Herausforderungen des Identity Management an Hochschulen – Problemfeld Datenintegration

Manuel Haim[1]

Abstract: Die zentrale Verwaltung von Personendaten, Benutzer-Accounts und Zugriffsrechten für verschiedene IT-Systeme – kurz die *Benutzerverwaltung* – erfolgt an Hochschulen meist autonom durch zentrale Einrichtungen wie Bibliotheken und Rechenzentren. Sie wird häufig als Kernaufgabe des *Identity Management (IDM)* verstanden, ist aber solchen Anforderungen wie der regelmäßigen und verbindlichen Datenpflege nicht mehr gewachsen. Vielmehr ist eine zunehmende Integration des IDM in die Geschäftsprozesse und IT-Systeme der Hochschule erforderlich, um auf Änderungen am Personenstamm zeitnah und zuverlässig reagieren zu können.
Dieser Beitrag erläutert am Beispiel der Philipps-Universität Marburg, mit welchen besonderen Herausforderungen Hochschulen bei Einführung, Betrieb und Weiterentwicklung eines IDM-Systems konfrontiert werden. Neben einer *Anforderungsanalyse* wird das Problemfeld der *Datenintegration* herausgearbeitet und ein Weg vorgestellt, wie sich die bestehende Benutzerverwaltung in wenigen Schritten zu einem vollwertig integrierten IDM-System weiterentwickeln lässt.

Keywords: Identity Management, IDM, Datenintegration, Provisioning.

1 Motivation

Stellen wir uns eine beliebige deutsche Hochschule vor: *Welche konkreten Personen* gehören jetzt im Augenblick z.B. zur Philipps-Universität Marburg? Und *in welcher Funktion* sind diese Personen tätig? – Was zunächst nach einer einfachen Zählaufgabe anmutet, erweist sich bei näherer Betrachtung als ein komplexes Problem: Hochschulen sind zumeist dezentral organisiert, Fachbereiche und Einrichtungen arbeiten weitestgehend autonom, ihre Angehörigen sind in den unterschiedlichsten Untereinheiten und Verhältnissen tätig. Das Studierendensekretariat pflegt zwar die Studierendendaten; die Personalabteilung kennt zumindest die Arbeitsverträge; viele weitere Personengruppen und Details müssen aber bei Bedarf mühsam bei zahlreichen Stellen erfragt werden (z.B. Doktoranden, Lehraufträge, konkrete Arbeitsgebiete, Funktionen, Ehrenämter, Gasttätigkeiten usw.).

Bislang wurde dieser Umstand von manchen Hochschulen selten als Problem empfunden, denn es gab wenig Grund und Anlass, ein tagesaktuelles Gesamtverzeichnis aller Personen zu führen. Die zentralen Einrichtungen wie Bibliotheken und Rechenzentren, die Dienste für alle Hochschulangehörigen anbieten, behelfen sich hingegen oftmals mit ihren eigenen Benutzerdatenbanken: Personen werden auf schriftlichen Antrag aufgenommen (z.B. gegen Vorlage von Verträgen oder Bestätigung von Vorgesetzten) und erhalten so einen persönlichen, in der Regel befristeten Zugang.

[1] Philipps-Universität Marburg, Hochschulrechenzentrum, Hans-Meerwein-Straße 6, 35032 Marburg, haim@hrz.uni-marburg.de

Die zunehmende Digitalisierung und der Zuwachs an bereichsübergreifenden Prozessen (Stichwort: Prozessorientierte Hochschule) führen jedoch zu neuen Anforderungen:

- Mittels *Shibboleth* [Sh17] soll die organisationsübergreifende Authentifizierung und Autorisierung gegenüber Online-Literatur und Webanwendungen anderer Hochschulen ermöglicht werden. Hierfür sind innerhalb der *DFN-AAI-Föderation* [DF15] sinnvollerweise nur persönliche Accounts zugelassen, deren Inhaber zweifelsfrei identifiziert wurden, ihre Zugangsdaten auf sicherem Weg erhalten haben und deren Daten bei Änderungen binnen zwei Wochen aktualisiert oder gesperrt werden.
- Im *integrierten Campus-Management (iCM)* sollen sämtliche Prozesse des Studienmanagements vereint werden: „[...] von der Bewerbung über modulbezogene Lehrveranstaltungen und Prüfungen bis hin zur Erstellung von Leistungsübersichten und Zeugnissen“ [Ph14]. Dazu müssen alle an Studium und Lehre beteiligten Personen bekannt sein – das sind neben den Studierenden und dem Personal insbesondere auch die wechselnden Lehrbeauftragten, Doktoranden, Ehrenamtliche und Gäste.
- Ein *Forschungs-Informations-System (FIS)* [Ph16, He17] soll die Forschungsprozesse unterstützen: Von der Planung und Dokumentation von Forschungsprojekten über die Erfassung von Publikationen bis hin zur Berichterstattung nach dem Landeshochschulgesetz. Hierfür müssen alle an der Forschung beteiligten Personen mitsamt ihrer Organisationszugehörigkeit und Funktion bekannt sein und zur Nutzung des FIS-Systems verpflichtet werden.
- Das neue *Hochschulstatistikgesetz (HStatG)* fordert eine regelmäßige Erhebung aller Doktorandinnen und Doktoranden der Hochschule ab dem Berichtsjahr 2017.

Diesen Anforderungen ist eine antragsbasierte Benutzerverwaltung nicht mehr gewachsen, da sie in puncto Qualität und Vollständigkeit der Benutzerdaten entsprechende Mängel aufweist. Vielmehr ist eine Integration der Benutzerverwaltung als *Identity-Management-System (IDM-System)* in die Geschäftsprozesse und IT-Systeme der Hochschule erforderlich, um Änderungen an Personendaten unmittelbar dort abzugreifen, wo sie entstehen.

2 Anforderungen an IDM-Systeme

Der Begriff *Identity Management (IDM)* wird in der Literatur sowie von diversen Software-Anbietern auf vielfältige Weise und in scheinbar unterschiedlichem Umfang interpretiert. In diesem Abschnitt sollen daher zunächst diejenigen Anforderungen genannt und erläutert werden, die bislang im Rahmen der Weiterentwicklung der Benutzerverwaltung an der Philipps-Universität identifiziert wurden und für IDM-Systeme an Hochschulen typisch erscheinen (vgl. [Ha16b, S. 17]). Im weiteren Verlauf des Beitrags können so die Defizite der Benutzerverwaltung konkret benannt werden. Die Anforderungen lassen sich in *funktionale* und *nicht-funktionale Anforderungen* unterteilen.

Die **funktionalen Anforderungen** decken sich überraschend gut mit einer allgemeinen Klassifizierung der Funktionen von IDM-Systemen, die Steffen Hofmann in [Ho07, S. 8–

14] aufgestellt hat. Zum besseren Vergleich werden Hofmanns Begriffe nachfolgend in eckigen Klammern *[]* beigefügt:

2.1 Zentrales Verzeichnis *[Informationsspeicher]*

Für Zwecke wie z.B. die öffentlichen Webseiten, Telefonverzeichnisse oder Webanwendungen wird eine *gemeinsame, verlässliche Datenbasis* benötigt, z.B. eine Datenbank oder ein Verzeichnisdienst. Hier überschneidet sich das vorwiegend auf Personen bezogene Identity Management ggf. mit dem auf beliebige Objekte bezogene Stammdaten-Management (Master Data Management, MDM) und es können ggf. ähnliche Technologien oder dieselbe Datenbank verwendet werden (vgl. auch Unterabschnitt 2.3).

2.2 Datenintegration *[Datenintegration]*

Die Integration des IDM-Systems in bestehende Geschäftsprozesse und IT-Systeme, die Personendaten liefern, ist an Hochschulen mit besonderen Schwierigkeiten verbunden. Es bedarf der nachhaltigen Unterstützung der Hochschulleitung, um die Mitwirkung aller beteiligten Stellen zu erreichen. Konkret gibt es hierbei die folgenden Herausforderungen:

2.2.1 Umgang mit zahlreichen Datenquellen

Unterschiedliche *Personengruppen* bzw. *Funktionen* werden von ganz unterschiedlichen Stellen erfasst bzw. zugeteilt. Tabelle 1 bietet einen groben (unvollständigen) Überblick.

Personengruppe	Verantwortlich	Aktion	Datenbank
Studierende	Studierendensekretariat	Immatrikulation	HISinOne
Professoren/-innen	Personalabteilung	Einstllg. / Verbeamtg.	Hessen-SAP
Mitarbeiter/-innen	Personalabteilung	Einstllg. / Verbeamtg.	Hessen-SAP
Landesbedienstetes Personal des Klinikums	Personalabt. Klinikum	Einstllg. / Verbeamtg.	Klinikums-SAP
apl. Professoren/-innen	Senat	Titelverleihung	unbekannt
Honorarprofessoren/-innen	Senat	Titelverleihung	unbekannt
Privatdozenten/-innen	jew. Dekanat	Titelverleihung	unbekannt
Lehrbeauftragte	jew. Dekanat	Erteilung Lehrauftrag	unbekannt
Doktoranden/-innen	jew. Dekanat	Annahme	unbekannt
Dekane/-innen	jew. Fachbereichsrat	Wahl	unbekannt
Fachbereichsbeauftragte	jew. Dekanat	Benennung	unbekannt

Tab. 1: Übersicht einiger Personengruppen und Datenquellen

Anmerkungen: Das Uni-Klinikum Marburg wurde privatisiert, ein Teil des Klinikumspersonals wird aber weiterhin vom Land Hessen bezahlt und wirkt am Fachbereich Medizin der Philipps-Universität mit. Daneben gibt es zahlreiche weitere Sondergruppen und Kooperationsvereinbarungen, die an dieser Stelle jedoch nicht näher erläutert werden sollen.

2.2.2 Umgang mit unterschiedlicher Datenqualität

Je nach Zielsetzung und Sorgfalt der datenführenden Stellen können die dort vorgehaltenen Informationen für ein zentrales Identity Management mehr oder weniger brauchbar sein.

Generell sollte die *Zweckmäßigkeit* des jeweils datenführenden Systems hinterfragt werden. Liegen die Daten z.B. auf Papier oder elektronisch vor? Werden die Daten regelmäßig aktualisiert? Ist die Bedeutung der Daten eindeutig? Entsprechen z.B. die zugeordneten Organisationseinheiten dem tatsächlichen Einsatzbereich einer Person? Oder deuten sie nur auf die verantwortlichen Kostenstellen hin, aus denen die Planstelle finanziert wird?

Darüber hinaus stellt sich die Frage nach der *Vertrauenswürdigkeit* der Personendaten. Hat eine Person sich z.B. per Webformular selbst registriert? Wurde die Identität der Person überprüft, z.B. anhand von Ausweis oder Zeugnissen? Wurde der Ausweis persönlich vorgelegt? Wurden weitere Identifizierungsmerkmale erfasst wie z.B. das Geschlecht oder das Geburtsdatum? Können reale Personen den Daten zweifelsfrei zugeordnet werden?

Wie steht es konkret um die *Erfassung des Namens*? Ist der Name vollständig, d.h. sind alle Vor- und Nachnamen enthalten? Oder nur der Rufname? Ist der Name evtl. verkürzt, z.B. „Bernd“ statt „Bernhard“? Wie wahrscheinlich sind Tippfehler? Stimmt die Schreibweise mit den amtlichen Ausweisdokumenten überein? Und ist der Name noch aktuell?

2.2.3 Umgang mit Dubletten

Dieselbe Person kann ggf. *mehrfach* erfasst worden sein – z.B. in unterschiedlichen Quellsystemen, in leicht abweichenden Schreibweisen, aufgrund einer zwischenzeitlichen Namensänderung oder ggf. völlig abweichenden Namen bei doppelter Staatsbürgerschaft.

Mögliche Dubletten sollten regelmäßig *aufgespürt* und *zusammengeführt* werden (ggf. in Rücksprache mit den betreffenden Personen, sofern sich die Übereinstimmung nicht ohnehin z.B. aus Name, Geburtsdatum und Fachgebiet ergibt). Auf die Zusammenführung von unterschiedlich vertrauenswürdigen Datensätzen (Stichwort: Selbstregistrierung) sollte ggf. verzichtet werden, nicht zuletzt um Identitätsdiebstahl zu verhindern.

2.3 Datenabgleich *[Provisioning]*

Sowohl im IDM als auch im Stammdaten-Management (Master Data Management, MDM) spielt der *regelmäßige Datenabgleich* (Synchronization bzw. Provisioning / Deprovisioning) zwischen IT-Systemen eine zentrale Rolle. Hierfür gibt es am Markt verschiedenste Softwarelösungen – teils eigenständig, teils in IDM- oder MDM-Software integriert – mit recht unterschiedlichem Funktionsumfang. Aber auch Eigenentwicklungen sind denkbar.

Das Grundproblem ist jedoch von der Software unabhängig: Ein *Regelwerk* muss definiert werden, das festlegt, nach welchen Regeln Daten miteinander abgeglichen werden

sollen. Schließlich müssen *Konnektoren* programmiert werden. Beides ist angesichts der vielfältigen Datenquellen an Hochschulen eine nicht-triviale Entwicklungsaufgabe.

2.4 Authentisierung / Authentifizierung *[Authentifizierung, Passwort Management]*

Um sich gegenüber IT-Systemen *ausweisen (authentisieren)* zu können, benötigt jede Person einen persönlichen Zugang bzw. Account. Die wohl gängigste Authentisierungsmethode ist die Eingabe von Benutzername und Passwort. Diese *Zugangsdaten* müssen der Person auf sicherem Weg übermittelt werden, z.B. persönlich am Helpdesk oder per Hauspost an die Dienstanschrift, damit sie nicht in die Hände Dritter gelangen.

Aus Gründen der Benutzerfreundlichkeit (und zur Vermeidung von Passwort-Spickzetteln) sollten möglichst alle IT-Systeme mit *denselben Zugangsdaten* zugänglich sein, idealerweise bereits nach einmaliger Anmeldung (Single Sign On, z.B. per Kerberos oder Shibboleth).

Eine Ausnahme stellen *kritische Systeme und Aktionen* dar, die man gesondert gegen Passwortdiebstahl absichern sollte, wie z.B. die Prüfungsanmeldung oder Finanzverwaltung. Hierfür ist die Abfrage eines *zweiten Faktors* sinnvoll, z.B. einer Transaktionsnummer (TAN) aus einer TAN-Liste, oder eines per Hardware-Token oder Smartphone-App generierten Einmalpassworts (One-Time Password, OTP).

Für Dienste wie WLAN, die aus praktischen Gründen häufig mit einer Speicherung des Passworts verbunden sind, sind zusätzliche *anwendungs- oder gerätebezogene Passwörter* wünschenswert, die nur zu dem einen Zweck verwendet werden können, z.B. das Smartphone mit dem Hochschul-WLAN zu verbinden.

2.5 Autorisierung und Access Management *[Autorisierung]*

Abhängig von ihrem Verhältnis zur Universität steht einer Person das Nutzungsrecht für eine definierte Teilmenge von IT-Systemen und deren Inhalten zu. *Rollen und Rechte* müssen dazu digital erteilt, gepflegt und entzogen werden – idealerweise automatisiert. Sie sollten dazu an *Bedingungen* geknüpft sein wie z.B. an die Personengruppe, an die Abteilung, an ein Ablaufdatum oder eine regelmäßig notwendige formale Verlängerung.

Ggf. sind *Übergangsfristen* nötig, damit z.B. Lehrbeauftragte noch Seminararbeiten korrigieren oder ehemalige Studierende ihre Prüfungsergebnisse elektronisch abrufen können.

Die *Anhäufung von Rechten* z.B. aufgrund mehrfacher Abteilungswechsel sollte vermieden werden (Stichwort: Praktikanten haben die meisten Rechte).

2.6 Webportal *[User Self-Service, (De-)Zentrale Administration, Workflow Mgmt.]*

Zur Einsicht und Verwaltung der Daten, Passwörter, E-Mail-Adressen usw. sollten entsprechende *Webformulare* oder eine eigene *Webanwendung* bereitgestellt werden. Ein *Self Ser-*

vice verringert nebenbei den Arbeitsaufwand für Helpdesk und Administratoren/-innen, da die Benutzer/innen sich selbst helfen können. Sofern Daten öffentlich sichtbar sind, haben letztere erfahrungsgemäß ein großes Interesse daran, sie regelmäßig zu aktualisieren.

2.7 Audit Logging *[Auditing]*

Änderungen an Account, Rollen und Rechten müssen (ggf. rechtssicher) *protokolliert* werden, um im Zweifelsfall nachvollziehen zu können, wer wann welche Änderungen vorgenommen hat. Sofern sich die Änderungen längerfristig auswirken (z.B. ein längerfristiges Ablaufdatum), sollten die Protokolleinträge entsprechend lang aufgehoben werden.

Nicht-funktionale Anforderungen:

2.8 Datenschutz

Die Verarbeitung personenbezogener Daten erfordert eine Einhaltung der geltenden Datenschutzgesetze. In Deutschland muss die Verarbeitung z.B. gemäß Bundes- (BDSG) und jeweiligem Landesdatenschutzgesetz (LDSG) in einem *Verfahrensverzeichnis* dokumentiert und dieses dem Datenschutzbeauftragten der Universität vorab zur Prüfung vorgelegt werden; die personenbezogenen Daten dürfen anschließend nur zu den festgelegten Zwecken verarbeitet und weiterverwendet werden. Sofern *Personaldaten* verarbeitet werden, ist außerdem der Personalrat einzubeziehen.

Aufbewahrungs- und Löschfristen sind zu definieren und einzuhalten. Die Aufbewahrungsfristen können sich je nach IT-System unterscheiden: Sie können rein technisch bedingt sein (z.B. durch Vorhalten einer allgemeinen Datensicherung/Backup für 3 Monate) oder sich auf Rechtsvorschriften stützen (z.B. gesetzliche Pflicht zur revisionssicheren Archivierung von Daten der Finanzbuchhaltung für 10 Jahre). Die Löschfristen sollten aus Gründen der Datensparsamkeit möglichst kurz gehalten werden.

Leicht übersehen wird der Umgang mit vormals vergebenen *Benutzernamen und E-Mail-Adressen*. Hierbei handelt es sich einerseits um personenbezogene Daten, die schnellstmöglich gelöscht werden sollten. Andererseits kann die Wiedervergabe an Dritte zu unerwünschten Nebeneffekten führen, wenn z.B. in Drittsystemen noch gleichnamige Benutzerkonten existieren oder E-Mail-Adressen schon allgemein bekannt sind. Sofern man solche Bezeichner von der Wiedervergabe an Dritte ausschließen möchte, müssen sie längerfristig gespeichert werden. Ferner bleibt die Frage, ob denn zumindest die Wiedervergabe an die ursprünglichen Nutzer/innen erwünscht ist. Falls ja, ist es nötig, zusätzlich zum Bezeichner auch persönliche Daten vorzuhalten, die eine spätere Identifikation ermöglichen.

Es ist außerdem festzulegen, *wer wann welche Daten bearbeiten und einsehen darf*. Soll z.B. die Personalabteilung Zugriff auf alle Identitäten im IDM-System erhalten, um bestehende Studierende direkt ins Personalsystem übernehmen zu können? Oder lieber auf ihr Personalsystem und das dort verzeichnete Personal beschränkt bleiben?

2.9 Randbedingungen / Kosten

Sofern die Anschaffung einer IDM-Software in Betracht gezogen wird, sind die *laufenden Kosten für Lizenzen, Anpassungen und Wartung* zu berücksichtigen. Die Anzahl der Personen an einer Hochschule ist vergleichsweise groß; sie kann aufgrund steigender Studierendenzahlen, neuer Angebote und Kooperationen jederzeit sprunghaft ansteigen. Lizenzmodelle, die sich linear an der Anzahl der zu verwaltenden Identitäten orientieren, sind für Hochschulen daher tendenziell unattraktiv; Lizenzverlängerungen, die von Jahr zu Jahr unvorhersehbar teurer werden, ebenso.

Darüber hinaus ist zu klären, *in welchem Zeitrahmen* Anpassungen möglich sind (wie z.B. Veränderungen von Workflows oder Entwicklung neuer Konnektoren) bzw. ob diese auch *eigenständig ohne Verletzung der Wartungsverträge* durchgeführt werden dürfen.

3 Bisherige Benutzerverwaltung an der Philipps-Universität

In diesem Abschnitt wird die bestehende Benutzerverwaltung der Philipps-Universität kurz vorgestellt. Hierbei werden insbesondere *Mängel bei der Datenintegration* deutlich.

Bereits seit 1995/1996 stellt das Hochschulrechenzentrum (HRZ) der Philipps-Universität allen interessierten Studierenden sowie Professoren/-innen und Mitarbeitern/-innen auf Antrag einen zentralen E-Mail-Account zur Verfügung. Dieser Account wurde schon bald für die Authentisierung gegenüber weiteren Diensten genutzt. Die Selbstverwaltung erfolgt seither über Webformulare (CGI-Skripte) auf den E-Mail-Servern.

Seit 2001 dient ein *OpenLDAP-System* als zentraler Verzeichnisdienst. Historisch bedingt wird zwischen Students- und Staff-Accounts sowie zwischen Students- und Staff-Personeneinträgen unterschieden, die in getrennten Zweigen gespeichert sind. Weitere Zweige umfassen u.a. Gebäude, Telefone, Netzwerk-Hosts sowie Organisationseinheiten (letztere hierarchisch). Die einzelnen Objekte werden über Links miteinander verknüpft.

Bei den *Studierenden und anderen Veranstaltungsteilnehmern (Students)* gibt es stets eine 1:1-Beziehung zwischen dem Account, dem Personeneintrag und dem Primärschlüssel im jeweiligen Quellsystem (z.B. der Matrikelnummer beim Studierendensekretariat), so dass einer automatischen Datenübernahme und Accountgenerierung nichts im Wege steht. Für die Datenübernahme gibt es entsprechende Perl-Skripte, die nach dem Erhalt eines Datenexports bislang noch manuell angestoßen werden.

Bei den *Mitarbeitern/-innen und anderweitig Tätigen (Staff)* ist es etwas komplizierter. Die Personeneinträge sind ggf. zugleich als *Visitenkarten* im Organigramm auf den Webseiten der Universität sichtbar – d.h. eine natürliche Person, die in mehreren Bereichen tätig ist, kann über mehrere solcher Einträge verfügen. Zusammengehörende Personeneinträge tragen nach Möglichkeit dieselbe Personen-ID, die sich aus Nachnamen, Vornamen und einer laufenden Nummer zusammensetzt. Einen Hauptdatensatz gibt es nicht, d.h. weitere Objekte wie Accounts sind jeweils nur mit einem der vorhandenen Personeneinträge verknüpft. Die Pflege der Personeneinträge und anderer LDAP-Objekte erfolgt manuell über

eine selbst entwickelte Webanwendung (myLDAPadmin), Anpassungen vorhandener Einträge sind so auch dezentral über die Fachbereiche möglich. So wird z.B. die internetweite Sichtbarkeit von Personeneinträgen durch eine schriftliche Einverständniserklärung geregelt, die durch Personendatenbeauftragte am Fachbereich gesammelt und bearbeitet wird.

Im Rahmen der Benutzerverwaltung bietet ein menügeführtes Perl-Skript den HRZ-Administratoren/-innen die Möglichkeit, *Accounts* zu verwalten oder *Mitarbeiterdaten* aus anderen Quellen zu übernehmen. Bei der Bearbeitung von Account-Anträgen, die in Form von Papierformularen eingereicht wurden, wird zunächst der Personeneintrag manuell per myLDAPadmin angelegt oder angepasst, dann der Account per Perl-Skript erzeugt, wobei Teilaufträge für weitere Systeme (in Form von Shell-Kommandos) in einer Queue abgelegt werden. Dahingegen werden bei der automatisierten Übernahme von Mitarbeiterdaten (z.B. bei Personalmeldungen) zunächst für jeden Quell-Personendatensatz die möglicherweise passenden LDAP-Personeneinträge aufgelistet; die tatsächliche Zuordnung oder Neuerstellung von Personeneinträgen erfolgt jedoch erst nach manueller Bestätigung. Veränderte Daten werden ggf. übernommen und ein ggf. vorhandener Account verlängert.

Das Problem hierbei: An den Personeneinträgen und Accounts der Mitarbeiter/innen ist meist nicht mehr ablesbar, welche Daten und Fristen sich aus welcher Quelle ergeben. Bei Unklarheiten müssen die archivierten Account-Anträge und Log-Dateien betrachtet werden. Dies macht eine automatisierte und zuverlässige Pflege der Rollen und Rechte unmöglich. Außerdem häufen Personen ggf. mehrere Personeneinträge an, die von den Fachbereichen oder vom HRZ manuell gepflegt oder bereinigt werden müssen.

4 Aktuelle Weiterentwicklung an der Philipps-Universität

Um die Benutzerverwaltung zu einem universitätsweiten IDM-System weiterzuentwickeln, müssen insbesondere die *Mängel bei der Datenintegration* beseitigt und ein *Regelwerk für den Datenabgleich* konzipiert werden. Die hierfür relevanten Nahziele (und bisherige Fortschritte) werden in den nachfolgenden vier Punkten kurz erörtert.

4.1 Durchgängige Erfassung der Personendaten aller an der Universität tätigen Personengruppen (dezentral durch geeignete Stellen, in geeigneten Systemen)

Anhand der laufend eingereichten Account-Anträge auf Papierformularen und bestehenden Datenaustausche wurde in den letzten Jahren zunächst eine *detaillierte Aufstellung aller zu erfassenden Personengruppen* erstellt. Hierbei war vor allem die Benennung derjenigen Gruppen interessant, die bislang nicht systematisch an das HRZ gemeldet werden, aber trotzdem einen Account oder Personeneintrag erhalten. Insgesamt konnten so knapp 40 Personengruppen ausfindig gemacht werden. Diese Größenordnung ist für eine Hochschule nicht ungewöhnlich, wie die Erfahrungen anderer Hochschulen zeigen (z.B. FAU Erlangen-Nürnberg [Fr16]).

Es folgten (teils motiviert durch weitere Aufgabengebiete wie Shibboleth oder Campus-Management) *Gespräche mit den Betreibern der bestehenden datenführenden Systeme*, um mehr über die Datenqualität, Pflegeprozesse und Bedeutung der Quelldaten zu erfahren. Im Vordergrund standen hierbei die Studierenden und das Personal, da diese den Großteil der Hochschulangehörigen ausmachen und ihnen im Universitätsalltag die meisten Rechte zustehen.

Noch offen ist die *Ausgestaltung der elektronischen Erfassungs- und Meldeverfahren* für viele übrige Gruppen, die bislang nicht oder nur unzureichend elektronisch erfasst werden. Hierfür müssen die verantwortlichen Stellen eingebunden und neue Prozesse geschaffen werden. Es ist denkbar, für die Datenpflege vorhandene Systeme wie HISinOne zu nutzen (mit zusätzlichen Personenklassen) oder eigene Datenbanken bereitzustellen.

4.2 Konsolidierung der Personendaten und zeitlich beschränkten Teil-Identitäten in einer zentralen Datenbank des Hochschulrechenzentrums

Bislang gingen beim Datenimport nach LDAP Informationen verloren, da Daten aus unterschiedlichen Quellen direkt zu einem LDAP-Personeneintrag verschmolzen wurden. Um die Informationen über die *Teil-Identitäten* nicht zu verlieren, wird daher künftig für jede Datenquelle ein separater Verzeichniszweig im LDAP bereitgestellt, in welchem die Quelldatensätze als (Schatten-)Kopie abgelegt werden. In einem weiteren Zweig wird dann separat die *Identität* gepflegt, welche Links zu beliebig vielen Teil-Identitäten erhalten kann. Ziel ist, für jede natürliche Person möglichst nur *eine* Identität vorzuhalten. Die bisherigen Personeneinträge sollen dahingegen wie unterschiedliche Visitenkarten behandelt und ebenfalls mit der Identität verknüpft werden.

Der Datenabgleich erfolgt folglich in zwei Schritten. In einem ersten Schritt werden die Quelldatenbanken 1:1 mit den Schattenkopien abgeglichen – und hierbei die Daten bereits in ein einheitliches Format gebracht. Zur Unterstützung des Datenabgleichs erhalten die Schattenkopien jeweils einen Zeitstempel, der Aufschluss über den Zeitpunkt der Erstellung, letzten Änderung sowie Löschung gibt. Dies ermöglicht später einen zeitnahen Delta-Abgleich mit weiteren LDAP-Zweigen (im Gegensatz zu einem zeitaufwendigen Komplettabgleich).

In einem zweiten Schritt müssen die Schattenkopien mit den Identitäten abgeglichen werden. Die Zuordnung neuer Schattenkopien zu Identitäten soll für die meisten Personengruppen vorerst weiterhin manuell erfolgen, um unnötige Dopplungen oder Falschzuordnungen zu vermeiden. Der automatische Abgleich wird dann nur für die bereits verlinkten Identitäten ausgeführt, wobei die Daten in den Schattenkopien (sofern sie voneinander abweichen) mit unterschiedlicher Priorität bewertet werden können. In Rücksprache mit den betreffenden Personen sind Overwrites möglich, z.B. falls nur der Rufname oder (z.B. bei Ligaturen) eine von den amtlichen Dokumenten abweichende Schreibweise erwünscht ist.

Als *Werkzeug für den Datenabgleich* wurde eine eigene, nur wenige hundert Zeilen Code umfassende Python-Bibliothek namens *CALYPSO* entwickelt [Ha16a]. Diese bietet neben einheitlichen Datenkonnektoren zwei wesentliche, leicht zu konfigurierende Algorithmen:

- Der *Sync-Algorithmus* vergleicht die Datensätze in Quell- und Zieldatenbank 1:1 anhand eines frei wählbaren Attribut-Mappings und reagiert auf vordefinierte Situationen (z.B. absent, found, ambiguous, source_missing) mit wählbaren, vordefinierten Aktionen (z.B. create, update, ignore, delete). Dieses Vorgehen ist an gängige IDM-Software angelehnt (z.B. OpenIDM [Fo16], midPoint [Ev14]).
- Der *Merge-Algorithmus* sucht alle Quell-Datensätze, die zu einem spezifischen Ziel-Datensatz passen, und führt deren Daten zu einem temporären Datensatz zusammen, um diesen anschließend mit dem Ziel-Datensatz abzugleichen. Auf diese Weise lässt sich z.B. die Summe der aktuellen Rollen und Berechtigungen für eine Identität oder einen Account ableiten. Für die Zusammenführung der gefundenen Datensätze kann eine eigene Funktion angegeben werden, z.B. um die Quelldaten mit unterschiedlicher Priorität zu behandeln und so der Namensschreibweise der Personalabteilung Vorrang vor anderen Schreibweisen zu gewähren.

Zu guter Letzt wollen die gesammelten Daten sinnvoll verwaltet werden. Für die weitere Administration und Selbstverwaltung von Personendaten und Accounts ist daher die *Entwicklung einer eigenen Webanwendung* auf Basis von AngularJS und Bootstrap (browserseitig) sowie Flask und uWSGI (serverseitig) geplant; die Kommunikation zwischen Browser und Server wird über REST erfolgen.

4.3 Automatisierte Ableitung von Accountlaufzeit, Rollen und Rechten

Um den Benutzern/-innen die ihnen unmittelbar zustehenden digitalen Berechtigungen zuweisen zu können, waren zunächst die *rechtlichen Zugangsvoraussetzungen für die einzelnen Dienste* zu klären. Hierzu erfolgten Gespräche z.B. mit der HRZ-Netzwerkabteilung bzgl. eduroam-WLAN und DFN-Internet, mit der Universitätsbibliothek bzgl. lizensierter Online-Literatur, oder mit der DFN-AAI-Föderation bzgl. Shibboleth.

Mit dem gesammelten Wissen konnte ein Datenquellen-Rollen-Rechte-Modell entwickelt werden; ggf. sind noch Übergangsfristen beim Entzug von Rechten zu klären. Die automatisierte Pflege der Rollen und Rechte wird durch CALYPSO geschehen, wie im vorigen Abschnitt angedeutet.

4.4 Automatische (De-)Provisionierung aller angeschlossenen Systeme

Schließlich sollen die zentral geführten Informationen *zeitnah* mit allen Anwendungssystemen *abgeglichen* werden. So können Benutzerprofile z.B. schon verfügbar gemacht werden, bevor ein Nutzer eine Anwendung erstmalig nutzt – dies vereinfacht z.B. die Gruppenzuordnung in Drittsystemen wie der Lernplattform ILIAS. Aber auch die Löschung von Daten in Drittsystemen soll automatisiert geschehen. Schattenkopien bieten auch hier eine Möglichkeit, im IDM abzubilden, welche Benutzerprofile in Drittsystemen bestehen, erstellt oder gelöscht werden sollen.

5 Erläuterungen zur Vorgehensweise

Die Einführung eines IDM-Systems an der Philipps-Universität hatte lange Zeit niedrige Priorität. Die Problemanalyse und Entwicklung erfolgte daher vorwiegend agil im Rahmen der Arbeitszeit, die für die Pflege und Weiterentwicklung der Benutzerverwaltung im HRZ dauerhaft vorgesehenen ist (60% einer unbefristeten Vollzeitstelle).

Im Jahr 2012 wurde begonnen, die Anforderungen und Mängel der Benutzerverwaltung neu zu dokumentieren sowie ab dem Jahr 2013 auch Fremdsoftware zu evaluieren, um ein universitätsweites IDM aufzubauen. Insbesondere wurde Open-Source-Software getestet, um eine möglichst kostenneutrale Lösung zu entwickeln.

Mit *OpenIDM* konnten dank mitgelieferter Konnektoren und nachvollziehbarer JSON-Konfigurationsdateien schnell erste Datenabgleiche implementiert werden. IDM-Spezifika wie ein sinnvolles Datenschema, eine Dublettensuche oder gar eine Rechteverwaltung suchte man im Jahr 2014 aber vergebens. Der produktive Einsatz (inkl. Updates und Maintenance-Releases) war außerdem nicht mehr Bestandteil der Open-Source-Lizenz, sondern an einen Supportvertrag gekoppelt: Auf die Philipps-Universität wären hierbei jährliche Ausgaben im Umfang von mehreren zehntausend Euro zugekommen (Stand 06/2014). Angesichts der zweifelhaften Mehrwerte wurde auf die Anschaffung verzichtet.

Gegen den Einsatz der aus OpenIDM hervorgegangenen, frei nutzbaren Software *midPoint* sprach im Jahr 2014 die vergleichsweise höhere Lernkurve sowie die nur mühsam lesbare XML-Konfiguration. Nebenbei war ein Datenabgleich ohne Zwischenspeicherung von Master-Objekten nicht vorgesehen. Die Dokumentation sowie die Roadmap klangen zwar vielversprechend, viele Funktionen ließen aber noch auf sich warten.

Ein gemeinsames Manko bei diesen und auch weiteren betrachteten IDM-Lösungen war die mäßige Unterstützung für Portalanpassungen, eigene Webformulare und Workflows. Die guten Erfahrungen von Kollegen mit AngularJS befürworteten eine Eigenentwicklung.

Da im Jahr 2014 bereits der Entwurf für ein neues Datenmodell bestand, beschränkte sich die weitere Suche vorerst auf eine allgemeine Software zum Datenabgleich, die auch in weiteren Bereichen des HRZ eingesetzt werden könnte. Für komplexe Datentransformationen hat sich hier die kostenlose Community Edition von *Pentaho Data Integration (PDI, ehemals Kettle)* nun schon mehrfach bewährt. Die Datenobjekte im IDM sind hingegen vergleichsweise einheitlich, die Algorithmen zum Datenabgleich ebenso – sie müssen lediglich mit dem z.B. aus OpenIDM bekannten Vokabular parametrisiert werden. Eine schlanke Eigenimplementierung in Form einer *Common Algorithm Library for the Provisioning and Synchronization of Objects (CALYPSO)* [Ha16a] lag somit nahe.

6 Fazit

In diesem Beitrag wurde gezeigt, dass sich die funktionalen Anforderungen der Philipps-Universität an ein IDM-System nicht wesentlich vom marktüblichen Funktionsumfang unterscheiden. Sie werden an der Philipps-Universität zum Großteil von bereits implementierten Lösungen abgedeckt, so dass der Erwerb zusätzlicher Software vorerst entfällt.

Auch der Zeit- und Personalaufwand für die sukzessive Weiterentwicklung ist, verglichen mit der Einführung fertiger Lösungen, überschaubar (vgl. Uni Konstanz: Austausch des *Sun Identity Manager* durch *OpenIDM* von ca. 09/2012 bis 02/2015 [ZK17]). Eine besondere Herausforderung stellen in beiden Fällen die *sehr zahlreichen Datenquellen und Verantwortlichkeiten* dar, die zunächst identifiziert bzw. geklärt werden müssen. Dies ist ein langwieriger, nicht-technischer Prozess mit vielen Beteiligten, der unabhängig von der gewählten Software nötig ist. Zudem muss ein *Webportal* meist selbst entwickelt werden.

Der eigentliche *Datenabgleich* erweist sich hingegen als ein geringeres Problem. Hierfür wurden ein Datenmodell und Algorithmen skizziert, wie sie für namhafte IDM-Software üblich sind. Anhand der Beispiel-Implementierung CALYPSO wird deutlich, dass sich die Abgleichmechanismen bereits mit geringem Aufwand nachbilden lassen.

Literaturverzeichnis

[DF15] DFN-Verein: Klassen der Verlässlichkeit in der DFN-AAI, https://www.aai.dfn.de/der-dienst/verlaesslichkeitsklassen/, Stand: 31.03.2015.

[Ev14] Evolveum: midPoint – Synchronization Situations, https://wiki.evolveum.com/display/midPoint/Synchronization+Situations, Stand: 07.01.2014.

[Fo16] ForgeRock: OpenIDM 4.5 Integrator's Guide – Synchronization Situations and Actions, https://backstage.forgerock.com/docs/openidm/4.5/integrators-guide/chap-synchronization#handling-sync, Stand: 19.12.2016.

[Fr16] Friedrich-Alexander-Universität Erlangen-Nürnberg: IdM-Portal, Kundengruppen/-typen, https://www.idm.uni-erlangen.de/aim/docs/affiliations, Stand: 19.12.2016.

[Ha16a] Haim, Manuel: CALYPSO – ein Python-Skript zum generischen Datenabgleich, https://www.zki.de/fileadmin/zki/Arbeitskreise/VD/Protokolle/2016-09-12/calypso-unimr.pdf, Stand: 12.09.2016.

[Ha16b] Haim, Manuel: Von der HRZ-Benutzerverwaltung zum hochschulweiten Identity Management, https://www.zki.de/fileadmin/zki/Arbeitskreise/VD/Protokolle/2016-03-14/idm-haim.pdf, Stand: 14.03.2016.

[He17] HeFIS-Verbund: Ziele und erwartete Mehrwerte, http://www.hefis-verbund.de/fis/mehrwerte, Stand: 03.01.2017.

[Ho07] Hofmann, Steffen: Architektur eines Identitätsmanagementsystems an einer Hochschule. Diplomarbeit, FernUniversität Hagen, Juni 2007, https://www.zedat.fu-berlin.de/pub/ZEDAT/FUDIS/Home/Architektur_eines_Identitaetsmanagementsystems_an_einer_Hochschule.pdf, Stand: 19.12.2016.

[Ph14] Philipps-Universität Marburg: Integriertes Campus-Management, http://www.uni-marburg.de/integriertes-campus-management/projekt, Stand: 09.06.2014.

[Ph16] Philipps-Universität Marburg: Forschungs-Informations-System, http://www.uni-marburg.de/administration/verwaltung/dez1/fis, Stand: 08.06.2016.

[Sh17] Shibboleth Consortium: What's Shibboleth?, http://shibboleth.net/about/, Stand: 03.01.2017.

[ZK17] ZKI-Arbeitskreis Verzeichnisdienste: Protokolle der halbjährlichen Arbeitskreistreffen, https://www.zki.de/arbeitskreise/verzeichnisdienste/protokolle/, Stand: 12.03.2017.