Entwurfsbeschreibung für das Rollen- und Rechtesystem des LMS OLAT

1 Allgemeines

OLAT ist ein rechtebasiertes System, in dem ein angemeldeter Benutzer nur genau die Funktionen in Anspruch nehmen kann, zu denen er im jeweils gegebenen Kontext berechtigt ist. Ein bestimmter Kontext ergibt sich dabei einerseits aus den intern festgelegten Systemrollen (bzw. Systemgruppen, die beschreiben, wozu ein Benutzer stets berechtigt ist, siehe 2.1) in Abhängigkeit vom Benutzer und dessen Zuordnung zu diesen. Andererseits kann der allein durch Systemrollen implizierte Kontext erweitert werden durch die Zugehörigkeit zu einer oder mehreren Benutzergruppen. Allgemeiner betrachtet orientiert sich dieses Rechte- und Rollenkonzept damit stark am Policy-Konzept von Java (siehe 3.)

2 Produktübersicht

2.1 OLAT-Systemrollen

OLAT (5.1) unterscheidet zunächst zwischen vier hierarchisch gegliederten Systemrollen - Gästen, Benutzern, Autoren und Administratoren, wobei jede dieser Rollen die Rechte für vorher genannte einschließt. Bspw. kann eine am OLAT in der Rolle eines Autors angemeldete Person ebenso an angebotenen Kursen teilnehmen wie ein weniger privilgierter Benutzer, und ein Administrator kann ebenso Kurse erstellen und veröffentlichen wie ein weniger priviligierter Autor.
Über diese vier grundlegenden Systemrollen hinaus existieren zwei weitere Systemgruppen - Gruppenverwalter und Benutzerverwalter. Insofern ein Benutzer oder Autor einer dieser Gruppen angehört, werden dessen Rechte auf bestimmte Funtionalitäten (siehe 2.x,2.y) erweitert. Es sei bemerkt, dass Administratoren diesen erweiterten Systemgruppen von vornherein angehören.
Gäste besitzen als anonyme, nicht registrierte Benutzer ohne eigenes Profil klarerweise nur sehr eingeschränkte Rechte und können daher weder Einstellungen an der Benutzeroberfläche vornehmen, noch angebotene Tests absolvieren oder Forumsbeiträge verfassen. **Gäste werden deshalb nicht Gegenstand der folgenden Betrachtung sein.**
Es folgt eine Auflistung aller Systemrollen und der wichtigsten Funktionen, zu denen diese jeweils berechtigen:

2.1.1 Benutzer

- **Anpassen der Oberfläche und der persönlichen Daten**
  Beschreibung: Ein Benutzer kann Spracheinstellungen und Einstellungen bei Schriftgröße und Zeichensatz vornehmen. Außerdem hat er die Möglichkeit Daten, wie Postanschrift oder Passwort, zu ändern und ein Visitenkarte aus den eingegebenen Daten zu erstellen. Nur Name und Benutzername lassen sich nicht ändern.

- **Teilnahme an Kurs**
  Beschreibung: Ein Benutzer ist dazu berechtigt, sich in bestehende Kursangebote einzutragen und darin angebotene Aufgaben und Tests zu lösen.

- **Erstellen einer Arbeitsgruppe**
  Beschreibung: Es besteht für einen Benutzer die Möglichkeit, eine Gruppe zu erstellen, in die er andere Benutzer einladen kann. Dieser Gruppe stehen Ressourcen, wie ein Wiki, Ordner oder ein Forum zur Verfügung.

2.1.2 Autor

- **Alle Funktionen, die Systemrolle Benutzer erfordern**
  Ein Autor kann alle Funktionen der Systemrolle Benutzer in Anspruch nehmen.

- **Erstellung einer Lernressource**
  Beschreibung: Ein Autor kann Lernressourcen erstellen. Diese sind: Kurs, Test, Fragebogen, Ressourcenordner, Wiki und Glosar und können vom Autor publiziert werden, damit sie sichtbar für alle Bentzer sind. Der Autor kann einem erstellen Kurs einen Test, Fragebogen, usw. zuordnen. In einem Kurs kann einer anderen Lernressource zusätzlich eine Bewertung hinzugefügt werden. Andere Benutzer können diesem Kurs beitreten, falls der Autor ihm eine Lerngruppe und eine Einschreibung zuordenet.

- **Lerngruppe für Lernressource erstellen**
  Beschreibung: Damit sich andere Benutzer in einen Kurs eintragen können, muss diesem eine Einschreibung und eine Lerngruppe zugeordnet sein. Benutzer, die sich in die Einschreibung eintragen, werden automatisch der Lerngruppe hinzugefügt. Es können aber auch Benutzer in eine Lerngruppe manuell vom Autor eingetragen werden.

- **Rechtegruppe für Lernressource erstellen**
  Beschreibung: Rechtegruppen zu einem Kurs können von einem Autor erstellt werden, um deren Mitgliedern besondere Rechte innerhalb einer Kurses zu geben. Zum Beispiel kann eine Gruppe K̈ontrolleureërstellt werden, die das Recht haben Bewertungen für in den eingetragene Benutzer vorzunehmen.

2.1.3 Gruppenverwalter

- **Alle Funktionen, die Systemrolle Benutzer erfordern**
  Ein Gruppenverwalter kann alle Funktionen der Systemrolle Benutzer in Anspruch nehmen.

- **Verwaltung von Gruppenkontexten**
  Beschreibung: Ein Gruppenverwalter kann einen Gruppenkontext erstellen, dem er Kursen zuordnen kann. Das ist von Vorteil, wenn eine bestimmte Menge von Benutzern mehrere Kurse belegen muss, z.B. alle Pflichtkurse im Grundstudium für Informatik. Einem Lernkontext werden kursübergreifende Lerngruppen hinzugefügt, in die Benutzer eingetragen werden, die dem Lernkontext zugehörigen Kurse besuchen müssen und analog wird einem Rechtekontext Rechtegruppen hinzugefügt.

- **Verwaltung von kursübergreifenden Lerngruppen**
  Beschreibung: Einer kursübergreifenden Lerngruppe wurden durch den Lernkontext, dem sie angehört, mehrere Kurse zugeordnet. Es macht zum Beispiel Sinn, einem Gruppenkontext mehrere kursübergreifende Lerngruppen zuzuordnen, wenn man die Benutzer in die Jahrgänge einteilen möchte.

- **Verwaltung von kursübergreifenden Rechtegruppen**
  Beschreibung: Kursübergreifende Rechtegruppen können genauso wie die kursüberkreifenden

Lerngruppen zu einem Gruppenkontext hinzugefügt werden. Sie dienen dem Gruppenverwalter, um Benutzern dieser Rechtegruppe bestimmte Rechte für eine bestimmte Menge von Kursen geben zu können.

2.1.4 Benutzerverwalter

- **Alle Funktionen, die Systemrolle Benutzer erfordern**
  Ein Benutzerverwalter kann alle Funktionen der Systemrolle Benutzer in Anspruch nehmen.
- **Neuen Benutzer erstellen**
  Beschreibung: Der Benutzerverwalter darf dem Olatsystem neue Benutzer hinzufügen. Er muss dazu Vorname, Nachname und E-Mailadresse eingeben, und einen Benutzername festlegen, Optional auch das Passwort, Adressen, und anderes. Der neu erstellte Benutzer, kann Namen und Benutzernamen nicht ändern, dadurch ist die Authentizität des Benutzers gegeben.
- **Neuen Benutzer importieren** Beschreibung: Falls der Benutzerverwalter eine Exel (oder OpenOffice) -Tabelle der neu zu erstellenden Benutzer mit allen relevanten Daten (Name, Benutzename, E-Mailadresse) besitzt, kann er diese importieren, und Olat erstellt daraus neue Benutzer.
- **Benutzer Rolle zuweisen**
  Beschreibung: Einem erstellten Benutzer kann der Benutzerverwalter eine oder mehrere der hier beschrieben Rollen zuweisen. Der Benutzer hat dann die Rechte, die mit der ihm zugewiesenen Rolle korrespondieren.

2.1.5 Administrator

- **Funktionen aller anderen Systemrollen**
  Ein Administrator kann alle Funktionen der bisher aufgeführten Systemrollen in Anspruch nehmen.
- **Administrative Funktionen**
  Beschreibung: Der Administrator hat das Recht Systeminformationen, wie zum Beispiel alle aufgetretenen Fehler anzusehen und zu verwalten, Quota verwalten und Properties erweitern. Die Benachrichtigungsmails für abonnierte Themen (z.B. Foren) werden jeweils einmal pro Tag versendet. Der Administrator kann die Benachrichtigungen sofort veranlassen.

3 Grundsätzliche Struktur- und Entwurfsprinzipien für das Gesamtsystem

3.1 Authentifizierung

Um überhaupt angebotene Funktionen in Anspruch zu nehmen muss sich eine Person am System per Login authentifizieren. Dies geschieht entweder als Gast oder, insofern bereits registriert, mit gewähltem Benutzernamen und Passwort. Einmal am System angemeldet ist eine Person berechtigt diejenigen Funktionen in Anspruch nehmen, die sich aus Angehörigkeit der oben beschriebenen Systemrollen ergeben.

3.2 Authorisierung/Typ- und Instanzrechte

Hat sich eine Person gegenüber dem System mit bestimmter Systemrolle authentifiziert, kann sie zunächst nur mit den durch ihre Systemrolle implizierten Rechten auf diverse Ressourcen zugreifen. Besitzt die Person kein Recht, Ressourcen eines bestimmten Typs zu erstellen, so kann sie dennoch von einem Eigentümer einer Ressource des gleichen Typs zu einem gleichrangigen Miteigentümer ernannt werden. Dies authorisiert die betreffende Person einerseits die betreffende Instanz des Ressourcentyps zu manipulieren. Andererseits ist sie nun auch berechtigt, neue Ressourcen des gleichen Typs erstellen, insofern und solange sie Eigentümer wenigstens einer Instanz dieses Ressourcentyps bleibt. Bspw. kann eine Person in der Rolle eines normalen Benutzers von einem Autor zu einem Miteigentümer einer Lernressource ernannt werden. Danach kann die betreffende Person ebenso Lernressourcen erstellen, obwohl sie nicht der Systemgruppe Autor angehört.

3.3 Policy-Konzept

Der Zugriff auf eine bestimmte Funktion im OLAT ist immer eindeutig gekennzeichnet durch die **Identität** des jeweiligen Benutzers und die Ressource auf der die Funktion ausgeführt werden soll. Um nun zu entscheiden, ob der der Benutzer im Kontext seiner Zugehörigkeit zu verschiedenen **Gruppen** tatsächlich **Rechte** besitzt, auf die gewünschte **Ressource** per gewünschter Funktion zuzugreifen, bedient sich OLAT sogenannter **Policies**.

- **Identität**
  Für jeden registrierten Nutzer existiert eine eindeutige, im System hinterlegte Identität (siehe 4.x), mittels derer die Zugehörigkeit zu Systemgruppen und anderen Benutzergruppen überprüft werden kann.

- **Gruppen**
  Im OLAT werden Gruppen ihrem Zweck nach grundlegend unterschieden in **SecurityGroups** und **BusinessGroups**. Während BusinessGroups die Zusammenarbeit von Benutzern durch entsprechende Funktionen unterstützen, erlangen SecurityGroups Bedeutung im Kontext Rechtemanagement.

- **SecurityGroups**
  SecurityGroups stellen zunächst nichts weiter dar als eine generische Implementierung einer Gruppe ohne jeden Kontext, die eine Menge von Identitäten zusammenfasst. Der Präfix S̈ecurityẅird dadurch gerechtfertigt, dass SecurityGroups durch eine oder mehrere Policies referenziert werden können. Wird eine SecurityGroup durch eine Policy referenziert, so erhalten alle Identitäten, die der betreffenden SecurityGroup angehören, das in der Policy verankerte Recht.

- **Ressource**
  OLAT unterscheidet eine Reihe verschiedener, abstrakter Ressourcentypen (siehe 4.x) unterschiedlicher Komplexität, bspw. Kursressourcen, aber auch einfachere wie BusinessGroups. Entscheidend dabei ist, dass konkreten Ressourcenausprägungen SecurityGroups zugeordnet werden. Bspw. werden einer Ressourcenausprägung vom Typ BusinessGroup stets zwei SecurityGroups zugeordnet, wodurch Eigentümern und Teilnehmern der BusinessGroup in voneinander getrennten SecurityGroups enthalten sein können, die wiederum durch verschiedene Policies referenzeiert werden können.

- **Rechte**
  OLAT-Rechte beschreiben in positiver Art und Weise die Erlaubnis, etwas zu tun. Die Gesamtheit aller Rechte eines Benutzers ergibt sich aus der Summe aller Rechte, die ihm aufgrund seiner Angehörigkeit zu bestimmten Gruppen zukommen. Rechte sind demnach additiv.

- **Policy**
  Ein Policy referenziert genau eine Ressource sowie genau eine SecurityGroup und enthält genau ein Recht. Dadurch wird es einem Benutzer des Systems ermöglicht, auf die betreffende Ressource mit angegebenem Recht zuzugreifen, falls seine Identität in mindestens einer SecurityGroup enthalten ist, die von einer das benötigte Recht enthaltenden Policy referenziert wird.

4 Grundsätzliche Struktur- und Entwurfsprinzipien einzelner Pakete

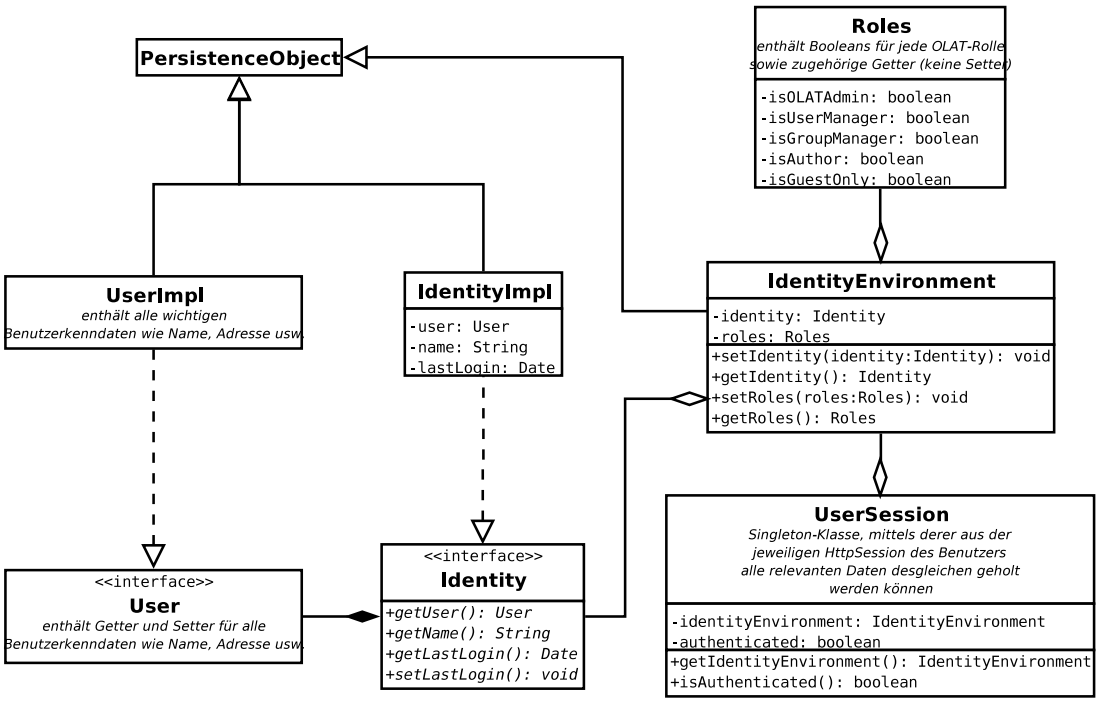
Im Rahmen unserer Analyse im Kontext Rechte- und Rollenmanagement erlangten folgende Pakete, Klassen und Interfaces Bedeutung:

4.1 org.olat.user.UserImpl

Diese Klasse dient der Repräsentation und Persistierung von systemunabhängigen Daten registrierter OLAT-Nutzer, wie Name, Anschrift oder Kontaktdaten.

4.2 org.olat.core.id

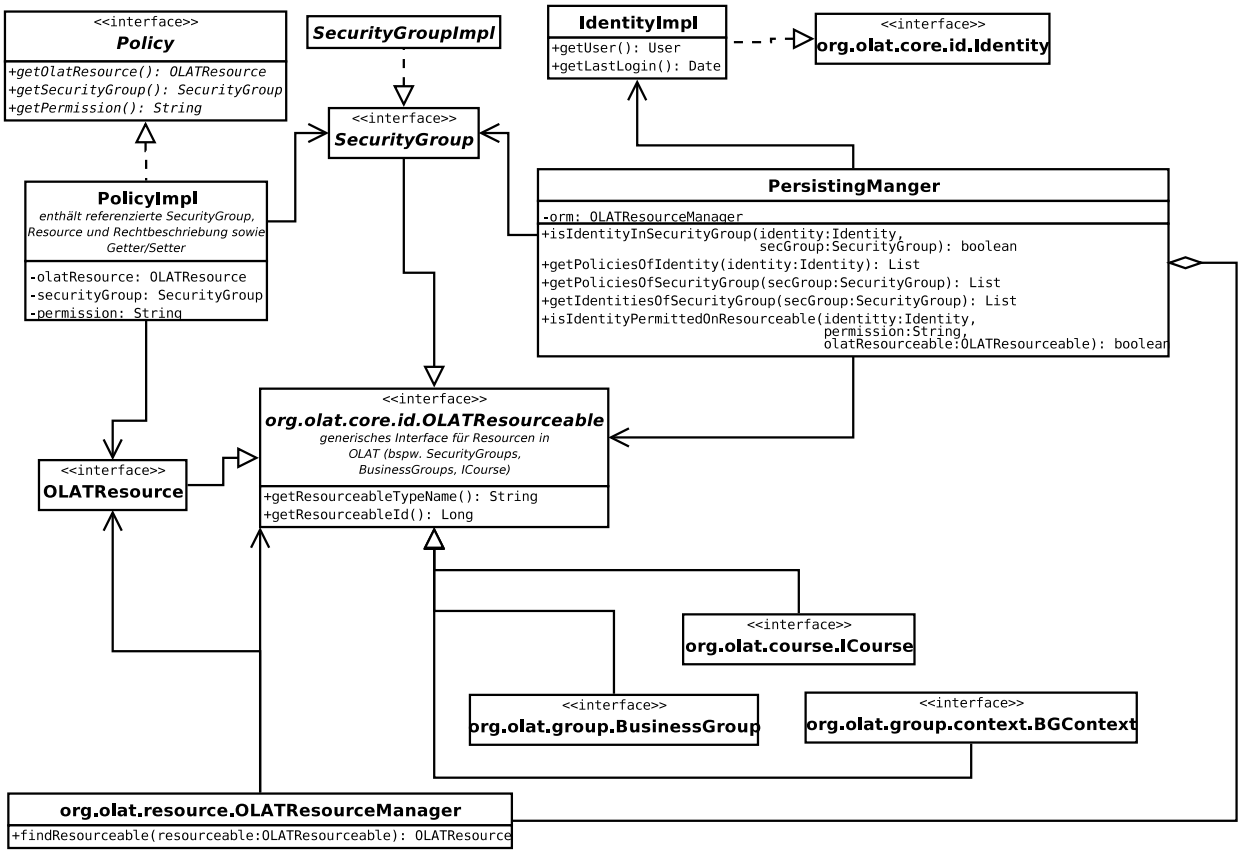
In diesem Package werden verschiedene Interfaces und Klassen definiert, die im Kontext des Rechte- und Rollenkonzepts besonders für die interne Repräsentation eines registrierten und angemeldeten OLAT-Nutzers von Interesse sind. Denn durch das Interface **Identity.java** (und deren konkreter Implementierung **org.olat.basesecurity.IdentityImpl.java** wird ein erster Grundpfeiler des Policy-Konzepts realisiert, wie das folgende Klassendiagramm zeigt:



Insofern man also auf die Instanz der aktuellen **UserSession** zugreifen kann, lassen sich die Identität und die zugewiesenen Systemrollen eines angemeldeten Nutzers eindeutig bestimmen.

4.3 org.olat.basesecurity

Dieses Packet ist das wohl wichtigste im Kontext des Rechte- und Rollenmanagements, da es weitere wichtige Interfaces ud Klassen zur Realisierung des Policy-Konzepts enthält. Weil das Policy-Konzept aber letztlich dem Zweck dient, Zugriffe auf Ressourcen im OLAT zu administrieren, müssen auch einige Interfaces und Klassen betrachtet werden, die OLAT-Ressourcen intern darstellen oder die zu deren Verwaltung dienen, wie das folgende Klassendiagramm zeigt:



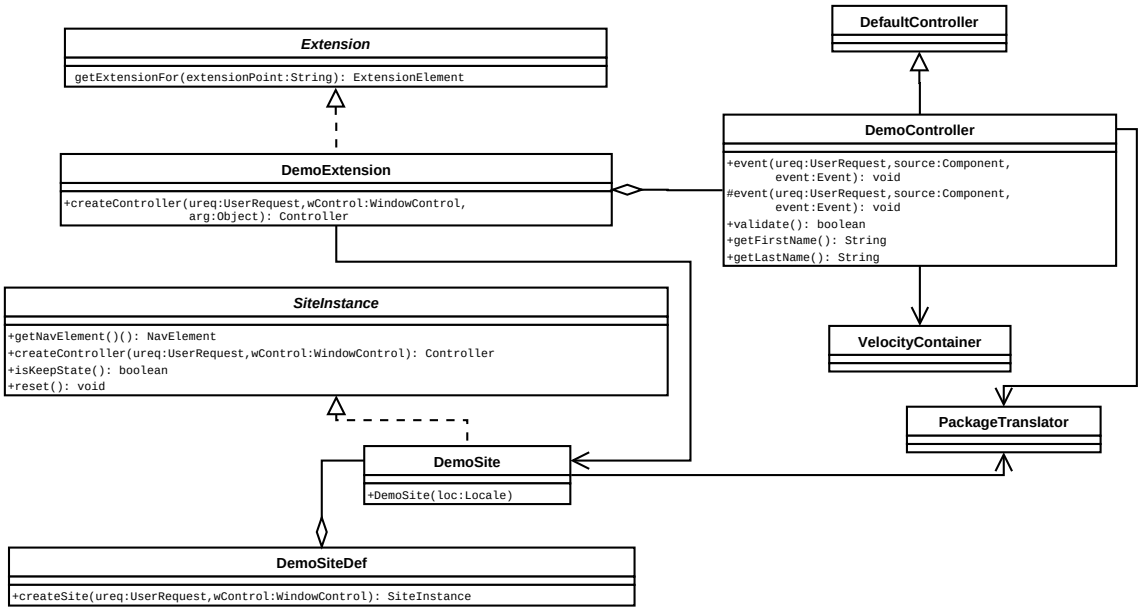
Man erkennt, dass über die Klasse PersistingManager, bzw. deren angebotenen Methoden, überprüft werden kann, welche Identities zu welchen SecurityGroups gehören oder welche Policies eine bestimmte SecurityGroup und damit implizit eine bestimmte Identity referenzieren. Mittels des vom PersistingManager referenzierten OLATResourceManagers, der den eindeutigen Schlüssel einer bestimmten OLAT-Ressource ermittelt, wird es darüber hinaus möglich über einen Datenbank-LookUp zu checken, ob eine Policy mit fraglichem Zugriffsrecht existiert, die über eine beliebige SecurityGroup eine bestimmte Identity referenziert. Damit kann eindeutig entschieden werden, ob einer bestimmten Identität der Zugriff auf eine fragliche OLAT-Ressource erlaubt oder verwährt wird.

5 Rechte- und Rollenkonzept in Erweiterungspunkten

Extensions, die in Olat entweder über die ExtensionPoints org.olat.home.HomeMainController und org.olat.gui.control.generic.dtabs.DTabs eingebunden werden, implementieren Interfaces, welche

createController(UserRequest ureq, WindowControl wControl, ...) anbieten.
D.h. wenn in der HomeView eine Extention vom Interface-Typ ActionExtension erzeugt wird, wird ein Controller übergeben, der in der Extension selber implementiert ist und zumeist vom Typ DefaultExtension abgeleitet wird. Dies geschieht in dem die Nutzereingaben verarbeitet werden. Wird DTabs erweitert so wird der SiteCreator implementiert, welcher eine SiteDefinition anbietet, und diese wiederum SiteInstances, die createController anbieten. Dieser erstellt die Seiten und führt die entsprechende Logik aus.
Dazu wird dem Controller ein **UserRequest-Objekt** übergeben, mit dem er Zugriff auf sämliche Nutzerdaten erhält. Mithilfe von **UserRequest.getUserSession()** erhält man zum Beispiel das UserSession-Objekt das angemeldeten Nutzer, aus dem man die **Rollen auslesen** kann. Damit lässt sich, basierend auf dem Olat-Rollenkonzept, das Rechtemanagement jeder Extension organisieren.
Nutzer lassen sich somit über das normale OLAT-Interface verwalten, ebenso Gruppen. In der Extension müssen nur die Rechte überprüft werden.

Ein Beispiel für eine Extension bildet die DemoExtension im Olat-Package ch.goodsolutions.demoextension. Da die DemoExtension bereits die interessantesten Schnittstellen nutzt, wird hier kurz mittels einen Klassendiagramm mit den wichtigsen Funktionen, ihre Integration in Olat dargestellt.



Die Klasse DemoExtension implementiert das Interface Extension und nutzt die ExtensionPoints org.olat.home.HomeMainController und org.olat.core.gui.control.generic.dtabs.DTabs um ein neuen Tab im HomeView zu erstellen und erzeugt mittels dessen einen neuen DemoController und eine neue SiteDef.
Der DemoController erweitert die Klasse DefaultController und benötigt auf alle Fälle Instanzen des PackageTranslators und des VelocityContainers um HTML und gegebenenfalls eine Übersetzung zu erzeugen. Er kümmert sich vorallem um die abgefangenen Events. Mittels der SiteDef werden neue Instanzen der DemoSite erzeugt, die wiederum von der Klasse DemoExtension genutzt werden.
Um diese Extension zu aktivieren, reicht es alle benötigten Dateien in eine Jar-Datei zu packen und diese in der olat_extension.xml anzugeben.