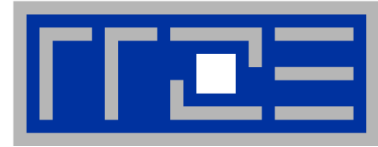


Das Web-Frontend WAID

und andere Entwicklungen im
Rahmen des Erlanger IdM-Projekts





- **Web-Frontend WAID**
 - **Web Administration for IDMone / Identity Management**

- **und andere Entwicklungen**
 - **Matching – Personen zuordnen**
 - **Integration in Novell's Identity Manager**
 - **First-Aid IdM-Toolkit**
 - **jpwgen**
 - **jidgen**
 - **idmsec**
 - **m.a.r.v.**



WAID – Web Administration for IDMone





- **Ursprüngliche Web-Oberfläche:**
 - **Novell's User Application**

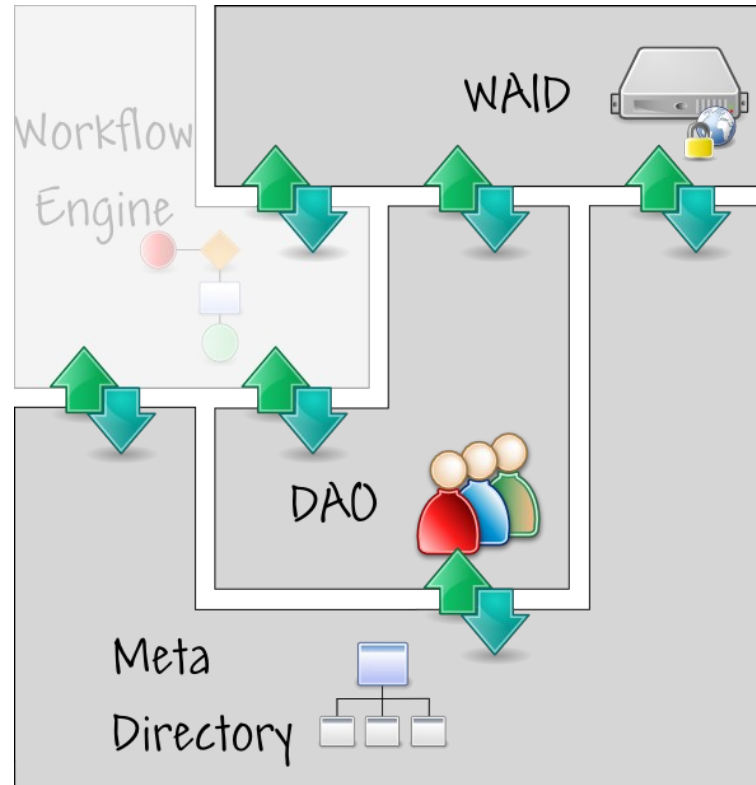
- **Blog- Eintrag vom 11.März 2008 (Peter Rygus)**
 - [...] Zum Schluss muss leider noch angemerkt werden, dass IDMone mit seinen Anforderungen die Fähigkeiten der User Application derartig überfordert, dass beschlossen wurde, hier eine Eigenentwicklung einzusetzen. Das wirft zwar den Zeitplan von IDMone deutlich zurück, verspricht aber langfristig eine bessere, flexiblere, barrierefreie Weboberfläche, die wir besser pflegen können.
[...]

- **Hauptgründe**
 - **Anpassungen meist sehr zeitintensiv**
 - **Ergebnis trotzdem nur ein Kompromis**

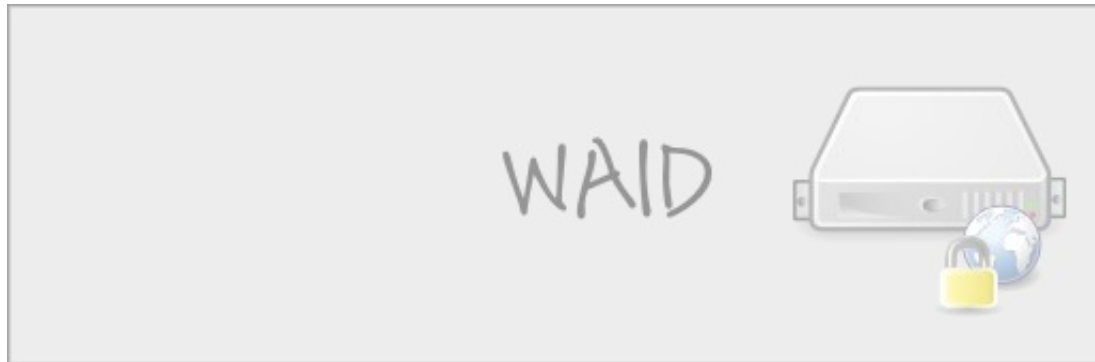


- **das Produkt hinter dem Identity Management (IdM) Self Service der FAU** <https://www.idm.uni-erlangen.de/>
- **basiert auf populäre Java-Frameworks**
 - **Tapestry als Web-Framework**
 - **Spring-Framework inkl.**
 - **Spring Security**
 - **Spring LDAP**
 - **weitere 3rd-Party Erweiterungen**
- **basiert auf dem Web-Baukasten der FAU** <http://www.vorlagen.uni-erlangen.de/>
 - **barrierefreie Webseiten**
 - **Corporate Design der FAU**
 - **strikte Trennung zwischen Inhalt und optischer Gestaltung**

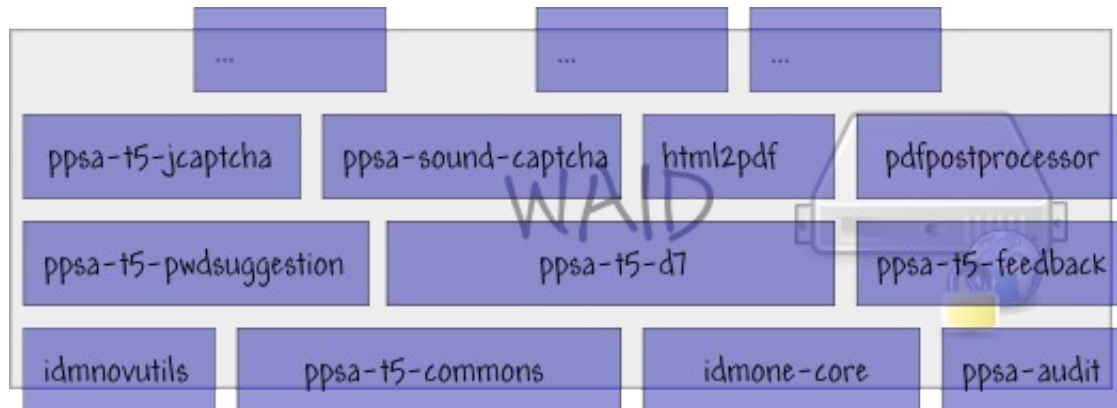
- **WAID in seiner Umgebung**



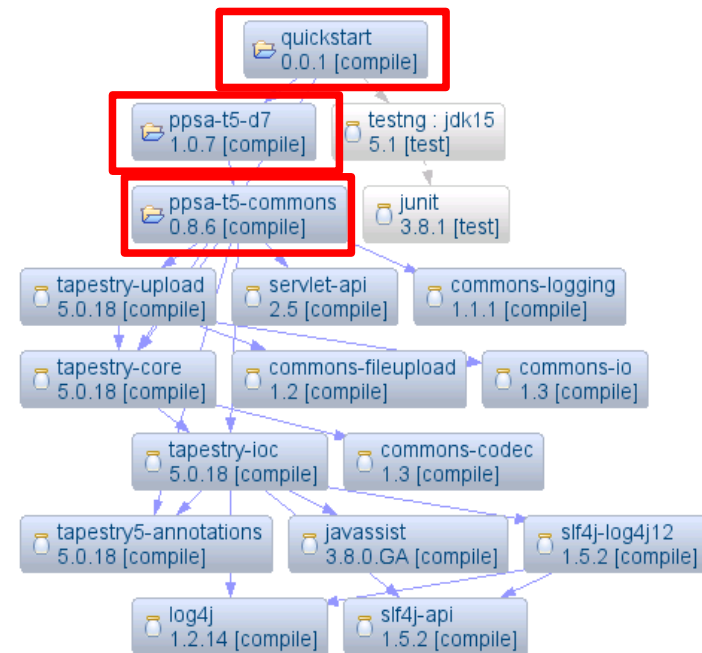
- **Wiederverwendbare Module**
 - z.B. Web-Baukasten für Tapestry
 - Grundsystem wird bereits für weitere Anwendungen eingesetzt



- **Wiederverwendbare Module**
 - z.B. Web-Baukasten für Tapestry
 - Grundsystem wird bereits für weitere Anwendungen eingesetzt



- **Quickstart in 2 Minuten**
 - inkl. Corporate Design
 - inkl. dynamisches Menu
 - Eclipse Demo
- **Maven-Archetyt**
 - **Abhängigkeitsbaum**





- **komplett lokalisiertes Web-Frontend**
 - akt. Deutsch und Englisch
 - beliebig erweiterbar
- **Barrierefreie Webseiten**
- **Corporate Design**
- **keine Kompromisse**
- **reduzierter Zeitaufwand für Erweiterungen**



- **Features**
 - **Datenschutzselbstauskunft**
 - **Dienstleistung(en) anzeigen**
 - **Zugehörigkeiten anzeigen**
 - **Passwort ändern**
 - **Sicherheitsfragen (Challenge & Response)**
 - **Feedbackseite**
 - **PDF Druck**
 - **Benutzer Info Brief**
 - **Dienstleistungs Info Brief**
 - **Funktionen für Administratoren**
 - **Suche nach Benutzern**
 - **(Initial-)Passwort setzen**
 - **Benutzer Info Brief drucken**
 - **Zugehörigkeitsanzeige**
 - **Dienstleistungsanzeige**
 - **...**



- **Komponenten sollen veröffentlicht werden**
 - **Voraussetzung**
 - **Produktreife**
 - **Dokumentation**
 - **Open Source**

- **eigenes IdM-Projekt?**
 - **helfen Sie mit**
 - **oder fordern Sie Unterstützung an**

- **falls Sie Interesse haben**
 - **schreiben Sie uns idmone@rrze.uni-erlangen.de**



Matching – Integration in Novell's Identity Manager





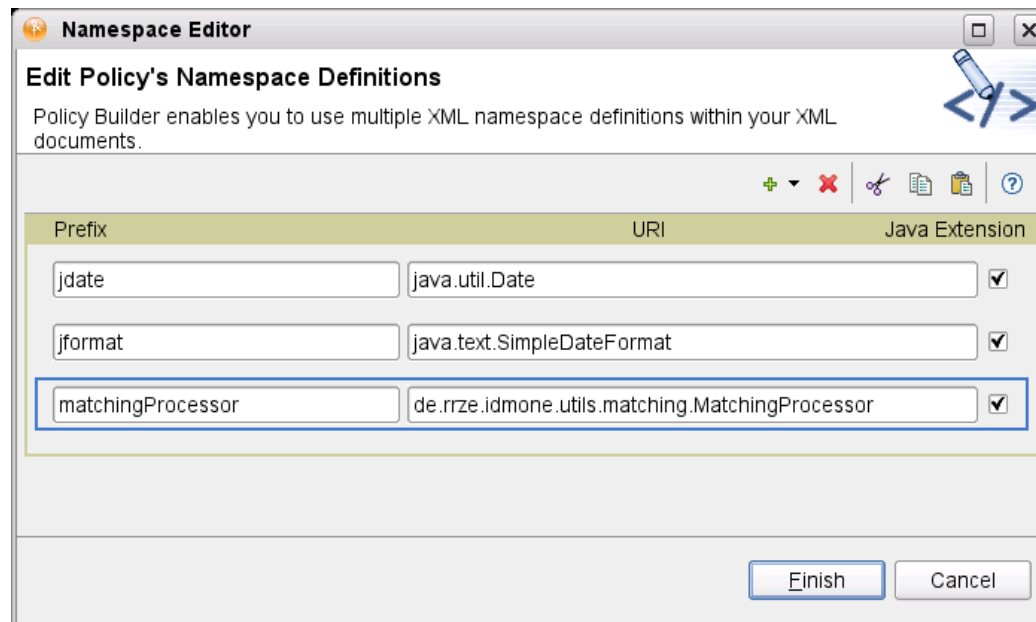
- **Grundlagen des Matchings:**
 - **siehe Vortrag: Data Linkage in IDM Systems**
 - **Rules-Engine (Drools)**
 - **Ähnlichkeitsfunktionen (phonetisch, Pattern Matching, kombiniert)**

- **Ergebnis**
 - **eine Java-Bibliothek**

- **Einfache Verwendung**
 - **Eingabe:**
 - **eine neue Person**
 - **mögliche Kandidaten (=vorhandene Personen)**
 - **Ausgabe:**
 - **bester Treffer inkl. Wahrscheinlichkeit**
 - **zweitbester Treffer inkl. Wahrscheinlichkeit**
 - **...**



- **JAR-Dateien kopieren nach**
 - `/opt/novell/eDirectory/lib/dirxml/classes/`
- **DirXML-Policy: Erweiterung des Namensraums**





- Variablen für Vorname, Nachname, Geburtsdatum und Geburtsort
- Initialisierung des MatchingProcessors mit der „neuen Person“
- Hinzufügen bel. vieler möglicher Kandidaten

```
Actions
✓ ⚡ set local variable("base", scope="policy", Parse DN("ldap", "dest-dn", Global Configuration Value("PEOPLE_CONTAINER")))
✓ ⚡ set local variable("givenname", scope="policy", Operation Attribute("Given Name"))
✓ ⚡ set local variable("surname", scope="policy", Operation Attribute("Surname"))
✓ ⚡ set local variable("dateofbirth", scope="policy", Operation Attribute("schacDateOfBirth"))
✓ ⚡ set local variable("placeofbirth", scope="policy", Operation Attribute("schacPlaceOfBirth"))
✓ ⚡ set local variable("matchingProcessor", scope="policy", object(XPath("matchingProcessor:new($givenname,$surname,$dateofbirth,$placeofbirth)"))
✓ ⚡ set local variable("candidates", scope="driver", nodeset(Query(class name="User", max-result-count="200", dn(Parse DN("ldap", "dest-dn", Global Co
—"PEOPLE_CONTAINER")), match("Given Name", Substring(length="2", Local Variable("givenname")+""), match("Surname", Substring(length="1"
—""), "Given Name", "Surname", "schacDateOfBirth", "schacPlaceOfBirth)))
✓ ⚡ if
    if local variable 'candidates' not equal ""
then
    set local variable("match result", scope="policy", object(XPath("matchingProcessor:addNodeSet($matchingProcessor.$candidates)"))
else
```



- **Auswertung starten**
- **Variablen für ersten DN, ersten Koeffizienten, zweiten DN und zweiten Koeffizienten**
- **Eigentliche Entscheidung via DirXML-Skript**

```
✓ ↗ set local variable("match_result", scope="policy", object(XPath("matchingProcessor:nodeSetMatching($matchingProcessor))))
✓ ↗ set local variable("firstDn", scope="policy", object(XPath("matchingProcessor:getFirstDn($matchingProcessor))))
✓ ↗ set local variable("firstCoeff", scope="policy", object(XPath("matchingProcessor:getFirstCoeff($matchingProcessor))))
✓ ↗ set local variable("secondDn", scope="policy", object(XPath("matchingProcessor:getSecondDn($matchingProcessor))))
✓ ↗ set local variable("secondCoeff", scope="policy", object(XPath("matchingProcessor:getSecondCoeff($matchingProcessor))))
✓ ↗ if
    if local variable 'firstCoeff' greater than "0.93"
    then
        set operation destination DN(dn(Local Variable("firstDn")))
        if
            if local variable 'secondCoeff' greater than "0.80"
            then
                trace message(level="3", "second coeff very very high")
            else
                trace message("good match")
        else
            if
                if local variable 'firstCoeff' greater than "0.70"
                then
                    trace message(level="3", "possible match")
                else
                    trace message("no match")
```



- **Erweiterbares Matching**
 - dank Java
- **volle Integration in Novell's Identity Manager**
 - keine externe LDAP-Verbindung nötig
 - Auswertung erfolgt in DirXML
- **wichtige Parameter jederzeit änderbar**
 - keine Programmierkenntnisse erforderlich!
- **alles in eine Library unterhalb des Driver Set gepackt**
 - von jedem Quellsystem-Treiber aus nutzbar
 - durch simples "Link a policy"

First-Aid IdM-Toolkit

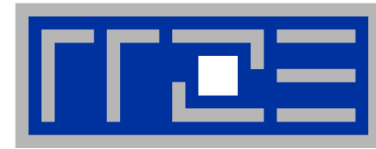
jpwgen

jidgen

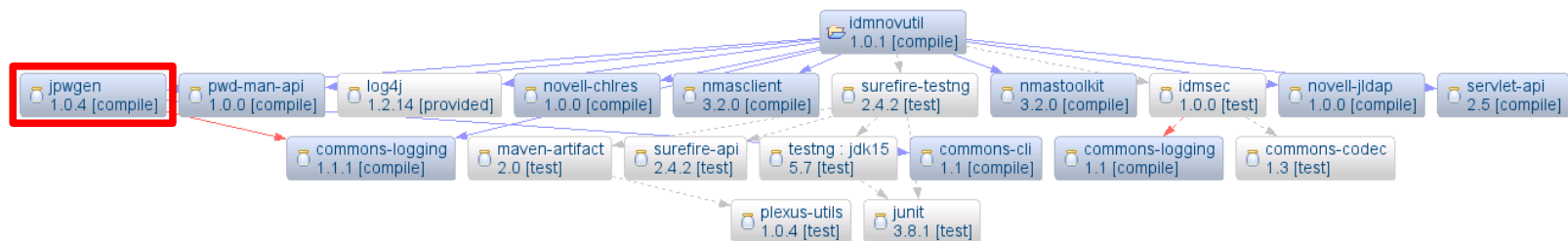
idmsec

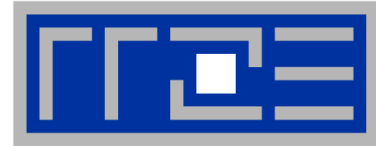
FAQ

MARY.



- <http://jpwgen.berlios.de/>
 - Open Source
- Java-basierter Passwortgenerator
 - Java-Version des bekannten pwgen Programms
- Verwendung
 - via Kommandozeile
 - oder als Bibliothek
- Anwendung in ppsa-t5-pwdsuggestion
 - generiert Passwort Vorschläge in WAID
 - Novell-Passwort-Policy konform!





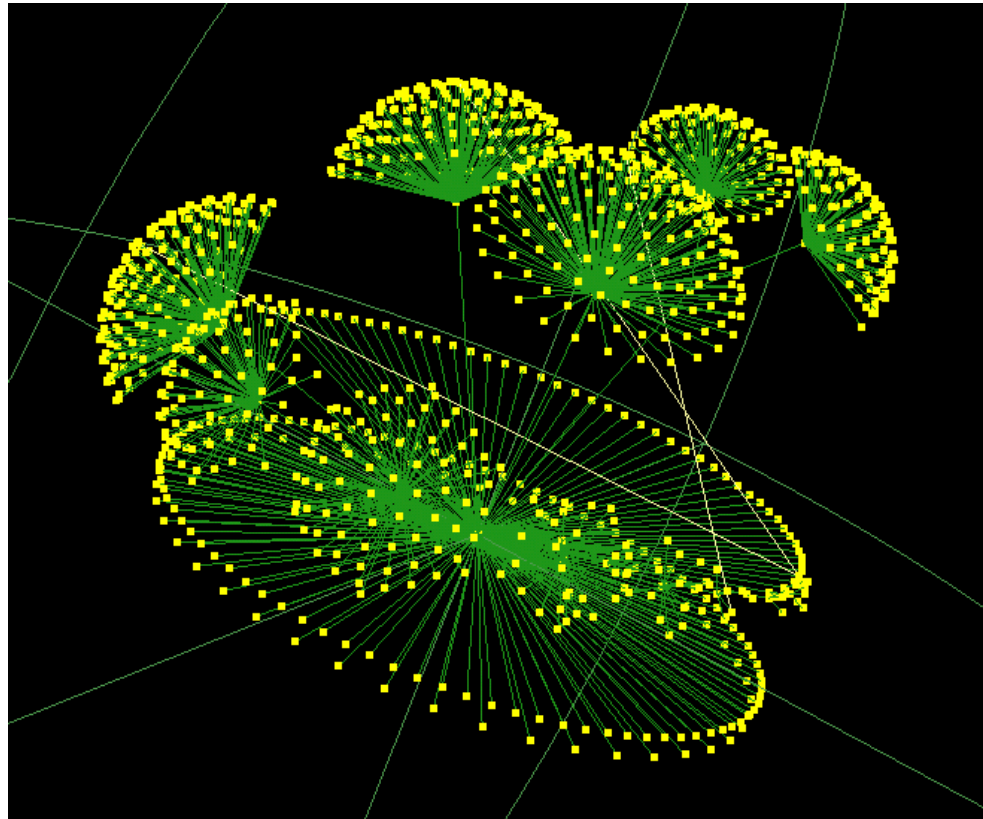
- <http://jidgen.berlios.de/>
 - Open Source
- Java-basierter ID-Generator
- Verwendung
 - via Kommandozeile
 - oder als Bibliothek
- Konfiguration mittel Template-Sprache
- Beispiele
 - `java -jar jidgen.jar -Ts i -Ty 2008 -TI Doe -Tf John -T s:y1:2l:2f`
 - i8dojo
 - `java -jar jidgen.jar -B -Bf blacklist_file -T C+:V+:N2+:C+:V+:C+`
 - no35dax



- **stellt u.a. einen Passwort-Service bereit**
 - also erzeugen von Passwort-Hashes für beliebige Systeme
- **Unterstützte Algorithmen und Kodierungen**
 - MD5, SMD5
 - SHA, SSHA
 - UNIX CRYPT, UNIX SMD5 CRYPT
 - Apache MD5, HIS/SOS Apache MD5
 - OpenBSD-style Blowfish
 - SMB, LANMAN, NTUNICODE
 - APHELION
 - ...
- **Aufruf**

```
✓ ↗ set local variable("pwdService", scope="policy", object(XPath("pwdService:getInstance()")))  
✓ ↗ set destination attribute value("fauSMD5Unix", XPath("pwdService:crypt($pwdService, add-attr[@attr-name='nspmDistributionPassword']/value, 'UMD5')))  
✓ ↗ set destination attribute value("fauSMD5Password", XPath("pwdService:crypt($pwdService, add-attr[@attr-name='nspmDistributionPassword']/value, 'HIS')))  
✓ ↗ set destination attribute value("unixPassword", XPath("pwdService:crypt($pwdService, add-attr[@attr-name='nspmDistributionPassword']/value, 'UNIX_CRYPT')))  
✓ ↗ set destination attribute value("fauSSHAPassword", XPath("pwdService:crypt($pwdService, add-attr[@attr-name='nspmDistributionPassword']/value, 'SSHA')))  
✓ ↗ set destination attribute value("ntPassword", XPath("pwdService:crypt($pwdService, add-attr[@attr-name='nspmDistributionPassword']/value, 'SMB_NTUNICODE')))  
✓ ↗ set destination attribute value("fauBFPassword", XPath("pwdService:crypt($pwdService, add-attr[@attr-name='nspmDistributionPassword']/value, 'BCRYPT')))
```

- **Modular Access Rights Visualization**
 - Visualisierungslösung für LDAP Access Controls
- akt. Stand: einfache OpenLDAP-ACLs





Vielen Dank!

