).

Bytes pro Dokument. Der Wert *CB_MinBytesReservedForDoc* hängt vom Wert *CB_ChunkBufferSizeInMegaBytes* ab. Er sollte 2 MB kleiner sein als der Wert von *CB_ChunkBufferSizeInMegaBytes*, obwohl dieses die voreingestellten Werte diese Regel nicht einhalten (Einheit: Bytes, Standardwert: 3,145,728).

Je nach Produkt und Version sind die Registry-Einträge *CB_ChunkBufferSizeInMegaBytes* und *CB_MinBytesReservedForDoc* in folgenden Schlüsseln in der Windows-Registry zu finden:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Office Server\<version>
        \Search\Global\Gathering Manager

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Shared Tools\Web Server Extensions\<version>
        \Search\Global\Gathering Manager
```

Einen Support-Artikel von Microsoft, der diese Werte für Microsoft Office SharePoint Server 2007 beschreibt, finden Sie hier:

support.microsoft.com/kb/970776/EN-US

5.3.3 Speicherbeschränkungen für Search Server

Für den Crawl-Prozess in Search Server 2008 gelten bestimmte Speichergrenzen, die das erfolgreiche Crawlen bestimmter Dokumente unter Umständen verhindern. Diese Grenzwerte könne mit Hilfe von Registry-Schlüsseln gesteuert werden:

- Im Registry-Schlüssel *DedicatedFilterProcessMemoryQuota* wird der Grenzwert für den Speicherplatz festgelegt.
- Wenn ein IFilter mehr Speicher benötigt, als im Registry-Schlüssel *FilterProcessMemoryQuota* angegeben, wird der Crawl-Prozess abgebrochen. Microsoft empfiehlt, den Standardwert zu erhöhen, falls eine 64-Bit-Variante von Search Server verwendet wird und der Index-Server mehr als 4 GB physikalischen Speicher hat.

Die oben aufgeführten Registry-Einträge sind in folgendem Schlüssel in der Windows-Registry zu finden:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Office Server\<version>
        \Search\Global\Gathering Manager
```

Für weitere Informationen siehe den folgenden Artikel:
technet.microsoft.com/en-us/library/dd630760.aspx

5.4 Fehleranalyse

Entsprechen die Suchergebnisse nicht Ihren Erwartungen und vermuten Sie Probleme bei der Extraktion der Textinhalte aus den indizierten Dokumenten, können die unten aufgeführten Tools zur Fehleranalyse hilfreich sein.

Ereignisprotokollierung der Anwendung. TET PDF IFilter erzeugt für verschiedene Ereignisse Einträge in der Windows-Ereignisanzeige. Sie können die Ereignisanzeige folgendermaßen öffnen:

- Windows Vista/7: Klicken Sie auf *Start*, geben Sie im Suchfeld *Ereignisanzeige* ein und klicken Sie auf das Programm *Ereignisanzeige*. Klicken Sie in Ereignisprotokolle auf *Windows-Protokolle, Anwendung*.
  Windows 8/10: Drücken Sie die Windows-Taste und F, um den Charm *Suche* zu öffnen, wählen Sie *Einstellungen* unterhalb des Suchfelds und geben Sie *Ereignisprotokolle* in das Suchfeld ein. Klicken Sie auf das dann sichtbare Symbol *Ereignisprotokolle anzeigen*. Klicken Sie in Ereignisprotokolle auf *Windows-Protokolle, Anwendung*.
- Für die Filterereignisse von TET PDF IFilter wird ein Eintrag mit der Quelle *TET PDF IFilter* erzeugt. Doppelklicken Sie auf die Zeile mit dem entsprechenden Fehler und analysieren Sie die Fehlermeldung.

Einträge in der Ereignisprotokollierung der Anwendung lassen sich für verschiedene Ereignisklassen im folgenden Registry-Schlüssel aktivieren:

```
HKEY_LOCAL_MACHINE\SOFTWARE\PDFlib\TET PDF IFilter5\5.0\logging
```

Setzen Sie den Wert DWORD gemäß Tabelle 5.1. Der Logging-Level ist standardmäßig auf 1 gesetzt. Beachten Sie, dass ein PDF-Dokument in mehr als einer Fehlermeldung im Ereignisprotokoll auftauchen kann, jede beschädigte Seite kann unter Umständen zu einem separater Eintrag führen.

Tabelle 5.1 Logging-Level für die Windows-Ereignisanzeige

Level	*Zusammen_ fassung*	*Protokollierte Ereignisse*
0	*silent*	*Keins: alle Fehlermeldungen werden unterdrückt*
1 (Standard-wert)	*errors*	*Alle gescheiterten TET-Funktionsaufrufe und von TET ausgelösten Exceptions, z.B. ungültiger oder fehlender Lizenzschlüssel, verschlüsselte PDF-Dateien, die ein Benutzerkennwort benötigen, stark beschädigte PDF-Dokumente, die nicht repariert werden können; Probleme beim Lesen der Registry; Probleme bei der Verarbeitung Konfigurationsdatei.*
2	*activity*	*Wie 1, plus alle Aufrufe der IFilter-Schnittstelle vom Typ* `Load()` *und* `Init()`*; Lesen der XML-Konfigurationsdatei*
3	*details*	*Wie 2, plus Details zur den Aufrufen der IFilter-Schnittstelle zur Abfrage von Text und Properties einschließlich LCID. Dieser Level erzeugt eine große Anzahl an Einträgen.*

Welche Properties und welcher Text werden von TET PDF IFilter für ein Dokument ausgegeben? Mit Hilfe des Tools *FiltDump.exe* im Windows SDK lässt sich der genaue Text anzeigen, den TET PDF IFilter aus einem bestimmten Dokument extrahiert. Für die Prüfung der 64-Bit-Variante der DLL von TET PDF IFilter ist wiederum die 64-Bit-Variante dieses Tools ist erforderlich. Für weitere Information siehe *msdn.microsoft.com/en-us/library/ms692535(VS.85).aspx*

Mit der Option *-o* kann die Ausgabe von *FiltDump.exe* in eine UTF-16-kodierte Datei umgeleitet werden. Dadurch lässt sich der genaue Unicode-Text und die erfasste Spracheinstellung (LCID) für den Text anzeigen, der mit TET PDF IFilter ausgegeben wird. Beispieleingabe:

```
FiltDump.exe -o udhr_japanese.txt udhr_japanese.pdf
```

Beispielausgabe:

```
FILE: udhr_japanese.pdf
IFILTER: CLSID == {47A1AF35-C345-475D-AE68-EB07E948BD07}
IFILTER: Using IPersistStream
IFILTER: IFilter->Init returned IFILTER_FLAGS_OLE_PROPERTIES flag


CHUNK: ----------------------------------------------------------------
    Attribute = {007867F0-C59B-43FC-AB1E-8EEE77057254}\3 (Unknown)
    idChunk = 1
    BreakType = 2 (Sentence)
    Flags (chunkstate) =  (Value)
    Locale = 1031 (0x407)
    IdChunkSource = 1
    cwcStartSource = 0
    cwcLenSource = 0

VALUE: ----------------------------------------------------------------
Type = 31 (0x1f), VT_LPWSTR
Value = "4.0"

CHUNK: ----------------------------------------------------------------
    Attribute = {007867F0-C59B-43FC-AB1E-8EEE77057254}\4 (Unknown)
    idChunk = 3
    BreakType = 2 (Sentence)
    Flags (chunkstate) =  (Value)
    Locale = 1031 (0x407)
    IdChunkSource = 3
    cwcStartSource = 0
    cwcLenSource = 0

VALUE: ----------------------------------------------------------------
Type = 64 (0x40), VT_FILETIME
Value = "2010/06/10:08:28:04.587"

CHUNK: ----------------------------------------------------------------
    Attribute = {B725F130-47EF-101A-A5F1-02608C9EEBAC}\19 (System.Search.Contents)
    idChunk = 11
    BreakType = 2 (Sentence)
    Flags (chunkstate) =  (Text)
    Locale = 9 (0x9)
    IdChunkSource = 11
    cwcStartSource = 0
    cwcLenSource = 0

TEXT: ----------------------------------------------------------------
UDHR – Japanese

CHUNK: ----------------------------------------------------------------
    Attribute = {B725F130-47EF-101A-A5F1-02608C9EEBAC}\19 (System.Search.Contents)
```

```
    idChunk = 12
    BreakType = 2 (Sentence)
    Flags (chunkstate) =  (Text)
    Locale = 17 (0x11)
    IdChunkSource = 12
    cwcStartSource = 0
    cwcLenSource = 0

TEXT: ------------------------------------------------------------
...Textinhalt des Dokuments...
```

Protokollausgabe des TET-Kern. Sie können eine ausführliche Protokollierung für TET aktivieren, um das von TET PDF IFilter gesteuerte Verhalten des TET-Kerns zu analysieren. Die Protokollierung für TET lässt sich folgendermaßen aktivieren:

- Setzen Sie geeignete TET-Optionen in der XML-Konfigurationsdatei (Achten Sie darauf, den Dateinamen der XML-Konfigurationsdatei in der Registry zu setzen, siehe Kapitel 6, »XML-Konfigurationsdatei«, Seite 73):

  ```
  <Tet>
       <TetOptions>logging={filename=C:\debug.log classes={pcos=2}}</TetOptions>
  </Tet>
  ```

  Dies erstellt eine Protokolldatei mit Informationen über interne Aufrufe von TET-Funktionen, Fehlermeldungen usw. Achten Sie darauf, einen für den Service, der TET PDF IFilter aufruft, gültigen Dateinamen zu verwenden, und denken Sie daran, dass die TET-Protokollierung sehr viel Ausgabe erzeugt und den Filterprozess verlangsamt.
- Setzen Sie eine Umgebungsvariable mit Powershell:

  ```
  PS C:\> ${env:TET PDF IFILTERLOGGING} = "filename=tet.log classes={filesearch=3}"
  ```

6 XML-Konfigurationsdatei

6.1 Arbeiten mit Konfigurationsdateien

TET PDF IFilter kann mit Hilfe einer XML-Konfigurationsdatei gesteuert werden. Beispiel-Konfigurationsdateien werden mit TET PDF IFilter installiert.

Speicherort für die Konfigurationsdatei. Die Konfigurationsdatei kann im folgenden Registry-Schlüssel angegeben werden, der einen String mit dem vollständigen Pfadnamen der Konfigurationsdatei enthält:

```
HKEY_LOCAL_MACHINE\SOFTWARE\PDFlib\TET PDF IFilter5\configfile
```

Hinweis *Wir empfehlen, die Konfigurationsdatei nicht im Installationsverzeichnis von TET PDF IFilter abzulegen. Dadurch bleibt die Konfiguration auch nach einer Änderung am Installationsverzeichnis, z.B. nach einem Update von TET PDF IFilter, erhalten.*

Ist dieser Registry-Eintrag nicht vorhanden oder enthält er einen leeren String, wird die Standardkonfiguration verwendet. Kann die im Registry-Eintrag angegebene Konfigurationsdatei nicht geöffnet werden, wird eine Warnung in die Ereignisprotokollierung der Anwendung geschrieben und die Standardkonfiguration zur Indizierung verwendet. Schlägt die XML-Verarbeitung der Konfigurationsdatei fehl, wird eine Warnung in das Ereignisprotokoll geschrieben und keine Indizierung durchgeführt.

Hinweis *Das Installationsprogramm erzeugt keinen Registry-Eintrag für eine Konfigurationsdatei. Dieser muss bei Bedarf vom Benutzer erzeugt werden.*

Für TET PDF IFilter kann pro Computer jeweils nur eine Konfigurationsdatei verwendet werden. Jedoch können für 32-Bit- und 64-Bit-Versionen auf dem selben Computer unterschiedliche Konfigurationsdateien verwendet werden, da der obige Registry-Eintrag jeweils in der 32-Bit- und 64-Bit-Registry gesucht wird.

Nach Änderungen an der Konfigurationsdatei müssen Sie den Index neu erstellen, um die Änderungen zu aktivieren.

Vordefinierte XML-Konfigurationsdateien. Mehrere vordefinierte Beispiel-Konfigurationsdateien werden mit TET PDF IFilter installiert:

- Die Datei *default.xml* beschreibt die internen Standardeinstellungen von TET PDF IFilter. Sie kann als Basis für eine benutzerdefinierte Konfigurationsdatei verwendet werden.
- Die Datei *starter.xml* enthält Property-Definitionen, die mit den Starter-Beispielen verwendet werden können, die mit TET PDF IFilter installiert werden.

XML-Namensraum und -Schemabeschreibung. In Tabelle 6.1 werden die in der XML-Konfigurationsdatei verfügbaren Elemente und Attribute aufgeführt. Der URI des Namensraums ist

```
http://www.pdflib.com/XML/TET_PDF_IFilter3/TET_PDF_IFilter_Config-3.0.xsd
```

Hinweis *Der URI des Namensraums für das Schema sowie der Download-Ordner enthalten die Versionsnummer 3, da das aktuelle Schema mit dem in TET PDF IFilter 3.0 verwendeten Schema kompatibel ist.*

Eine XSD-Schemabeschreibung für die XML-Konfigurationssprache wird mit TET PDF IFilter installiert und ist auch an der obigen Adresse verfügbar, die als Namensraum-ID dient. Sie können die Schemadatei mit einem geeigneten XML-Editor bearbeiten um sicherzugehen, dass die Syntax für TET PDF IFilter gültig ist.

Benutzerdefinierte Datentypen für XML-Elemente und -Attribute. Alle Elemente sind leer, sofern kein Wert angegeben ist. Folgende benutzerdefinierte Datentypen werden in der XML-Konfigurationsdatei verwendet:

- LCID: hexadezimaler oder dezimaler Identifier für die Spracheinstellung; siehe

  msdn.microsoft.com/en-us/library/ms776294(VS.85).aspx

  Der Wert 0x0800 wird in die aktuelle Standard-Spracheinstellung des Systems übersetzt.
- GUID: eindeutiger 128-Bit-Identifier in hexadezimaler Schreibweise gemäß ITU-T Rec. X.667 (siehe *www.itu.int/ITU-T/studygroups/com17/oid/X.667-E.pdf)*. Die einzelnen Bestandteile müssen durch Bindestriche »-« getrennt werden. Zur Erzeugung von GUIDs stehen mehrere Tools zur Verfügung; Sie können auch Online-Dienste verwenden, siehe z.B. unter

  www.itu.int/ITU-T/asn1/uuid.html
- pCOS-Pfad: erweiterter pCOS-Pfad für ein PDF-Objekt, siehe das pCOS-Referenzhandbuch sowie die pCOS-Erweiterungen in »Erweiterte pCOS-Pfade«, Seite 42
- Optionsliste: String mit einer Optionsliste gemäß der Syntax im *PDFlib TET Referenzhandbuch*.
- Sprach-Identifier: XMP-Sprachangabe gemäß RFC 1766 oder *x-default,* was die Standardsprache im Dokument bezeichnet.

6.2 XML-Elemente und -Attribute

Tabelle 6.1 enthält Einzelheiten zu Elementen und Attributen der XML-Konfigurationsdatei. Ausführliche Informationen zur XML-Konfiguration finden Sie in den entsprechenden Abschnitten dieses Handbuchs. Die erforderlichen und empfohlenen Einstellungen für bestimmte IFilter-Clients werden in den Client-spezifischen Abschnitten in Kapitel 3, »Indizierung von Metadaten«, Seite 41 aufgeführt.

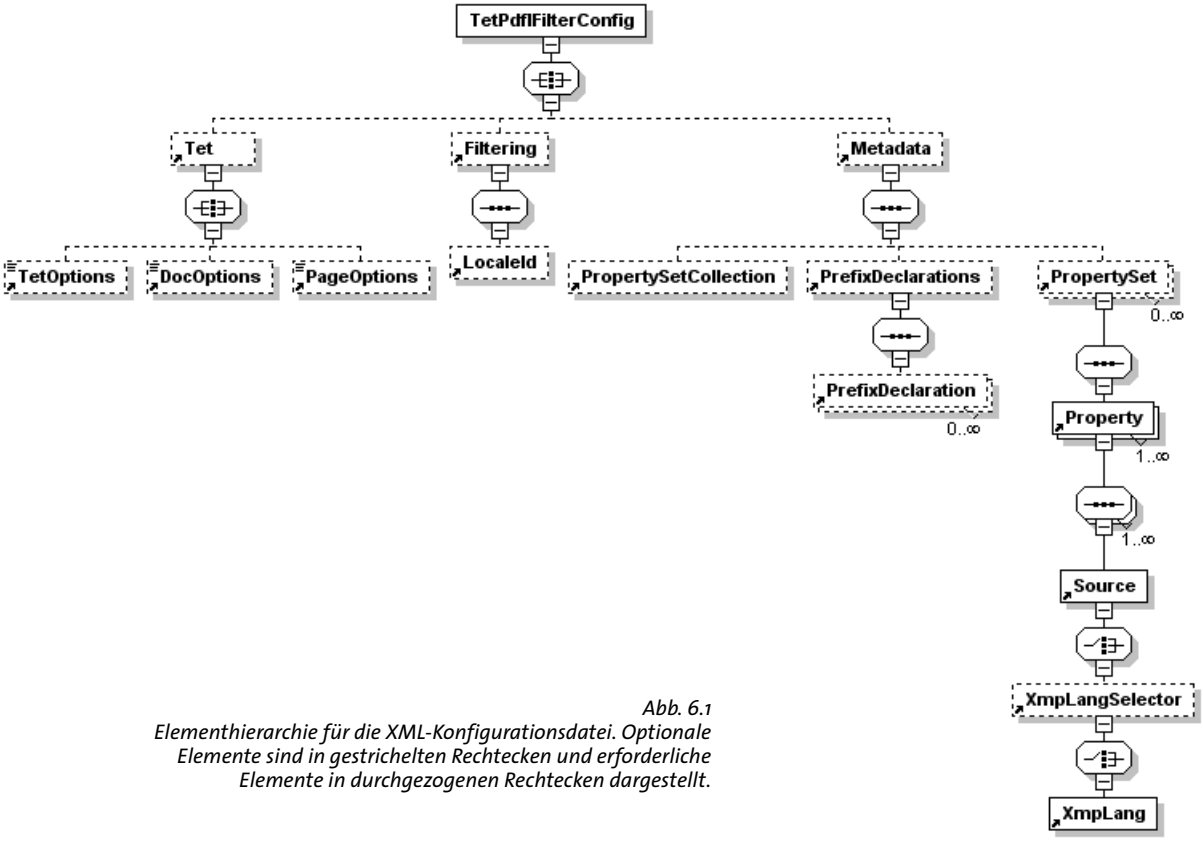


Abb. 6.1
Elementhierarchie für die XML-Konfigurationsdatei. Optionale Elemente sind in gestrichelten Rechtecken und erforderliche Elemente in durchgezogenen Rechtecken dargestellt.

Tabelle 6.1 XML-Elemente und -Attribute in der Konfigurationsdatei

Element	Beschreibung und Attribute
DocOptions *übergeordnetes Element: Tet*	*(Kann null oder einmal auftreten) Optionsliste für die TET-Funktion TET_open_document( ).*
Filtering *übergeordnetes Element: TetPdfIFilterConfig*	*(Kann null oder einmal auftreten) Informationen zur Verarbeitung von PDF-Filtern. Unterstützte Attribute:* ***errorIndicator*** *(String; optional) An den IFilter-Client übergebener String, wenn während der Verarbeitung eines Dokuments ein TET-Funktionsaufruf fehlschlägt. Informationen zu dem Problem finden Sie in der Windows-Ereignisanzeige (siehe »Ereignisprotokollierung der Anwendung«, Seite 70). Der Fehlerindikator kann zur Identifizierung des Problems bei der Indizierung nützlich sein. Er wird zusätzlich zum Text(ausschnitt) ausgegeben, der vom Dokument abgefragt werden kann. Wir empfehlen, einen eindeutigen String ohne Satzzeichen zu übergeben, damit der Fehlerindikator nicht mit den tatsächlichen Indexeinträgen in Konflikt gerät, z.B.* `TETPDFIFILTERERROR`*. Standardwert: kein Fehlerindikator* ***indexNestedPdf*** *(Boolean; optional) Rekursive Verarbeitung von PDF-Anhängen (siehe Abschnitt 2.1, »PDF-Dokumentdomänen«, Seite 21). Standardwert:* `true` ***indexPageContents*** *(Boolean; optional) Angabe, ob die Inhalte von PDF-Seiten indiziert werden oder nicht. Das Deaktivieren der Indizierung von Seiteninhalten kann in Szenarien nützlich sein, in denen ausschließlich nach Metadaten-Properties gesucht wird. Standardwert:* `true` ***metadataHandling*** *(Auswahl; optional) Auswahl des Typs der Metadaten-Verarbeitung (siehe Abschnitt 3.6, »Indizieren von Metadaten-Properties als Text«, Seite 50). Standardwert:* `property` ***ignore*** *Alle Metadaten-Properties werden ignoriert. Dies kann bei der Fehlersuche oder zur Leistungsoptimierung nützlich sein, wenn keine Metadaten erforderlich sind.* ***property*** *Verarbeitung von Metadaten als Properties* ***propertyAndPrefixedText*** *Verarbeitung von Metadaten als Properties und zusätzlich Voranstellen des in* `textIndexPrefix` *angegebenen Präfix (falls vorhanden) für benutzerdefinierte Properties und die Präfixe gemäß Tabelle 3.2, Seite 50 sowie für vordefinierte Properties. Ergebnisse werden zusätzlich als reiner Text weitergegeben.* ***propertyAndText*** *Verarbeitung von Metadaten als Properties und zusätzlich als reiner Text.* ***useIdentifier*** *(Boolean; optional) Legt fest, ob Properties mit* `identifier` *oder* `friendlyName` *identifiziert werden, wenn für das Element* `Property` *beide Attribute vorhanden sind. Standardwert:* `true`

Tabelle 6.1 XML-Elemente und -Attribute in der Konfigurationsdatei

Element	Beschreibung und Attribute
LocaleId *übergeordnetes Element: Filtering*	*(Kann null oder einmal auftreten) Konfiguriert die Erkennung der Sprach-ID (siehe Abschnitt 2.2, »Automatische Spracherkennung«, Seite 27). Unterstützte Attribute:* **arabic** *(LCID; optional) LCID für arabischen Text. Standardwert: 0x0401 Arabisch (SA)* **chinese** *(LCID; optional) LCID für chinesischen Text. Standardwert: 0x0804 Chinesisch (Volksrepublik China)* **cyrillic** *(LCID; optional) LCID für kyrillischen Text. Standardwert: 0x0419 Russisch (RU)* **default** *(LCID; optional) Globaler LCID für alle Textabschnitte, wenn die Erkennung deaktiviert ist. Standardwert: 0x0800 (Spracheinstellung des Systems)* **detection** *(Auswahl; optional) Steuert die automatische LCID-Erkennung. Standardwert:* `auto` **auto** *Erkennung des LCID durch Schriftsystem und statistische Sprachanalyse* **disabled** *Deaktiviert LCID-Erkennung; alle anderen Attribute außer* `default` *und* `useCatalogLang` *werden ignoriert.* **script** *(TET PDF IFilter 4.0) Schriftsystem-basierte Erkennung des LCID* **latin** *(LCID; optional) LCID für lateinischen Text. Standardwert: 0x0409 Englisch (US)* **useCatalogLang** *(Boolean; optional; TET PDF IFilter 4.0) Legt fest, ob der Eintrag* `Lang` *im Dokumentkatalog ausgewertet wird. Bei* `true` *prüft TET PDF IFilter den Eintrag* `Lang` *im PDF-Dokumentkatalog. Falls vorhanden, wird der Eintrag* `Lang` *in einen LCID konvertiert. Bei erfolgreicher Konvertierung überschreibt der LCID den Wert des Attributs* `LocaleId/@default`**;** *wenn der LCID zum arabischen, chinesischen, kyrillischen oder lateinischen Schriftsystem gehört, überschreibt er den Wert des entsprechenden Attributs des Elements* `LocaleId`*. Standardwert:* `true`
Metadata *übergeordnetes Element: TetPdfIFilterConfig*	*(Kann null oder einmal auftreten) Legt Metadaten-Properties fest (siehe Abschnitt 3.4, »Benutzerdefinierte Metadaten-Properties«, Seite 46). Falls vorhanden, muss dieses Element nach* `Filtering` *und* `Tet` *auftreten.*
PageOptions *übergeordnetes Element: Tet*	*(Kann null oder einmal auftreten) Optionsliste für die TET-Funktion TET_open_page( ).*
PrefixDeclaration *übergeordnetes Element: PrefixDeclarations*	*(Kann null oder mehrmals auftreten) Deklariert ein Namensraum-Präfix, das in* `Source/@xmpName` *verwendet werden kann. Unterstützte Attribute:* **prefix** *(String ohne Doppelpunkte »:«; erforderlich) Präfix als Abkürzung für den URI des Namensraums.* **uri** *(URI; erforderlich) URI des Namensraums*
PrefixDeclarations *übergeordnetes Element: Metadata*	*(Kann null oder einmal auftreten) Deklariert Namensraum-Präfixe für XMP-Properties im Attribut* `xmpName` *des Elements* `Source`*.*

Tabelle 6.1 XML-Elemente und -Attribute in der Konfigurationsdatei

Element	Beschreibung und Attribute
Property *übergeordnetes Element: PropertySet*	*(Kann ein- oder mehrmals auftreten) Legt eine Metadaten-Property für die Indizierung fest (siehe Abschnitt 3.4, »Benutzerdefinierte Metadaten-Properties«, Seite 46).* *Mindestens eins von* `identifier` *und* `friendlyName` *muss vorhanden sein. Werden beide übergeben, wird* `identifier` *an der IFilter-Schnittstelle verwendet, sofern nicht* `Filtering/@useIdentifier=false`. *Unterstützte Attribute:* **identifier** *(Integer >=2; optional) Zahl, die die Property in einem* `PropertySetCo` *eindeutig identifiziert.* **emitAsVector** *(Boolean; optional) Bei* `true` *wird der Wert der Property unabhängig von der Anzahl der Werte als ein einzelnes Vektorelement ausgegeben. Bei* `false` *wird die Property als fester Wert ausgegeben. Wurde mehr als ein Quellelement gefunden, werden mehrere Einzel-Properties ausgegeben. Standardwert:* `false` **friendlyName** *(String; optional) Eindeutiger Name der Property in einem* `PropertySet`. *Kann zur Dokumentierung der Property oder als Alternative zu* `identifier` *verwendet werden.* **precedence** *(Auswahl; optional) Vorrang für mehrere Elemente* `Source` *(Standardwert:* `first-wins`*):* **first-wins** *Die erste nicht leere Quelle wird verwendet.* **try-all** *Alle nicht leeren Quellen tragen zur Property bei.* **textIndexPrefix** *(String; optional) String, der der Property vorangestellt wird, wenn* `Filtering/@metadataHandling` *gleich* `propertyAndPrefixedText` *ist. Standardwert: leer* **type** *(Auswahl; optional) Windows-Datentyp für die Metadaten-Property. Unterstützte Datentypen sind* `Boolean`, `DateTime`, `Double`, `Int32` *und* `String`. *Standardwert:* `String`
PropertySet *übergeordnetes Element: Metadata*	*(Kann null oder mehrmals auftreten) Filtern eines benutzerdefinierten Property-Sets mit der gleichen GUID (siehe Abschnitt 3.4, »Benutzerdefinierte Metadaten-Properties«, Seite 46).* *Falls vorhanden, muss dieses Element nach* `PropertySetCollection` *und* `PrefixDeclarations` *auftreten.* *Unterstützte Attribute:* **guid** *(GUID; erforderlich) Eindeutiger 128-Bit-Identifier für das Property-Set in hexadezimaler Schreibweise.*
PropertySetCollection *übergeordnetes Element: Metadata*	*(Kann null oder mehrmals auftreten) Filtern von vordefinierten Sammlungen von Property-Sets (siehe Abschnitt 3.3, »Vordefinierte Metadaten-Properties«, Seite 45). Eine Liste aller Properties finden Sie in Anhang A, »Vordefinierte Metadaten-Properties«. Unterstützte Attribute:* **documentXmp** *(Boolean; erforderlich) Ausgabe von XMP-Properties auf Dokumentebene. Standardwert:* `false` **imageXmp** *(Boolean; erforderlich) Ausgabe von XMP-Properties von Bildern. Standardwert:* `false` **internal** *(Boolean; erforderlich) Ausgabe interner Properties von TET PDF IFilter. Standardwert:* `true` **pdf** *(Boolean; erforderlich) Ausgabe von PDF-spezifischen Properties. Standardwert:* `false` **shell** *(Boolean; erforderlich) Ausgabe von Shell-Properties. Standardwert:* `true`

Tabelle 6.1 XML-Elemente und -Attribute in der Konfigurationsdatei

Element	*Beschreibung und Attribute*
Source *übergeordnetes Element: Property*	*(Kann ein- oder mehrmals auftreten) Eine oder mehrere Quellen für eine Metadaten-Property.* *Die Reihenfolge der Elemente ist relevant, wenn* `Property/@precedence` *den Wert* `first-wins` *hat. Mindestens eins der unten aufgeführten Attribute muss übergeben werden.* *Unterstützte Attribute:* ***pdfObject*** *(pCOS-Pfad; optional) Erweiterter pCOS-Pfad für ein oder mehrere PDF-Objekte vom Typ Boolean, Zahl, Name oder String, die die Property enthalten. Standardwert: /* `Root/Metadata` *(d.h. XMP auf Dokumentebene)* ***xmpName*** *(String aus dem Schema-Präfix, einem Doppelpunkt »:« und dem Property-Namen; optional) Vollständig qualifizierter Name der XMP-Property. Statt des URI des Namensraums kann ein Präfix verwendet werden, sofern es in einem Element* `PrefixDeclaration` *deklariert wurde. Dieses Attribut wird nur verwendet, wenn* `pdfobject` *auf ein oder mehrere XMP-Streams verweist. Standardwert: leer*
Tet *übergeordnetes Element: TetPdfIFilterConfig*	*(Kann null oder mehrmals auftreten) Verarbeitungsoptionen für den TET-Kern; siehe das TET-Referenzhandbuch für eine Beschreibung der Syntax von Optionslisten sowie verfügbare Optionen. Einige Optionen werden von TET PDF IFilter überschrieben.*
TetOptions *übergeordnetes Element: Tet*	*(Kann null oder einmal auftreten) Optionsliste für die TET-Funktion TET_set_option( ).*
TetPdfIFilterConfig *übergeordnetes Element: keins*	*(Muss genau einmal als Stammelement auftreten) Stammelement der XML-Konfigurationsdatei. Unterstütztes Attribut:* ***version*** *(String; optional; TET PDF IFilter 4.0) Version von TET PDF IFilter, für das diese Konfiguration erstellt wurde. Da die Konfigurationsdatei von TET PDF IFilter 3 dieses Attribut nicht unterstützt, ist der Standardwert* `3.0`*. Neue Konfigurationen sollten dieses Attribut mit der korrekten Versionsangabe* `(4.0` *für TET PDF IFilter 4 enthalten. Standardwert:* `3.0`
XmpLang *übergeordnetes Element: XmpLangSelector*	*(Muss genau einmal auftreten, wenn* `XmpLangSelector/@languages=subset`*) Sprache einer XMP-Property. Unterstütztes Attribut:* ***lang*** *(Sprach-ID; erforderlich). Name der Sprache; zur Zeit ist* `x-default` *der einzige unterstützte Wert.*
XmpLangSelector *übergeordnetes Element: Xmp*	*(Kann null oder einmal auftreten) Wählt eine Sprachvariante einer XMP-Property für die Indizierung. Dies ist nur für Properties mit einer XMP-Quelle vom Typ* `LangAlt` *relevant. Unterstütztes Attribut:* ***languages*** *(Auswahl) Sprachspezifische Indizierung der Property (Standardwert:* `all`*):* ***all*** *Alle verfügbaren Spracheinträge der Property werden indiziert.* ***subset*** *Nur die im Element* `XmpLang` *angegebenen Sprachen werden indiziert.*

6.3 Beispiel für XML-Konfigurationsdatei

Die folgende Auflistung zeigt eine vollständige XML-Konfigurationsdatei für TET PDF IFilter:

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
        XML configuration file for TET PDF IFilter
        (c) PDFlib GmbH 2008-2015 www.pdflib.com
        This file must be configured in the following registry key:
        HKEY_LOCAL_MACHINE\SOFTWARE\PDFlib\TET PDF IFilter5\configfile
-->

<n:TetPdfIFilterConfig
    xmlns:n="http://www.pdflib.com/XML/TET_PDF_IFilter3/TET_PDF_IFilter_Config-3.0.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.pdflib.com/XML/TET_PDF_IFilter3/TET_PDF_IFilter_
Config-3.0.xsd http://www.pdflib.com/XML/TET_PDF_IFilter3/TET_PDF_IFilter_Config-3.0.xsd"
        version="4.0">

<n:Tet>
        <n:TetOptions></n:TetOptions>
        <n:DocOptions></n:DocOptions>
        <n:PageOptions></n:PageOptions>
</n:Tet>

<n:Filtering indexNestedPdf="true" metadataHandling="property" useIdentifier="true">
        <n:LocaleId
            detection="auto"
            useCatalogLang="true"
            default="0x0800"
            arabic="0x0401"
            chinese="0x0804"
            cyrillic="0x0419"
            latin="0x0409"/>
</n:Filtering>

<n:Metadata>
        <n:PropertySetCollection
            documentXmp="false"
            imageXmp="false"
            internal="true"
            pdf="false"
            shell="true"/>

        <n:PropertySet guid="b01ca440-1b9f-11dd-8b87-0002a5d5c51b">
            <n:Property identifier="2">
                <n:Source pdfObject="/Info/Producer"/>
            </n:Property>
        </n:PropertySet>
</n:Metadata>

</n:TetPdfIFilterConfig>
```

A Vordefinierte Metadaten-Properties

Die in Tabelle A.1 aufgeführten Properties sind in TET PDF IFilter implementiert und müssen daher nicht gesondert konfiguriert werden. Um diese Properties zu verwenden, müssen Sie lediglich die gewünschten Sammlungen von Property-Sets in der XML-Konfigurationsdatei aktivieren (siehe Abschnitt 3.3, »Vordefinierte Metadaten-Properties«, Seite 45). Für Windows Search müssen Sie die Properties mit dem Tool *registerpropdesc.exe* registrieren.

Tabelle A.1 Verarbeitung von Properties in TET PDF IFilter: vordefinierte Sammlungen von Property-Sets

Property-Name für Windows Search; auch verwendet, um ein Präfix abzuleiten, falls die Property als Text indiziert wird.	***Datentyp***	***mehr als ein Wert***	***GUID des Property-Sets/ID der Property***	***Quelle: XMP-Property oder pCOS-Pfad***
Sammlung von Shell-Property-Sets				
System.Document.Contributor	*String*	*ja*	*F334115E-DA1B-4509-9B3D-119504DC7ABB/100*	*dc:contributor*
System.Document.DateCreated	*DateTime*	*nein*	*F29F85E0-4FF9-1068-AB91-08002B27B3D9/12*	*xmp:CreateDate /Info/CreationDate*
System.Document.DateSaved	*DateTime*	*nein*	*F29F85E0-4FF9-1068-AB91-08002B27B3D9/13*	*xmp:ModifyDate, /Info/ModDate*
System.Document.DocumentID	*String*	*nein*	*E08805C8-E395-40DF-80D2-54F0D6C43154/100*	*dc:identifier*
System.Document.PageCount	*Int32*	*nein*	*F29F85E0-4FF9-1068-AB91-08002B27B3D9/14*	*length:pages*
System.Document.Version	*String*	*nein*	*D5CDD502-2E9C-101B-9397-08002B2CF9AE/29*	*xmpMM:VersionID*
System.Search.Contents	*String*	*ja*	*B725F130-47EF-101A-A5F1-02608C9EEBAC/19*	*Textinhalte der PDF-Seiten*
System.Title	*String*	*nein*	*F29F85E0-4FF9-1068-AB91-08002B27B3D9/2*	*dc:title["x-default"], /Info/Title*
System.Subject	*String*	*nein*	*F29F85E0-4FF9-1068-AB91-08002B27B3D9/3*	*dc:description["x-default"], /Info/Subject*
System.Author	*String*	*ja*	*F29F85E0-4FF9-1068-AB91-08002B27B3D9/4*	*dc:creator, pdf:Author, xmp:Author, /Info/Author*
System.Keywords	*String*	*ja*	*F29F85E0-4FF9-1068-AB91-08002B27B3D9/5*	*pdf:Keywords, /Info/Keywords*
System.MIMEType	*String*	*nein*	*0B63E350-9CCC-11D0-BCDB-00805FCCCE04/5*	`application/pdf` *(festgelegt)*
System.DateModified (IS: Write)	*DateTime*	*nein*	*B725F130-47EF-101A-A5F1-02608C9EEBAC/14*	*xmp:ModifyDate, /Info/ModDate*
System.ApplicationName	*String*	*nein*	*F29F85E0-4FF9-1068-AB91-08002B27B3D9/18*	*xmp:CreatorTool, /Info/Creator*
System.Kind	*String*	*nein*	*1E3EE840-BC2B-476C-8237-2ACD1A839B22/3*	`Document` *(festgelegt)*

Tabelle A.1 Verarbeitung von Properties in TET PDF IFilter: vordefinierte Sammlungen von Property-Sets

Property-Name für Windows Search; auch verwendet, um ein Präfix abzuleiten, falls die Property als Text indiziert wird.	**Datentyp**	**mehr als ein Wert**	**GUID des Property-Sets/ID der Property**	**Quelle: XMP-Property oder pCOS-Pfad**
Sammlung von PDF-Property-Sets				
PDFlib.TETPDFIFilter.pdfversion (enthält die PDF-Version multipliziert mit 10, z.B. »16«)	String	nein	E544AFE6-13E2-40F1-A702-DCEBE8FB7B02/2	pdfversion
PDFlib.TETPDFIFilter.pdfa	String	nein	E544AFE6-13E2-40F1-A702-DCEBE8FB7B02/3	pdfa
PDFlib.TETPDFIFilter.pdfx	String	nein	E544AFE6-13E2-40F1-A702-DCEBE8FB7B02/4	pdfx
PDFlib.TETPDFIFilter.font	String	ja	E544AFE6-13E2-40F1-A702-DCEBE8FB7B02/5	fonts[*]/name
PDFlib.TETPDFIFilter.bookmark	String	ja	E544AFE6-13E2-40F1-A702-DCEBE8FB7B02/6	bookmarks[*]/Title
PDFlib.TETPDFIFilter.annotation	String	ja	E544AFE6-13E2-40F1-A702-DCEBE8FB7B02/7	pages[*]/annots[*]/Contents
PDFlib.TETPDFIFilter.width	Double	ja	E544AFE6-13E2-40F1-A702-DCEBE8FB7B02/8	pages[*]/width
PDFlib.TETPDFIFilter.height	Double	ja	E544AFE6-13E2-40F1-A702-DCEBE8FB7B02/9	pages[*]/height
PDFlib.TETPDFIFilter.producer	String	nein	E544AFE6-13E2-40F1-A702-DCEBE8FB7B02/10	/Info/Producer
PDFlib.TETPDFIFilter.trapped	String	nein	E544AFE6-13E2-40F1-A702-DCEBE8FB7B02/11	/Info/Trapped
Sammlung von Property-Sets zu XMP-Dokumentmetadaten (aus der XMP-Spezifikation 2005)				
Dublin Core				
PDFlib.TETPDFIFilter.dc.contributor	String	ja	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/2	dc:contributor
PDFlib.TETPDFIFilter.dc.coverage	String	nein	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/3	dc:coverage
PDFlib.TETPDFIFilter.dc.creator	String	ja	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/4	dc:creator
PDFlib.TETPDFIFilter.dc.date	DateTime	ja	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/5	dc:date
PDFlib.TETPDFIFilter.dc.description	String	ja	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/6	dc:description
PDFlib.TETPDFIFilter.dc.format	String	nein	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/7	dc:format
PDFlib.TETPDFIFilter.dc.identifier	String	nein	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/8	dc:identifier
PDFlib.TETPDFIFilter.dc.language	String	ja	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/9	dc:language
PDFlib.TETPDFIFilter.dc.publisher	String	ja	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/10	dc:publisher
PDFlib.TETPDFIFilter.dc.relation	String	ja	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/11	dc:relation

Tabelle A.1 Verarbeitung von Properties in TET PDF IFilter: vordefinierte Sammlungen von Property-Sets

Property-Name für Windows Search; auch verwendet, um ein Präfix abzuleiten, falls die Property als Text indiziert wird.	**Datentyp**	**mehr als ein Wert**	**GUID des Property-Sets/ID der Property**	**Quelle: XMP-Property oder pCOS-Pfad**
PDFlib.TETPDFIFilter.dc.rights	String	ja	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/12	dc:rights
PDFlib.TETPDFIFilter.dc.source	String	nein	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/13	dc:source
PDFlib.TETPDFIFilter.dc.subject	String	ja	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/14	dc:subject
PDFlib.TETPDFIFilter.dc.title	String	ja	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/15	dc:title
PDFlib.TETPDFIFilter.dc.type	String	ja	D92BB3CA-CE2B-4B9B-972A-5BF54B468171/16	dc:type
XMP Basic				
PDFlib.TETPDFIFilter.xmp.Advisory	String	ja	C60E822A-074F-4BD5-9889-6EBD372F2000/2	xmp:Advisory
PDFlib.TETPDFIFilter.xmp.BaseURL	String	nein	C60E822A-074F-4BD5-9889-6EBD372F2000/3	xmp:BaseURL
PDFlib.TETPDFIFilter.xmp.CreateDate	DateTime	nein	C60E822A-074F-4BD5-9889-6EBD372F2000/4	xmp:CreateDate
PDFlib.TETPDFIFilter.xmp.CreatorTool	String	nein	C60E822A-074F-4BD5-9889-6EBD372F2000/5	xmp:CreatorTool
PDFlib.TETPDFIFilter.xmp.Identifier	String	ja	C60E822A-074F-4BD5-9889-6EBD372F2000/6	xmp:Identifier
PDFlib.TETPDFIFilter.xmp.Label	String	nein	C60E822A-074F-4BD5-9889-6EBD372F2000/7	xmp:Label
PDFlib.TETPDFIFilter.xmp.MetadataDate	DateTime	nein	C60E822A-074F-4BD5-9889-6EBD372F2000/8	xmp:MetadataDate
PDFlib.TETPDFIFilter.xmp.ModifyDate	DateTime	nein	C60E822A-074F-4BD5-9889-6EBD372F2000/9	xmp:ModifyDate
PDFlib.TETPDFIFilter.xmp.Nickname	String	nein	C60E822A-074F-4BD5-9889-6EBD372F2000/10	xmp:Nickname
PDFlib.TETPDFIFilter.xmp.Rating	Int32	nein	C60E822A-074F-4BD5-9889-6EBD372F2000/11	xmp:Rating
XMP Rights Management				
PDFlib.TETPDFIFilter.xmpRights.Certificate	String	nein	0DE7A11C-E2C5-4EFA-8017-BECD888E7EC9/2	xmpRights:Certificate
PDFlib.TETPDFIFilter.xmpRights.Marked	Boolean	nein	0DE7A11C-E2C5-4EFA-8017-BECD888E7EC9/3	xmpRights:Marked
PDFlib.TETPDFIFilter.xmpRights.Owner	String	ja	0DE7A11C-E2C5-4EFA-8017-BECD888E7EC9/4	xmpRights:Owner
PDFlib.TETPDFIFilter.xmpRights.UsageTerms	String	ja	0DE7A11C-E2C5-4EFA-8017-BECD888E7EC9/5	xmpRights:UsageTerms
PDFlib.TETPDFIFilter.xmpRights.WebStatement	String	nein	0DE7A11C-E2C5-4EFA-8017-BECD888E7EC9/6	xmpRights:WebStatement
XMP Basic Job Ticket				
PDFlib.TETPDFIFilter.xmpBJ.JobRef	String	ja	EBC983EF-C1CF-45C8-A29E-993543A0ECFB/2	xmpBJ:JobRef

Tabelle A.1 Verarbeitung von Properties in TET PDF IFilter: vordefinierte Sammlungen von Property-Sets

Property-Name für Windows Search; auch verwendet, um ein Präfix abzuleiten, falls die Property als Text indiziert wird.	**Datentyp**	**mehr als ein Wert**	**GUID des Property-Sets/ID der Property**	**Quelle: XMP-Property oder pCOS-Pfad**
XMP Paged-Text				
PDFlib.TETPDFIFilter.xmpTPg.NPages	Int32	nein	7A9EB492-35AB-49FE-B364-A21FC9575C28/2	xmpTPg:NPages
PDFlib.TETPDFIFilter.xmpTPg.PlateNames	String	ja	7A9EB492-35AB-49FE-B364-A21FC9575C28/3	xmpTPg:PlateNames
Adobe PDF				
PDFlib.TETPDFIFilter.pdf.Keywords	String	nein	17EB8447-FC9B-4D4D-81DF-31E9AA770CBF/2	pdf:Keywords
PDFlib.TETPDFIFilter.pdf.PDFVersion	String	nein	17EB8447-FC9B-4D4D-81DF-31E9AA770CBF/3	pdf:PDFVersion
PDFlib.TETPDFIFilter.pdf.Producer	String	nein	17EB8447-FC9B-4D4D-81DF-31E9AA770CBF/4	pdf:Producer
Sammlung von Property-Sets zu XMP-Bildmetadaten (aus der XMP-Spezifikation 2005)				
Photoshop				
PDFlib.TETPDFIFilter.images.photoshop.AuthorsPosition	String	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/2	photoshop:AuthorsPosition
PDFlib.TETPDFIFilter.images.photoshop.CaptionWriter	String	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/3	photoshop:CaptionWriter
PDFlib.TETPDFIFilter.images.photoshop.Category	String	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/4	photoshop:Category
PDFlib.TETPDFIFilter.images.photoshop.City	String	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/5	photoshop:City
PDFlib.TETPDFIFilter.images.photoshop.Country	String	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/6	photoshop:Country
PDFlib.TETPDFIFilter.images.photoshop.Credit	String	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/7	photoshop:Credit
PDFlib.TETPDFIFilter.images.photoshop.DateCreated	DateTime	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/8	photoshop:DateCreated
PDFlib.TETPDFIFilter.images.photoshop.Headline	String	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/9	photoshop:Headline
PDFlib.TETPDFIFilter.images.photoshop.Instructions	String	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/10	photoshop:Instructions
PDFlib.TETPDFIFilter.images.photoshop.Source	String	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/11	photoshop:Source
PDFlib.TETPDFIFilter.images.photoshop.State	String	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/12	photoshop:State
PDFlib.TETPDFIFilter.images.photoshop.SupplementalCategories	String	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/13	photoshop:SupplementalCategories
PDFlib.TETPDFIFilter.images.photoshop.TransmissionReference	String	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/14	photoshop:TransmissionReference
PDFlib.TETPDFIFilter.images.photoshop.Urgency	Int32	ja	C9F08C60-189D-11DD-8441-0002A5D5C51B/15	photoshop:Urgency

Tabelle A.1 Verarbeitung von Properties in TET PDF IFilter: vordefinierte Sammlungen von Property-Sets

Property-Name für Windows Search; auch verwendet, um ein Präfix abzuleiten, falls die Property als Text indiziert wird.	**Datentyp**	**mehr als ein Wert**	**GUID des Property-Sets/ID der Property**	**Quelle: XMP-Property oder pCOS-Pfad**
Sammlung von internen Property-Sets				
PDFlib.TETPDFIFilter.version	*String*	*nein*	*007867F0-C59B-43FC-AB1E-8EEE77057254/2*	`5.0` *(festgelegt)*
PDFlib.TETPDFIFilter.tetversion	*String*	*nein*	*007867F0-C59B-43FC-AB1E-8EEE77057254/3*	`5.0` *(festgelegt)*
PDFlib.TETPDFIFilter.indextime	*DateTime*	*nein*	*007867F0-C59B-43FC-AB1E-8EEE77057254/4*	*Datum und Uhrzeit der Indizierung*
PDFlib.TETPDFIFilter.eval	*Int32*	*nein*	*007867F0-C59B-43FC-AB1E-8EEE77057254/5*	*Nummer der Exception, wenn IFilter im Evaluierungsmodus läuft*

B Änderungen an diesem Handbuch

Änderungen an diesem Handbuch

Datum	*Änderungen*
12. August 2016	► *Deutsche Übersetzung des Handbuchs zu TET PDF IFilter 5.0r1*
27. Oktober 2015	► *Deutsche Übersetzung des Handbuchs zu TET PDF IFilter 5.0*
18. Juli 2015	► *Deutsche Übersetzung des Handbuchs zu TET PDF IFilter 4.4*

Index

PDFlib GmbH

PDFlib GmbH
Franziska-Bilek-Weg 9
D-80339 München
www.pdflib.com
Tel. +49 • 89 • 452 33 84-0
Fax +49 • 89 • 452 33 84-99

Bei Fragen können Sie die PDFlib-Mailing-Liste abonnieren
und sich deren Archiv ansehen unter groups.yahoo.com/neo/groups/pdflib/info

Vertriebsinformationen
sales@pdflib.com

Support
support@pdflib.com *(geben Sie bitte immer Ihre Lizenznummer an)*