

Individuelle IT-Konzepte und Softwarelösungen



Gesellschaft für
Informations- und
Kommunikationssysteme mbH

Von MediaWiki und Movable Type zu Confluence

gearconf 2011

City Hostel, Düsseldorf

10.06.2011

Autor:

Hans-Jörg Stangor

Agenda

- **Wiki und Weblog im Unternehmenseinsatz**
- **Problem nach Einführung**
- **Vorzüge von Confluence im Unternehmenseinsatz**
- **Migration bestehender Inhalte**
- **Erreichte Ziele**

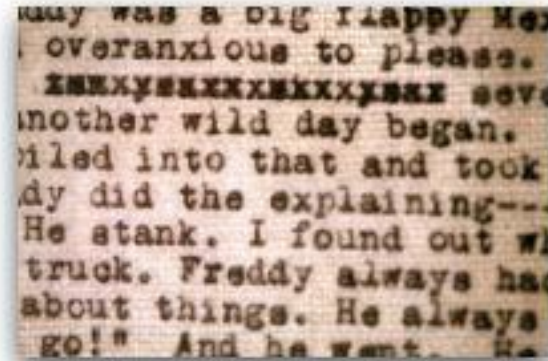
Agenda

- **Wiki und Weblog im Unternehmenseinsatz**
- Problem nach Einführung
- Vorteile von Confluence im Unternehmenseinsatz
- Migration bestehender Inhalte
- Erreichte Ziele

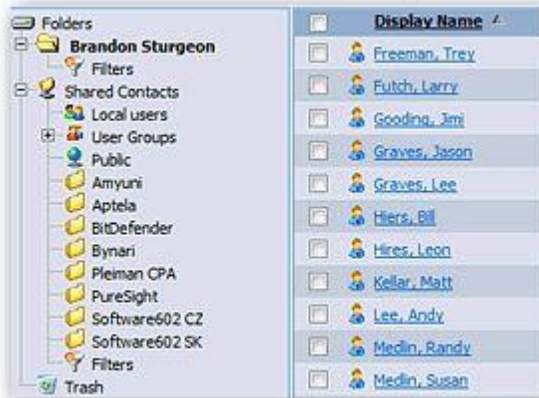
Kernprobleme in der Unternehmenskommunikation



Email



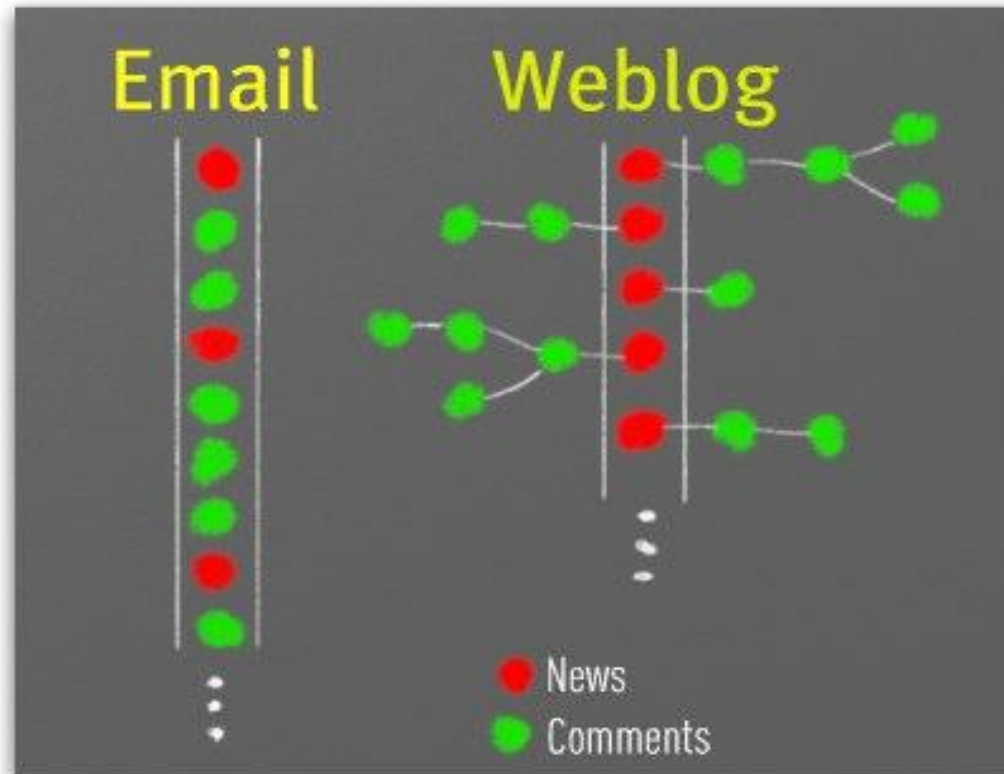
Dokumentenversionen



Ansprechpartner

**ZU VIEL VON
ALLEM!!!**

Blogs / Weblogs



- Lösung für die Probleme „Ansprechpartner“ und „Email“

Unsere Entscheidung für Movable Type

● **Multiblog-System**

- Wordpress zu diesem Zeitpunkt noch Single-Blog
- Wordpress Mu nicht produktiv einsetzbar

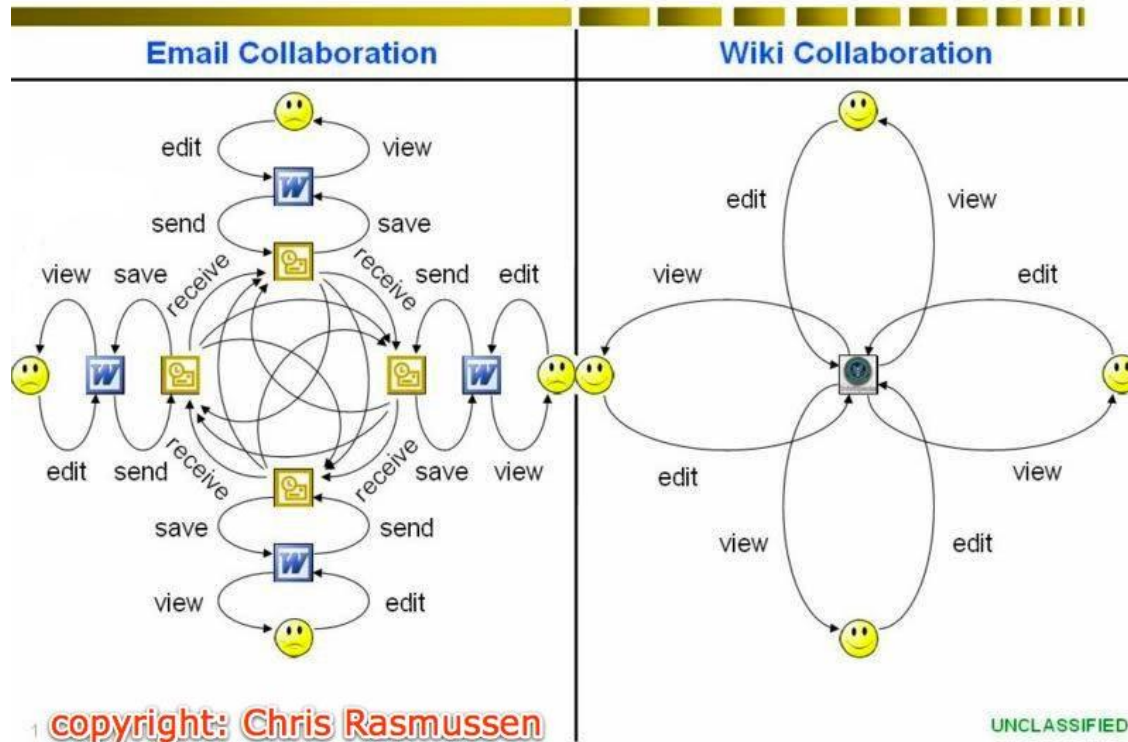
● **Typepad („MT as a Service“) als System bereits bekannt**

- Erzielte höchsten Bewerbungserfolg
- Templates waren für Business direkt geeignet

Wiki

UNCLASSIFIED

Email vs. Wiki Collaboration (U)



● Lösung für das Problem „Dokumentenversionen“

Unsere Entscheidung für MediaWiki

- **Dieselbe Software die Wikipedia antreibt**
- **Vertrauen in die aktive Weiterentwicklung**
- **Freie Software**

- **Grundidee, dass wir nur eine Wiki für alles brauchen**
 - Transparenz
 - Community-Gedanke leben, Hierarchien aufbrechen

Agenda

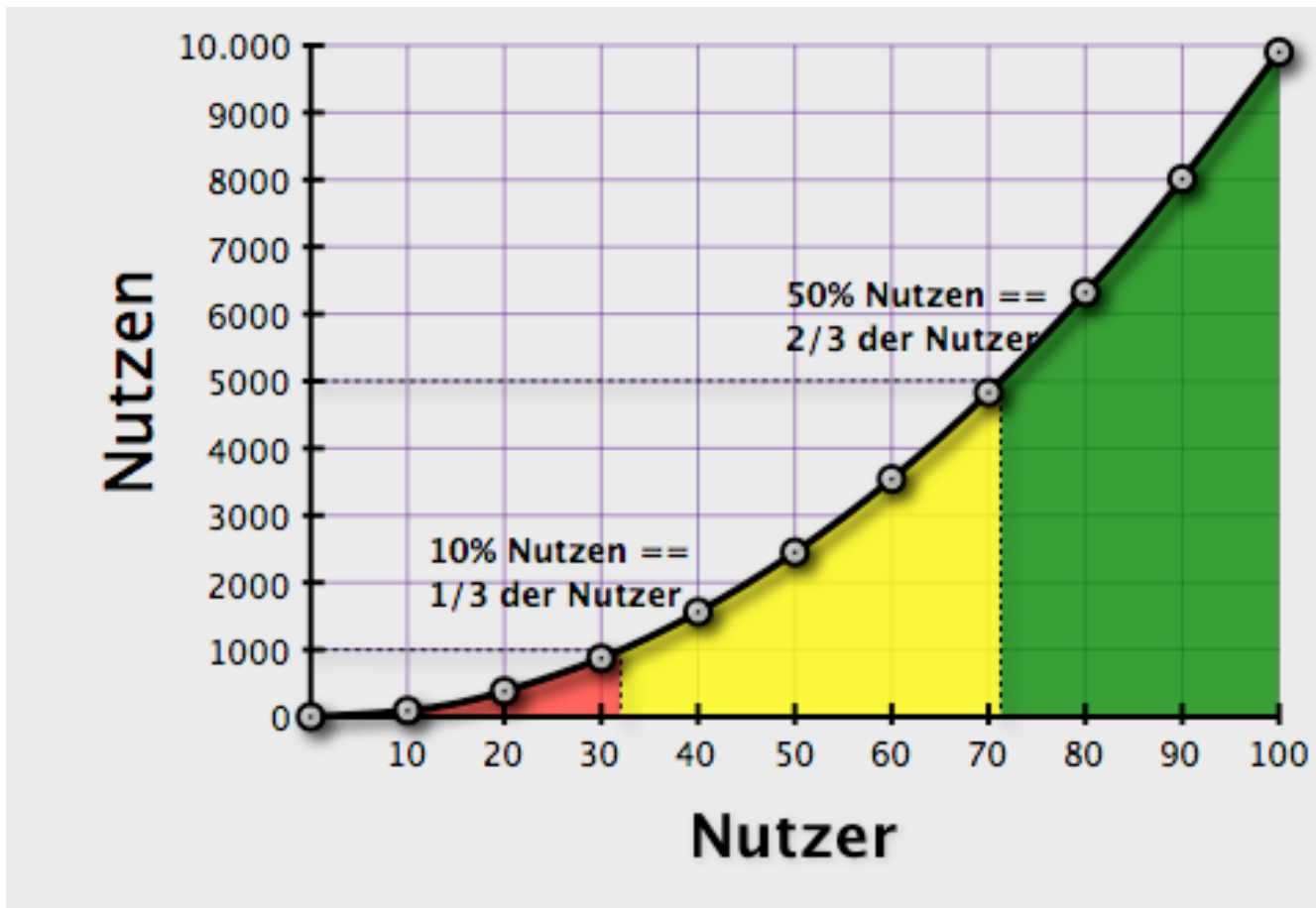
- Wiki und Weblog im Unternehmenseinsatz
- **Problem nach Einführung**
- **Vorzüge von Confluence im Unternehmenseinsatz**
- **Migration bestehender Inhalte**
- **Erreichte Ziele**

Das Problem der Einführung

- Soziale Applikationen sind schnell installiert
- Der Administrationsaufwand ist gering
- Im Unternehmen tritt schnell folgendes Problem auf:

Die Mitarbeiter benutzen die Werkzeuge nicht!

Metcalfe: „Nutzen steigt quadratisch“



Hemmnisse

- „Ich habe keine Ahnung, wie das funktioniert“
- „Ich weiß nicht, wie ich das schreiben soll“
- „Der Eintrag ist nicht von mir, da geh ich nicht dran“
- „Ich weiss nicht, ob meine Themen jemanden interessieren“
- „Akzeptiert jeder, dass ich dafür Zeit aufwende?“
- „Das ist nur schon wieder ein neues Werkzeug“
- „Das soll niemand lesen bevor es nicht reif ist“
- „Der Kunde sollte bei Projekt-Themen direkt mitlesen können“

Hemmnisse aus dem Weg räumen

- Wenn technischer Natur: **mit Schulung, FAQ, Tutorials**
- Wenn Frage des Verhaltens: **mit Richtlinie (Policy)**
- **Auch Wahl der Software spielt eine wichtige Rolle**
- **Community-Tools funktionieren oft nicht gut im Unternehmen**
- **Community-Gedanke wünschenswert, aber auch schwierigste Form der Verhaltensänderung**

Agenda

- Wiki und Weblog im Unternehmenseinsatz
- Problem nach Einführung
- **Vorzüge von Confluence im Unternehmenseinsatz**
- Migration bestehender Inhalte
- Erreichte Ziele

Gründe für Confluence (1/3)

- **Single Sign On**
 - Active Directory mitbenutzen
- **Wiki und Weblog in einem**
 - Nahtlose Integration, ein Tool, ein Editor
- **WYSIWYG Editor**
 - Für viele war Markup kein guter Einstieg
- **Volltextsuche**
 - Über alle Inhalte

Gründe für Confluence (2/3)

- **Seitenhierarchie (Kapitel, Unterkapitel, Eltern/Kind-Seiten)**
 - Basis für den Export als Dokument (Word, PDF...)
- **Anhänge**
 - Beliebige Anhangstypen, Teil der Volltextsuche, Versionierung
 - Wichtige Typen von Anhängen werden direkt gerendert
- **Plugin-Konzept und Benutzermakros**

Gründe für Confluence (3/3)

- **Diskussionsplattform inklusive RSS-Feeds & Email-Benachrichtigungen**
 - Kommentare und Feedback (Faktor „Anerkennung“)
 - Überwachung von Änderungen an Inhalten
- **Enterprise-Wiki:**
 - Berechtigungen für Inhalte und Bereiche bei Bedarf möglich
 - Multi-Wiki für beliebig viele Bereiche / Spaces in selber Instanz
 - Personal Spaces für angemeldete Benutzer
- **Confluence mehr und mehr Wiki-Standard im Unternehmen**

Agenda

- Wiki und Weblog im Unternehmenseinsatz
- Problem nach Einführung
- Vorzüge von Confluence im Unternehmenseinsatz
- **Migration bestehender Inhalte**
- Erreichte Ziele

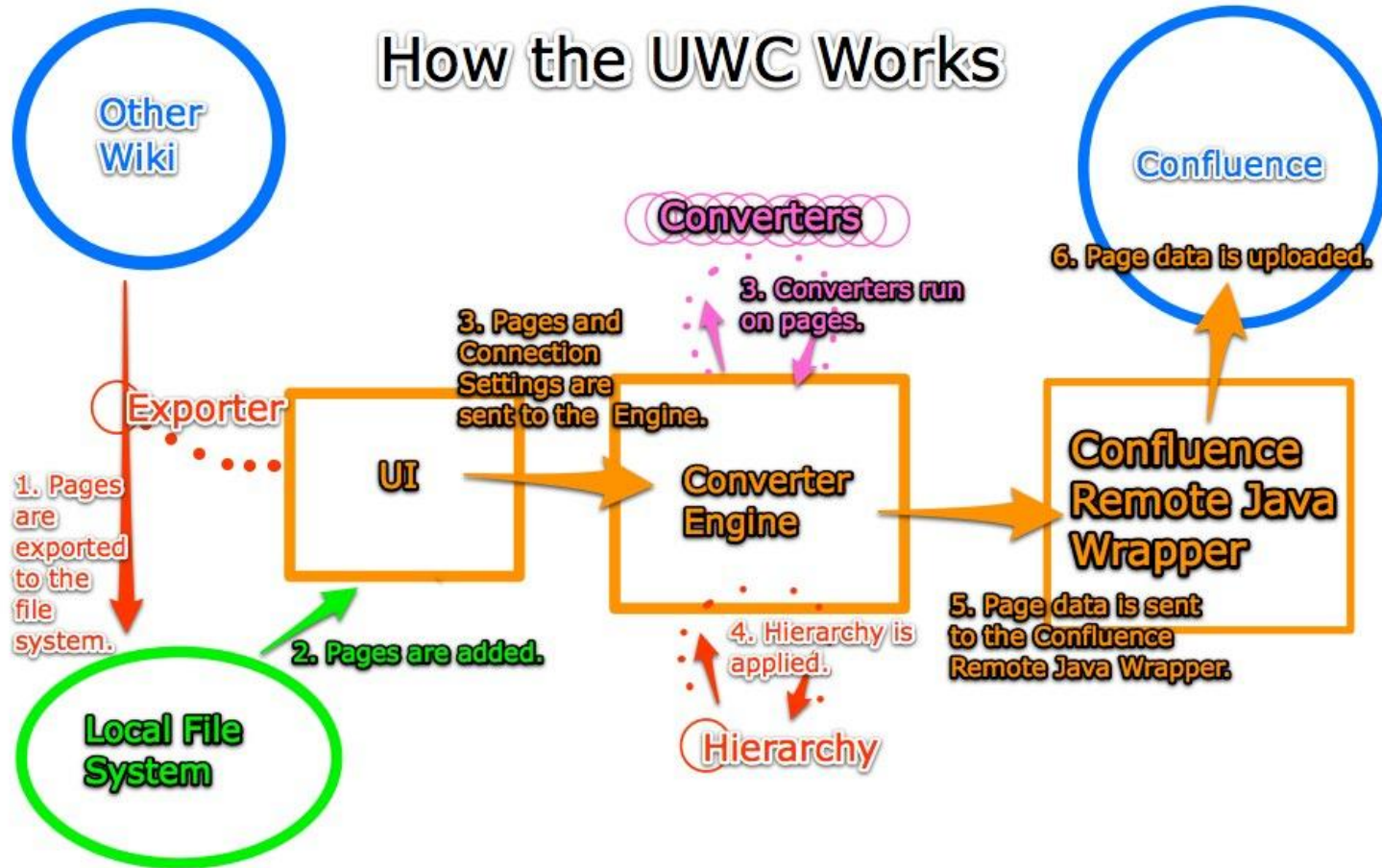
Übernahme von Movable Type

- **Hierfür existierten keinerlei Tools**
- **Extrakt der Metadaten aus der Movable Type DB**
- **Eigene Wandlungsroutinen in Confluence Markup**
- **Hochladen der Seiten via XMLRPC API von Confluence**
- **Bilder analog dazu konvertiert und übernommen**
- **Frage: Wie überführt man mehrere Blogs in Bereiche/Stichworte?
Was machen wir mit Kategorien und Tags?**

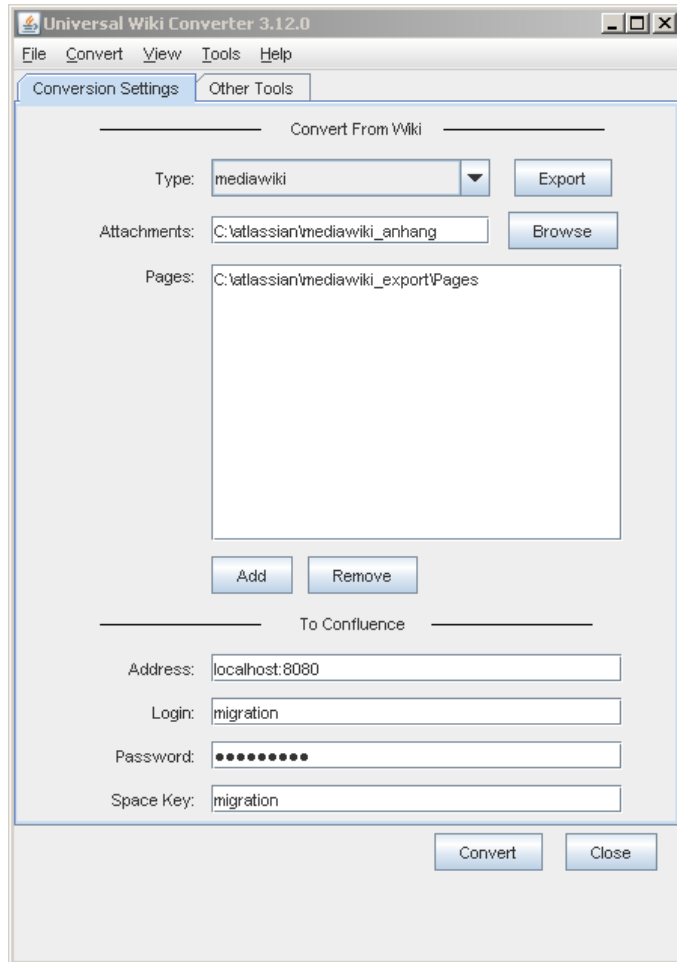
Übernahme von MediaWiki

- **Atlassian bietet hierfür den Universal Wiki Converter (UWC)**
- **Muss vor Einsatz konfiguriert werden**
- **Hat auch dann noch einige Tücken**

Funktionsweise des UWC

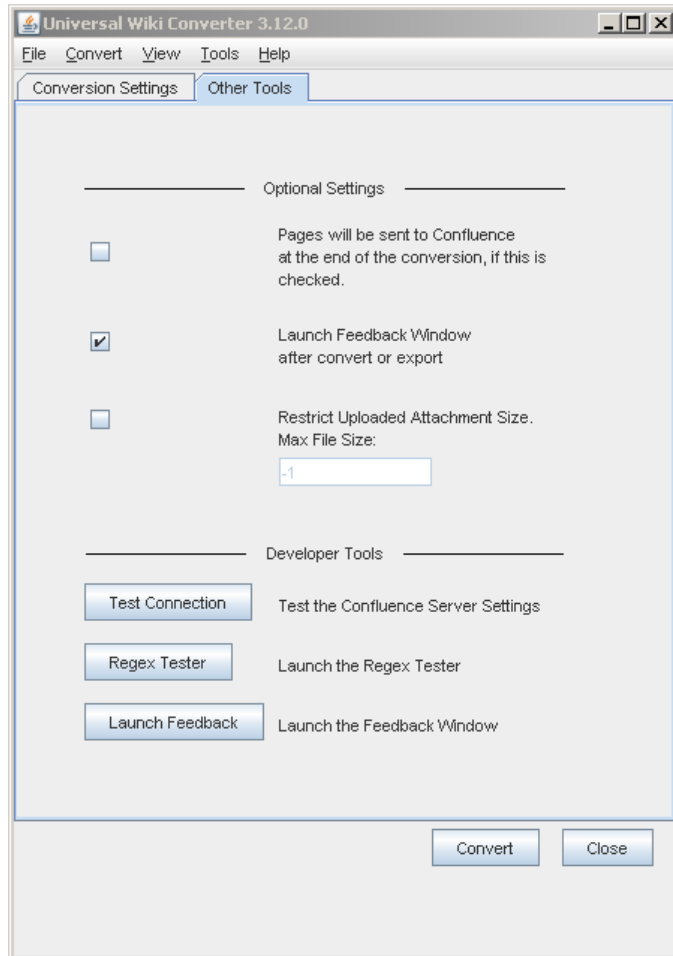


UWC in Aktion (1/2)



- **Typ der Migration auswählen**
- **EXPORT ausführen**
 - Parameter dafür via Config-Datei
- **Export holt keine Attachments ab**
 - Nicht Teil der MediaWiki DB
- **Attachment-Ordner angeben**
- **Page-Extract-Ordner benennen**
- **CONVERT ausführen**

UWC in Aktion (2/2)



- Während der Konfiguration der Konvertierung die Ergebnisse noch nicht an Confluence senden lassen
- Die Konvertierungs-Ergebnisse landen in UWC-Zielordnern
 - Textdateien mit Confluence-Markup

MediaWiki Exporter: Basis-Einstellungen

Important Changes inside conf/exporter.mediawiki.properties

```
# Exporter class
exporter.class=com.atlassian.uwc.exporters.MediaWikiExporter

### Mediawiki database connection information

## database name is the name of the mediawiki database
databaseName=wikidb

## dbUrl is the JDBC connection url. The following is an example mysql url
dbUrl=jdbc:mysql://localhost:3306

# The JDBC driver you wish to use
# Note: You will have to provide the JAR, unless you use MySQL. See:
# http://confluence.atlassian.com/display/CONFEXT/UWC+F.A.Q.#UWCF.A.Q.-HowdoIaddalibraryjartotheUWC%3F
# The Mysql driver has been provided. The following would be the class if you use MySQL.
jdbc.driver.class=com.mysql.jdbc.Driver

## Login info to connect to this database. You will need to replace this.
#login=mylogin
login=mwusr
#password=mypassword
password=mwpwd

## This is the output directory. The export will send text files with Mediawiki syntax to this directory.
#output=/Users/laura/Desktop/
output=c:/atlassian/uwc/
```

MediaWiki Exporter: Metadaten

exporter.mediawiki.properties

```
...  
## User and Timestamp data properties (udmf)  
## Set this to true if you would like user and time  
## preserved. Data will be added to the beginning of  
## {user:foobar}  
## {timestamp:yyyymmddhhmmss}  
udmf=true  
  
## output the original title to page content as an {orig-title:foo bar} macro  
## Useful for hierarchy data that's maintained in the page title like that  
## maintained with SubPageList3 plugin  
origtitle=true  
  
...
```

Metadaten
User / DateTime

Ergebnis des Exports

Name ^	Größe	Typ
1000_Namen_-_ein_und_die_selbe_Ki...	1 KB	Textdokument
5250_Emulation.txt	1 KB	Textdokument
A_Comprehensive_Introduction_to_S...	1 KB	Textdokument
Active_Tracing.txt		
Active_Tracing_App_Feature-Ideen.txt		
Active_Tracing_App_Mockup.txt		
Active_Tracing_iPhone_App.txt		
Active_Tracing_iPhone_App_Worksho...		
Active_Tracing_XML_Service.txt		
Administratorkennwort.txt		
ADO.NET_Datenbankprogrammierung...		
Adressbereinigung.txt		
Ajax.txt		
Ajax__Ein_Übersichtsvortrag.txt		
Aktivierungsgruppen.txt		
Alarmanlage.txt		
Allgemein.txt		
Allgemeines.txt		
Andere_Formate.txt		
Android.txt		

Extrahiertes MediaWiki Markup

```

1 {user:B.Steven}
2 {timestamp:20091005161556}
3 Über XML steht im Internet (z.B. Wikipedia) schon so viel
4 geschrieben, dass ich es hier nicht noch einmal auflisten möchte.
5 XML ist die Abkürzung für E''x''tensible ''M''arkup
6 ''L''anguage und ist eine Auszeichnungssprache für den
7 Datenaustausch zwischen Computersystemen.
8
9 Im Gegensatz zu [[Edifact]] ist diese Sprache nicht an feste
10 Formen gebunden, die Austauschformate können individuell
11 definiert werden.
12 XML ist - im Gegensatz zu [[Edifact]] - visuell gut lesbar, da
13 es nur aus Zeichen und nicht aus verschlüsselten Codes besteht.
14 Die Spezifikationen werden im [[DTD]] definiert und die [[Tag]]-
15 Bezeichnungen können individuell benannt werden.
16
17 Mehr über XML ist unter anderem unter diesem Link zu finden:
18 (http://xml.coverpages.org/xml.html#smlSoftware)

```

Anpassungen an der Konvertierung

- **Konvertierung MediaWiki-Markup:**

- `conf/converter.mediawiki.properties`

- **Enthält Kette von Konvertierern**

- Keys der Property-Datei werden alphabetisch abgearbeitet
- Keys darum geschrieben als **Mediawiki.0001.*** etc.
- Funktion des Keys aus Endung ablesbar:
 - `*.property` Setzen einer Globalen Property
 - `*.java-regex` Regulärer Ausdruck, Match/Replace
 - `*.class` Angabe einer eigenen Klasse für die Konvertierung

XML Converter ausschalten

- Converter macht Annahmen darüber, wann Markup XML/HTML ist
- Er erkennt diese am Auftreten von `<` oder `>` im Markup
- Er kommt dadurch bei Markup wie `< 10 Tage` ins Schleudern

XML Converter ausschalten

```
...  
## HTML: Parse the xml document  
## Mediawiki.1590.xmlconverter.class=com.atlassian.uwc.converters.xml.XmlConverter  
...
```

Bold-Text in Listen erlauben

- In Confluence * sowohl für Fettschrift als auch für Listen
 - Standardsetup: erst Fettschrift-, dann Listen-Konvertierung
 - Effekt: Fehler bei Wandlung von Kombinationen von beidem
- Einfache Anpassung: Ausführungsreihenfolge ändern

Alte Ausführungsreihenfolge

```
...  
Mediawiki.0300-re_bold_italics.java-regex='{5}(.*?)'{5}{replace-with}*_{1}_*  
Mediawiki.0330-multiline-edgel.java-regex=''+\r?\n(=+. *?=\r?\n)''+{replace-with}$1  
Mediawiki.0340-re_multiline.class=com.atlassian.uwc.converters.mediawiki.MultilineBasicConverter  
Mediawiki.0350-re_bold.java-regex='{3}(.*?)'{3}{replace-with}*_{1}*  
Mediawiki.0350-re_italics.java-regex='{2}(.*?)'{2}{replace-with}_{1}_  
...
```

Fix: von 03xx nach 33xx verschieben

Deutsche MediaWiki Namespaces nutzen

- In deutscher MediaWiki `[[Bild:...]]`
statt `[[Image:...]]`
- Anpassung vor eigentlichem ImageConverter:

Deutsche BILD-Inhalte zulassen

```
...  
Mediawiki.0370-aaahjs.java-regex=(?i)\[\[Bild:([\^\\]+)\]\]\{replace-with}\[\[Image:$1]\]  
Mediawiki.0370-re_images.class=com.atlassian.uwc.converters.mediawiki.ImageConverter  
...
```

RegExr - Mozilla Firefox

RegExr

http://gskinner.com/RegExr/

Match Replace

`(?)\\[[Bild:([^\]]+)\]]`

global ignoreCase extended dotall multiline

Share Link

`[[Image:$1]]`

Das ist ein Beispiel für ein `[[Bild:MeinBild.jpg]]` das wir umsetzen wollen.
Und zwar auch für:
`[[Bild:MeinBild2.jpg]]` ,
`[[Bild:MeinBild3.jpg]]`
und was immer da sonst noch so kommt an Bildern.

Das ist ein Beispiel für ein `[[Image:MeinBild.jpg]]` das wir umsetzen wollen.
Und zwar auch für:
`[[Image:MeinBild2.jpg]]` ,
`[[Image:MeinBild3.jpg]]`
und was immer da sonst noch so kommt an Bildern.

RegExp: `/(?)\\[[Bild:([^\]]+)\]]/g`
pattern: `(?)\\[[Bild:([^\]]+)\]]`
flags: `g`
2 capturing groups:
group 1: `(?)`
group 2: `[[^\]]+`

0.3.1b built by gskinner.com

About R <http://gskinner.com/RegExr/>

Samples: show all

- .
- \w
- \W
- \d
- \D
- \s
- \S
- [ABC]
- [^ABC]

Matches any character, except for line breaks if dotall is false.

Autorennamen umsetzen (1/3)

```
1 {user:A.Adler}
2 {timestamp:20091005161556}
3 Über XML steht im Internet (z.B. Wikipedia) schon so viel
  geschrieben, dass ich es hier nicht noch einmal auflisten möchte.
```

- **Oft nötig bei Einsatz eines Active Directory (AD)**
- **Autorennamen decken sich nicht mit Namen im AD**
- **Property-Datei aufbauen für die Umsetzung**

author-mapping.properties

```
A.Adler=aad
G.Hetzner=ghe
W.Wulff=wwu
D.Dolm=ddo
...
```

Autorennamen umsetzen (2/3)

```
com.iks_gmbh.uwc.converters.UsernameConverter.java

package com.iks_gmbh.uwc.converters;

import java.io.FileInputStream;
import java.io.IOException;
import java.util.Properties;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

import com.atlassian.uwc.converters.BaseConverter;
import com.atlassian.uwc.ui.Page;

public class UsernameConverter extends BaseConverter {

    public static String authorMappingFile = "author-mapping.properties";
    private static Properties authorMapping;

    public UsernameConverter() {
        super();
        if (authorMapping == null) {
            authorMapping = new Properties();
            FileInputStream fis;
            try {
                fis = new FileInputStream(authorMappingFile);
                authorMapping.load(fis);
                fis.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
}
```

Autorennamen umsetzen (3/3)

```
@Override
public void convert(Page page) {
    String input = page.getOriginalText();
    String converted = convert(input);
    page.setConvertedText(converted);
}

public String convert(String sSource) {
    Pattern pattern = Pattern.compile("\\{user:(.*)\\}");
    Matcher matcher = pattern.matcher(sSource);
    StringBuffer sb = new StringBuffer();
    while (matcher.find()) {
        String user = matcher.group(1);
        String newUser = (String) authorMapping.get(user);
        matcher.appendReplacement(sb, "{user:" + (newUser == null ? user : newUser) +
    }
    matcher.appendTail(sb);
    return sb.toString();
}
```

Usernamen an iks LDAP anpassen

```
...
Mediawiki.0004.aaahjs-usernames.class=com.iks_gmbh.uwc.converters.UsernameConverter
...
```

User/Date-Konvertierung

UDMF Konvertierung aktivieren, auch unbekannte Nutzer zulassen

```
...  
Mediawiki.0004.userdate.class=com.atlassian.uwc.converters.mediawiki.UserDateConverter  
Mediawiki.0004.users-must-exist.property=false  
...
```

MediaWiki-Kategorien in Label verwandeln

- Hierfür gibt es einen mitgelieferten Converter
- Dieser kommt jedoch nicht mit `Kategorie:` als Namespace klar
- Derselbe Trick wie bei Bildern/Images:

Kategorien in Labels verwandeln

```
...
Mediawiki.0400-aaahjs.java-regex=(?i)\[\[Kategorie:([^\]]+)\]\]\]{replace-with}[[Category:$1]]
Mediawiki.0400-categories2labels.class=com.atlassian.uwc.converters.mediawiki.CategoryConverter
...
```

In Labels verwandelte Kategorien aus Quelle entfernen

```
...
Mediawiki.0401-re_links_no_categories.java-regex=((?i)\[\[(category:[^\]]+)\]\]\s*)+{replace-with}
...
```

Links auf Kategorien in Links auf Label wandeln

Kategorie-Links in Label-Links verwandeln

```
...  
Mediawiki.9992-kategorielink.class=com.iks_gmbh.uwc.converters.KategorielinkConverter  
...
```

```
@Override  
public void convert(Page page) {  
    String input = page.getOriginalText();  
    String converted = convert(input);  
    page.setConvertedText(converted);  
}  
  
public String convert(String sSource) {  
    Pattern pattern = Pattern.compile("(?i)Kategorie__([^\s]+)\s");  
    Matcher matcher = pattern.matcher(sSource);  
    StringBuffer sb = new StringBuffer();  
    while (matcher.find()) {  
        String s = matcher.group(1);  
        s = s.replace(" ", "").toLowerCase();  
        s = "///label/" + s + "];";  
        matcher.appendReplacement(sb, Matcher.quoteReplacement(s));  
    }  
    matcher.appendTail(sb);  
    return sb.toString();  
}
```

➔ Ein nachgelagerter Prozeß wandelt `///label/mylabel` in `---label-mylabel`. Dieser Prozeß ist lustigerweise nicht Teil der konfigurierbaren Konvertierungskette. Wir werden die Schuldigen suchen, ermitteln und eliminieren.

Absolute URL-Referenzen auflösen

- UWC wandelt MEDIA-Elemente korrekt in Attachments
- Problem wenn Markup diese mit absoluter URL anspricht
- Analoges Problem: Wenn MediaWiki absolut auf MediaWiki zeigt

Absolute URL-Referenzen aus iks-Wissen in Attachments umwandeln

```

...
Mediawiki.0030-aaahjs-ikswiki.java-regex=(?i)\\[\\http://www.iksblogs.de/wiki/ikswikiuploads[^\\]*/([^\\]/)*\\]\\{replace-with}[[Media:$1]]
Mediawiki.0030-aabhjs-ikswiki_a1.java-regex=(?i)\\[\\[Media:[^\\]]*\\]\\%C3\\%84([^\\]]*\\]\\}\\{replace-with}$1Ä$2
Mediawiki.0030-aabhjs-ikswiki_a2.java-regex=(?i)\\[\\[Media:[^\\]]*\\]\\%C3\\%A4([^\\]]*\\]\\}\\{replace-with}$1ä$2
Mediawiki.0030-aabhjs-ikswiki_o1.java-regex=(?i)\\[\\[Media:[^\\]]*\\]\\%C3\\%96([^\\]]*\\]\\}\\{replace-with}$1Ö$2
Mediawiki.0030-aabhjs-ikswiki_o2.java-regex=(?i)\\[\\[Media:[^\\]]*\\]\\%C3\\%B6([^\\]]*\\]\\}\\{replace-with}$1ö$2
Mediawiki.0030-aabhjs-ikswiki_u1.java-regex=(?i)\\[\\[Media:[^\\]]*\\]\\%C3\\%9C([^\\]]*\\]\\}\\{replace-with}$1Ü$2
Mediawiki.0030-aabhjs-ikswiki_u2.java-regex=(?i)\\[\\[Media:[^\\]]*\\]\\%C3\\%BC([^\\]]*\\]\\}\\{replace-with}$1ü$2
Mediawiki.0030-aabhjs-ikswiki_s1.java-regex=(?i)\\[\\[Media:[^\\]]*\\]\\%C3\\%9F([^\\]]*\\]\\}\\{replace-with}$1ß$2
...

```

Absolute Mediawiki URLs erkennen und umsetzen

```

...
Mediawiki.9991-iksinternalurl.class=com.iks_gmbh.uwc.converters.IksInternalURLConverter
...

```

Macken im Quellcode des UWC

- **Das UWC Binary und seine Konfigurationsdateien sollten eigentlich zur Anpassung der Konvertierung reichen**
- **Zum Zeitpunkt unserer Konvertierung gab es noch Probleme, die nur im Quellcode repariert werden konnten**
- **Der UWC Quellcode ist ständig im Wandel und kann via Subversion heruntergeladen werden**

com.atlassian.uwc.exporters. MediaWikiExporter

- **Export von Pages fehlerhaft**
 - Teilweise alte Revisionen exportiert
- **Notbremse: immer und nur „latest“ als Revision ziehen**
- **Seit Mai 2011 hier große Änderungen im Code**

```
Workspace file: MediaWikiExporter.java

        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
if (gettingUserdate()) { //date for udmf framework: usernames
    if (!this.running) return null;
    // String userdate = getUserDateData(rev, revUdmfMap);
    String userdate = getUserDateData(latest, revUdmfMap);
    text = userdate + text;
}

//save the data into a local object
MediaWikiPage mwpage = new MediaWikiPage(title, text, namespace);
pages.add(mwpage);
//next: 1) handle URL decoding when converting, 2) handle other
//next: refactor you can use the jdbc URL to set the UTF-8 encoding
}
}
} catch (SQLException e) {
    log.error("Problem while examining data.");
    e.printStackTrace();
}
return pages;
}

protected String getNamespaceWhereClause() {
```

com.atlassian.uwc.converters.mediawiki. UserDataConverter

- Das Erkennungs-pattern für User/Date-Metadaten lief nicht unter Windows
- CR LF vs LF
- Seit Mai 2011 scheint das repariert zu sein

```
Repository file: UserDataConverter.java

    log.debug("page.getAuthor: " + page.getAuthor());
    page.setConvertedText(converted);
    log.debug("Adding User and Date metadata - Complete");
}

Pattern username = Pattern.compile("^\\{user:([^\n]*)}\\}\\n");
protected String getUser(String input) {
    Matcher userFinder = username.matcher(input);
    if (userFinder.find()) {
        return userFinder.group(1);
    }
    return null;
}

Pattern date = Pattern.compile("(?:\n|^)\\{timestamp:([^\n]*)}\\}\\n");
protected Date getDate(String input) {
    Matcher dateFinder = date.matcher(input);
    if (dateFinder.find()) {
        String timestamp = dateFinder.group(1);
        DateFormat dateFormat = new SimpleDateFormat("yyyyMMddHHmmss");
        try {
            return dateFormat.parse(timestamp);
        } catch (ParseException e) {
            log.error("Couldn't format date: " + timestamp);
            e.printStackTrace();
            return null;
        }
    }
}
```

com.atlassian.uwc.converters.mediawiki. TableConverter

- URL-Hyperlinks in Tabellen werden fälschlicherweise als HTML-Attribute erkannt und gefiltert
- Notbremse mit „false &&“ ziehen und so die Attributererkennung ganz auslassen

```
Workspace file: TableConverter.java
Pattern itemEnd = Pattern.compile(" " +
    "\\|\\|s$");
/**
 * adds Confluence syntax for one data cell to the growing replacement string
 * @param cellContents string that will be added as contents of new cell
 * @param cToken current cell delimiter, either CELL_DELIM or HEADER_DELIM
 * @param replacement current replacement string to be added to
 * @return new replacement string, with new cell added
 */
private String addDataSyntax(String cellContents, String cToken, String replacement) {
    //strip out attributes (styles, etc.) as Confluence can't use them
    if (false && isAttribute(cellContents)) // BIG BUG, FILTERS OUT URLs (HJS)
        return replacement;

    //get rid of extraneous whitespace
    cellContents = cellContents.trim();

    //prepare for handling list items
    if (isListItem(cellContents)) {
        cellContents = getListContents(cellContents);
    }

    //add the data cell
    if (cToken.endsWith("|") cToken += " "; //don't add the space for list deli
    if ("".equals(cellContents))
        replacement += cToken;
```

Anpassungen für den Confluence Upload

- **XMLRPC API muss offen sein**
- **Für User/Date Metadaten muss UDMF Framework aktiviert werden**
- **Dafür gibt es das sogenannte UDMF Confluence Plugin**
- **Einspielen erfolgt durch kopieren des JARs in den Classpath**

Aufwand unserer Wiki-Migration

- **Dokumente: 1000 Wikiseiten, 300 Attachments**
- **Dauer pro Lauf: 3 Minuten, mit Upload Confluence eher 15 Minuten**
- **Cycles: um die 50, davon 20% inklusive Upload**
 - Problem: Ziel-Space löschen, Berechtigungen neu aufsetzen
- **Coding und Einarbeitung ungefähr 3-4 MT**
- **Ergebnis**
 - Ordentliche Konvertierung
 - Kein Content-Verlust
 - Ca. 1% relevante Markupfehler nach allen Anpassungen

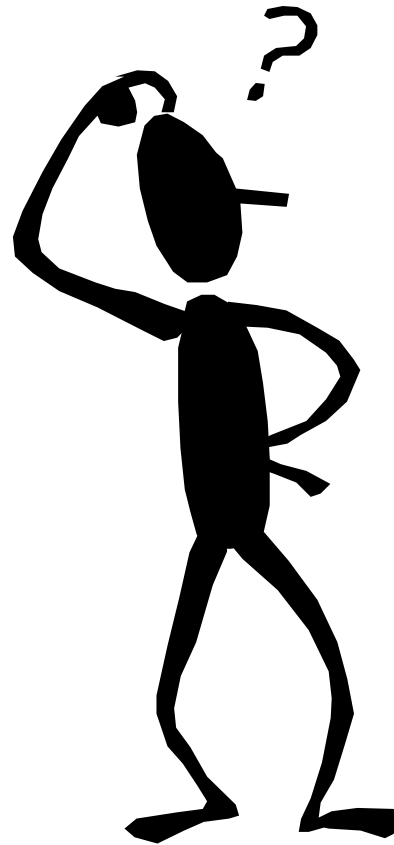
Agenda

- Wiki und Weblog im Unternehmenseinsatz
- Problem nach Einführung
- Vorzüge von Confluence im Unternehmenseinsatz
- Migration bestehender Inhalte
- **Erreichte Ziele**

Lohnt sich der Aufwand? -- Erreichte Ziele

- **Deutlich höhere Teilnahme**
- **Neue Nutzerkreise erschlossen (WYSIWYG & Berechtigungen)**
- **Diskussionen finden häufig im Tool statt**
- **Multi-Wiki mit Berechtigungen lässt Hierarchie einhalten falls nötig**
- **Benutzbare Suchergebnisse, vernünftiges Ranking**
- **Eltern/Kind-Hierarchie ideal für Dokumente**
- **Es ist leichter Ansprechpartner zu Themen zu finden seit sich jeder selbst darstellen darf und kann**
- **Einzigste Probleme:**
 - Microblogging: Noch in Yammer (Confluence nicht stark genug)
 - Personenverzeichnis leider immer komplett sichtbar

Fragen?



www.iks-gmbh.com