Handbuch Forschungsdatenmanagement

Herausgegeben von
Stephan Büttner, Hans-Christoph Hobohm, Lars Müller

BOCK + HERCHEN Verlag
Bad Honnef
2011

ISBN 978-3-88347-283-6

BOCK+HERCHEN Verlag, Bad Honnef
Printed in Germany

2.7 Systeme und Systemarchitekturen für das Datenmanagement

Matthias Razum
FIZ Karlsruhe – Leibniz-Institut für Informationsinfrastruktur

2.7.1 Einführung

Datenmanagementsysteme (DMS) stellen die technische Basis für die Erfassung, Anreicherung und Bereitstellung von Forschungsdaten dar. Sie umfassen typischerweise neben der eigentlichen Speicherung der Datenobjekte weitere Dienste, etwa zur Registrierung, Suche oder Verwaltung von Zugriffsrechten. Die Publikation von Daten und damit verbundene Dienste gehören im Allgemeinen nicht dazu. Treloar, Groenewegen und Harboe-Ree unterscheiden in „*The Data Curation Continuum*“ (2007) zwei grundlegende Domänen der Verwaltung von Forschungsdaten:

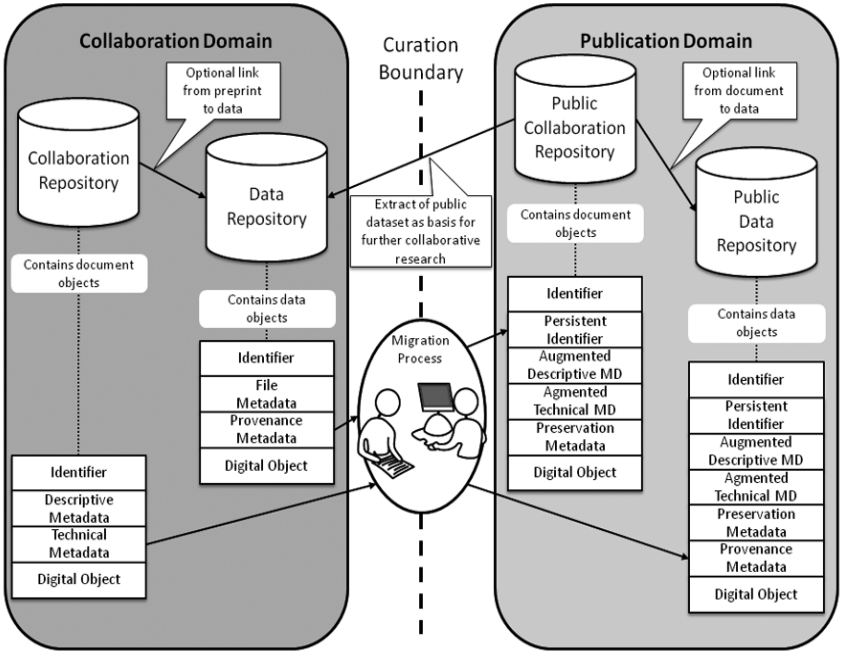

Abb. 1: Collaboration, Publication, and the Curation Boundary (Quelle: nach Treloar, 2007)

DMS decken meist die linke Domäne ab, also das Management von Daten für die eigentliche Forschung und die Zusammenarbeit in (verteilten) Forschungs-

gruppen. Die Publikationsdomäne auf der rechten Seite wird meist durch spezialisierte Systeme abgedeckt.

Publizierte Daten müssen langfristig (perspektivisch: *ad infinitum*) verfügbar bleiben, wie herkömmliche Publikationen zitiert und damit die Grundlage weiterer Publikationen bilden können. Gleichzeitig sind diese Daten statisch, d.h. sie unterliegen keiner Änderung mehr. Derartige Eigenschaften werden zunehmend als Grundvoraussetzung publizierter Daten eingefordert. Entsprechend entwickeln sich zurzeit Zertifizierungsverfahren für Datenzentren, um diese Leistungen zuzusichern, etwa durch *DataCite*[1], aber auch aus der Praxis heraus wie zum Beispiel bei der *Open Access* Datenpublikation *Earth System Science Data*[2], die Standards für die Repositorien einfordert, in denen die publizierten Daten hinterlegt sind.

Systeme zum Datenmanagement verwalten dagegen potenziell alle Daten, auch solche, die nie publiziert werden. Sie berücksichtigen Daten mit einer begrenzten Lebensdauer und eingeschränkter Sichtbarkeit. Sie stellen diese für die weitere Verwendung innerhalb eines Instituts oder einer Forschergruppe bereit, etwa zur Analyse, Aggregation oder für Vergleiche mit anderen Daten. Diese Daten können sich noch ändern bzw. in neueren Versionen gespeichert werden. Hier sind Funktionen wie *Lifecycle Management*, z. B. zur automatischen Überwachung von Haltefristen, etwa gemäß den Empfehlungen der Deutschen Forschungsgemeinschaft (DFG, 1998), Versionierung und Autorisierung gefragt. Oft wählen Autoren nur einen Bruchteil der hier verwalteten Daten aus und machen sie in der „*Publication Domain*“ allgemein verfügbar. Der Übergang von Daten aus einem DMS über die *Curation Boundary* zu einem Datenzentrum sieht z. B. das neue *World Data System* der ICSU (2008) mit der Unterscheidung von „*Data Collection and Processing Facilities*“ und den „*Data Archiving and Publication Facilities*“ vor.

Datenmanagement bildet damit eine grundlegende Voraussetzung für die Datenpublikation, ist aber für sich alleine genommen schon eine der meistgeforderten Dienstleistungen im wissenschaftlichen Alltag, wie aktuelle Befragungen zeigen (Kroll & Forsman, 2010; TIB Hannover, FIZ Chemie Berlin & Universität Paderborn, 2010).

2.7.2 Funktionale Anforderungen an Systeme zum Datenmanagement

Die Anforderungen an ein Datenmanagementsystem (DMS) hängen von den verwalteten Datentypen und den Einsatzszenarien des Systems ab. Es lassen

1. http://www.datacite.org/ [Zugriff am 13.08.2011].
2. http://earth-system-science-data.net/ [Zugriff am 13.08.2011].

sich allerdings einige grundlegende Funktionen benennen. Die folgende Aufzählung erhebt weder Anspruch auf Vollständigkeit, noch müssen immer alle funktionalen Anforderungen für ein spezifisches DMS implementiert sein. Allerdings findet man diese Funktionen in vielen Systemen wieder; sie bilden damit generisch Anforderungen an Systeme zum Datenmanagement ab.

2.7.2.1 Verknüpfung von Daten und Metadaten

Daten ohne Beschreibung sind meist wertlos. Messdaten müssen mit Einheiten verknüpft sein, es muss klar sein, was und wie gemessen wurde. Zu den Messdaten gehören Konfigurations- und Kalibrierungsdaten der Instrumente. Die Entstehungs- und Bearbeitungsgeschichte („*Provenance*“) ist wichtig: wer hat wann was mit den Daten gemacht? Sind die Daten Rohdaten, bereits um Fehler bereinigt oder gar aggregiert? Welche Algorithmen oder Webservices wurden zur Lemmatisierung von Texten herangezogen? Wer hat einen altsprachlichen Text transkribiert oder übersetzt? Alle diese Informationen sind für das Verständnis und die Nachnutzung der Forschungsdaten entscheidend, und entsprechend müssen diese Metadaten gemeinsam mit den eigentlichen Daten gespeichert und verwaltet werden. Dabei können die Metadaten unterschiedliche Ausprägungen haben:

- Technische Metadaten (z. B. Dateiformat, Dateigröße, Mime-type)
- *Provenance* Metadaten (z. B. gemäß *Open Provenance Model* (Moreau et al., 2007) oder PREMIS (Caplan & Guenther, 2005)
- Deskriptive Metadaten (fachspezifisch)
- Lizenz-Metadaten

Aus der Vielzahl der möglichen Profile, insbesondere bei den fachspezifischen deskriptiven Metadaten ergibt sich die Anforderung an ein DMS, mehrere Metadatensätze gemäß beliebiger Profile mit den Forschungsdaten verwalten zu können. Während ein konkreter Metadatensatz durchaus über ein Profil definiert sein sollte, empfiehlt sich ein schemafreier Ansatz für das darunterliegende DMS.

Die Erstellung von Metadaten ist ein aufwändiger und im wissenschaftlichen Alltag nur schwer zu bewältigender Prozess. Gerade bei Datensätzen, die wahrscheinlich nie publiziert werden, unterbleibt vielfach die Beschreibung durch Metadaten. Damit verlieren diese Daten allerdings dramatisch an Wert, da sie quasi nicht mehr auffindbar und nur durch den Urheber noch zu interpretieren sind. Insofern sind für DMS Verfahren zur automatischen Metadatengenerierung interessant. Im Bereich der technischen Metadaten gibt es vielversprechende Ansätze, etwa das *File Information Tool Set* FITS (Stern & McEwen, 2009). Werden die Forschungsdaten bereits in einem sehr frühen Stadium – am besten zum Zeitpunkt der Entstehung – in einem DMS gespeichert, kann das DMS die *Provenance*-Metadaten weitgehend automatisch erfassen.

2.7.2.2 Versionierung

Daten verändern sich gegebenenfalls in den ersten Stadien ihres Lebenszyklus. Sie können um Fehler bereinigt, aggregiert und umformatiert werden. Aber nicht nur die Daten selbst, auch die sie beschreibenden Metadaten können in dieser Phase vor der eigentlichen Publikation ergänzt und korrigiert werden. Wichtig ist, Daten und Metadaten auch hier als Einheit zu betrachten und als Einheit zu versionieren.

Gerade bei verteilt arbeitenden Forschungsgruppen erlaubt eine versionierte Speicherung die Nachvollziehbarkeit von Änderungen. Sie können Personen und Zeitpunkten zugeordnet werden und bilden damit die Grundlage für *Provenance*-Metadaten – und damit Informationen, die die Einschätzung der Daten durch Dritte hinsichtlich Vertrauenswürdigkeit und Korrektheit erleichtern.

2.7.2.3 Datenformate

Forschungsdaten kommen in einer Vielzahl von Dateiformaten vor. Der Versuch einer Vereinheitlichung oder Beschränkung kollidiert mit der Dynamik des Forschungsprozesses. Ein DMS wird aber nur akzeptiert werden, wenn es die Forschenden unterstützt, statt sie einzuschränken. Darüber hinaus ist es oftmals sinnvoll, ein Datenobjekt in mehreren Repräsentationen abzuspeichern, etwa einen Scan eines Manuskripts in voller Auflösung im Dateiformat TIFF, eine reduzierte Auflösung für die Darstellung im Web im JPEG-Format und schließlich eine Vorschau als GIF-Datei. Ein anderer Anwendungsfall mehrerer Repräsentationen kann das Originalformat und eine in ein Standardformat migrierte Version sein. Also sollten DMS möglichst keine Annahmen hinsichtlich von Datenformaten treffen und die Verwaltung mehrerer Repräsentationen eines digitalen Objekts unterstützen.

Bei einer Vielzahl von zu unterstützenden Datenformaten kann es sinnvoll sein, einen Dienst zur Charakterisierung der Formate einzusetzen, um automatisch das Format zu bestimmen und geeignete technische Metadaten zu extrahieren. Ein Beispiel für einen solchen Dienst ist das bereits erwähnte FITS.

2.7.2.4 Semantische Relationen zwischen Datenobjekten

Datenobjekte stehen selten alleine und losgelöst da. Sie stehen in Beziehung zu anderen Datenobjekten. Messwerte sind mit Kalibrierungs- bzw. Konfigurationsdaten von Instrumenten verknüpft, Transkripte und Übersetzungen mit dem Originaltext, Fotos von archäologischen Artefakten mit Informationen zur Ausgrabungsstätte.

Meist drückt man diese Beziehungen durch Techniken des *Semantic Webs*[3] aus; insbesondere das *Resource Description Framework* RDF[4] und die *Web*

3. http://www.w3.org/2001/sw/ [Zugriff am 13.08.2011].
4. http://www.w3.org/RDF/ [Zugriff am 13.08.2011].

Ontology Language OWL[5] spielen hierbei eine wichtige Rolle. In den meisten Projekten oder zumindest Disziplinen kommen mehrere Ontologien zum Einsatz. Über „*(Open) Linked Data*“ (Campbell & MacNeill, 2010) entstehen zurzeit in Teilbereichen Standardisierungsbestrebungen, die sinnvollerweise modular aufgebaut und für einen konkreten Anwendungsfall evaluiert und entsprechend kombiniert werden müssen. Für DMS bedingt das die Unterstützung einer Vielzahl von Ontologien[6]. Auch müssen sie sowohl Relationen innerhalb des DMS als auch darüber hinaus berücksichtigen. Sinnvoll ist es, wenn ein DMS diese Relationen nicht nur verwalten, sondern über eine entsprechende Komponente (z. B. einen *Triple Store*) und eine standardisierte Abfragesprache (etwa SPARQL[7]) durchsuchbar machen.

2.7.2.5 Lebenszyklus von Datenobjekten

Datenobjekte durchlaufen verschiedene Phasen von ihrer Entstehung bis zur Publikation oder gegebenenfalls Löschung. Beispiele solcher Phasen umfassen die Entstehung des Datenobjekts, dessen Anreicherung mit Metadaten, das Durchlaufen einer (gegebenenfalls mehrstufigen) Qualitätssicherung, die Archivierung, die Auswahl und Publikation oder die Löschung des Datenobjekts nach einer festgelegten Haltefrist. Diese Phasen zusammengenommen bilden den Lebenszyklus eines Datenobjekts. Sie unterscheiden sich in ihren konkreten Ausprägungen zwischen Datentypen und Disziplinen, aber es lassen sich grundlegende Anforderungen ableiten:

- Die DFG empfiehlt in ihrer Denkschrift zur guten wissenschaftlichen Praxis eine Haltefrist von 10 Jahren für Daten, die die Basis für eine (herkömmliche) Publikation bilden (DFG, 1998). Grundsätzlich werden viele Daten nicht notwendigerweise auf Dauer gespeichert. Vielfach gibt man eine Haltefrist vor, nach deren Erreichen man über das weitere Vorgehen mit den Daten entscheidet.
- Bereitstellung einer Benachrichtigungsfunktion, die bei definierten *Trigger Events* (z. B. Ende einer Haltefrist, Ende einer Embargofrist, Freigabe eines Datensatzes) die Besitzer der Objekte bzw. die Administratoren informieren.
- Da Haltefristen oftmals weit über die Laufzeit von Projekten hinausgehen, sind die ursprünglichen Erzeuger der Objekte gegebenenfalls nicht mehr greifbar. In diesen Fällen ist ein *Ownership Management* sinnvoll, bei dem der Besitz an Objekten (und damit die Verfügungsgewalt, etwa hinsichtlich weiterer Haltefristen oder Löschungen) an andere Personen übergehen

5. http://www.w3.org/TR/owl-guide/ [Zugriff am 13.08.2011].
6. http://linkeddata.org/ [Zugriff am 13.08.2011].
7. http://www.w3.org/TR/rdf-sparql-query/ [Zugriff am 13.08.2011].

kann. Hierbei kommen neben technischen allerdings auch rechtliche Fragen ins Spiel, etwa im Zusammenhang von Urheber- und Nutzungsrecht: wem gehören die Daten, nachdem ein Mitarbeiter eine Institution verlässt?

2.7.2.6 Registrierung

Die Vergabe von *Persistent Identifiern* ist für die Publikation von Daten eine wichtige Voraussetzung. Üblicherweise vergibt man diese erst im Verlauf des Publikationsprozesses, also in der „*Publication Domain*“, und damit nur für wenige ausgewählte Datensätze und zu einem späten Zeitpunkt im Lebenszyklus der Datenobjekte. Es gibt allerdings Szenarien, in denen eine frühere Vergabe sinnvoll erscheint, gegebenenfalls sogar für eine Vielzahl von Datenobjekten oder -fragmenten. In der Computerlinguistik will man z. B. innerhalb eines Textkorpus häufig bis auf Wort- (bzw. *Token*) oder sogar Phonemebene hinab Fragmente über *Persistent Identifier* dauerhaft zitierfähig machen. In anderen Disziplinen gibt es enorm große Datensätze (z. B. in der Klimaforschung oder bei der Sequenzierung von Genomen), die oftmals nur in Ausschnitten zitiert werden sollen. In diesen Fällen kann es sinnvoll sein, schon im DMS der „*Collaboration Domain*“ für diese Fragmente *Persistent Identifier* zu vergeben. DMS sollten also Schnittstellen zu entsprechenden Systemen vorsehen.

2.7.2.7 Content Models

In den vorangegangenen Abschnitten wurde die Flexibilität immer wieder hervorgehoben, so dass ein DMS letztlich als BLOB-*Store* (*Binary Large Object*) erscheinen könnte. Tatsächlich erschwert aber eine fehlende Typisierung sowohl die Entwicklung fachspezifischer Anwendungen als auch die Validierung der gespeicherten Objekte. Insofern sollten DMS die Typisierung von Objekten erlauben. Ähnlich wie ein relationales Datenbanksystem erst durch ein Datenmodell eine sinnvolle Nutzung erlaubt, sollten DMS einen ähnlichen Mechanismus vorsehen: *Content Models*. Dies ist insbesondere dann relevant, wenn innerhalb eines DMS mehrere *Content Models* nebeneinander zum Einsatz kommen sollen. Datenobjekten weist man einem *Content Model* zu, dem sie „gehorchen“ und damit auch gewisse Eigenschaften zusichern.

Was genau ein *Content Model* vorschreibt, hängt einerseits vom konkreten Einsatzzweck ab, zum anderen aber auch von der zugrundeliegenden Architektur. Typischerweise will man aber zu verwendende Metadatenprofile, Dateiformate oder auch Formate für einzelne Eigenschaften festlegen können. Weiterhin können *Content Models* auch die Darstellung von Datenobjekten steuern.

2.7.2.8 Authentifizierung und Autorisierung

Wissenschaft findet heute meist in verteilten Arbeitsgruppen statt, die sich auch über Institutionsgrenzen hinweg erstrecken können. Oft findet man in diesem Zusammenhang den Begriff „virtuelle Organisation“, die z. B. Projektstrukturen

abbildet und gegebenenfalls nur eine kurze Lebensdauer hat. Virtuelle Organisationen stehen typischerweise orthogonal zur internen Aufbauorganisation der beteiligten Institutionen. Verteilte Authentifizierungssysteme wie etwa *Shibboleth* (Scavo & Cantor, 2005), aber auch *OpenID* (Recordon & Reed, 2006) und ähnliche kommerzielle Ansätze spielen hier eine zunehmend wichtige Rolle.

Forschungsdaten können explizit oder implizit sensitive Information enthalten. Sie geben Einblick in Arbeitsweisen, eingesetzte Verfahren und noch nicht publizierte Erkenntnisse. Daten können ohne hinreichende kontextuelle Informationen zu Fehlinterpretationen einladen, andere enthalten personenbezogene Informationen, die besonders zu schützen sind. Entsprechend sind nicht alle Forschungsdaten von Anfang an frei zugreifbar. Vielfach werden überhaupt nur ausgewählte, aggregierte oder anonymisierte Datensätze nach Veröffentlichung eines Artikels publiziert. In anderen Fällen werden aber auch schon sehr frühe Versionen öffentlich zugänglich gemacht, um Dritten zu erlauben, auf Basis dieser Daten eigene Publikationen zu erarbeiten (z. B. *Sloan Digital Sky Survey*[8]). Selbst innerhalb einer Institution oder Forschergruppe sind Zugriffsbeschränkungen durchaus üblich. Die Entscheidung über die entsprechenden Regeln sollte bei den Wissenschaftlern liegen.

Die Regeln selbst können sich sowohl am Zugreifenden und seinen Rechten als auch am Objekt und seinen Eigenschaften festmachen. Beispiele möglicher Kriterien für Zugriffsregeln umfassen:

- Unterscheidung von Daten und Metadaten
- Unterscheidung der verschiedenen Repräsentationen eines Objekts
- Embargofristen eines Objekts
- Status eines Objekts im Lebenszyklus
- Status eines Benutzers als Mitglied einer Arbeitsgruppe oder virtuellen Organisation

Die Komplexität der möglichen Regeln und zu berücksichtigenden Eigenschaften führt zur nächsten Herausforderung: Wissenschaftler müssen verstehen, welche Zugriffsrechte tatsächlich aus ihren Einstellungen resultieren. Dies erfordert eine Benutzungsoberfläche, die die Konsequenz einer Regeländerung verständlich darstellt, also den Spagat zwischen der Umsetzung komplexer Funktionalitäten und einfacher Bedienung schafft.

2.7.2.9 Vertrauen

Inwieweit kann man Daten Dritter vertrauen? In dieser Frage liegt eine der großen Herausforderungen für die Nachnutzung von Daten, die auch die *High Level Expert Group on Scientific Data* in ihrem Bericht (Giaretta et al., 2010) hervor-

8. http://www.sdss.org [Zugriff 13.08.2011].

hebt. Vertrauen ist weitgehend ein sozio-kultureller Prozess und kann nur teilweise durch technische Systeme unterstützt werden.

Etablierte Prozesse wie *Peer Reviewing* finden zunehmend auch bei der Publikation von Daten statt (z. B. bei der Zeitschrift *Earth System Science Data*[9]). Diese Zeitschrift definiert auch Standards, die von den DMS zugesichert werden müssen, um dort hinterlegte Daten für eine Publikation zu akzeptieren. Darunter finden sich etwa Anforderungen an persistente *Identifier*, *Open Access*, langfristige Archivierung und eine wissenschaftsfreundliche Lizenz. Alle diese Anforderungen beziehen sich auf die „*Publication Domain*". Doch auch in der „*Collaboration Domain*" kann durch technische Maßnahmen das Vertrauen in die Daten erhöht werden. Beispiele hierfür sind

- interne Freigabeprozesse mit Qualitätssicherung, siehe auch *Object Lifecycle*
- *Checksums*, um die Integrität der Daten abzusichern
- *Audit Trails*, um Änderungen an Daten nachverfolgen zu können

Besonders wichtig für die Einschätzung der Vertrauenswürdigkeit von Daten ist deren Entstehungsgeschichte. Es gibt einige Ansätze (Razum et al., 2010; Rajbhandari, Hedges & Fabiane, 2010), die Daten zum Zeitpunkt ihrer Entstehung in einem DMS zu erfassen und dabei kontextuelle Informationen automatisch einzubeziehen, etwa Zeitstempel, die Daten des angemeldeten Benutzers, Verknüpfungen zu Konfigurations- und Kalibrierungsdaten eingesetzter Instrumente oder verwendete Programme. Die Datenmanagementgruppe am IFM-GEOMAR in Kiel[10] arbeitet beispielsweise daran, die Untersuchung von Proben über *Workflows* abzubilden. Jeder Schritt eines *Workflows* (etwa das Reinigen einer Probe, die Durchführung einer Messung, usw.) erfasst neben den Daten auch automatisch Metadaten, die nachvollziehbar machen, welche Prozessschritte wann, von wem und wie oft durchlaufen wurden und wie es schließlich zum Endergebnis kam. DMS müssen in der Lage sein, solche Informationen zu verwalten und möglichst deren (semi-)automatische Erfassung zu unterstützen.

2.7.3 Grundlegende Architekturen

Die Funktionalität von DMS lässt sich in Schichten aufteilen:

- eine Persistenzschicht für die eigentliche Speicherung der Daten,
- eine Kernschicht für die zentralen Funktionen eines DMS sowie
- eine Diensteschicht mit erweiterten Funktionen.

9. http://earth-system-science-data.net/ [Zugriff am 13.08.2011].
10. https://portal.ifm-geomar.de/web/guest/about-us [Zugriff am 13.08.2011].

Diese Aufteilung ist logisch zu verstehen; je nach Art des Systems können diese Schichten explizit oder eher implizit vorhanden sein:

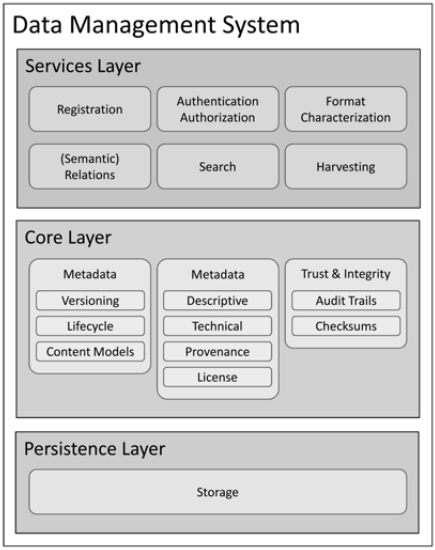

Abb. 2: Schichten und Komponenten einer DMS-Architektur

Bei der Implementierung der Persistenzschicht eines DMS muss man zwischen datensatzorientierten und dateiorientierten Systemen unterscheiden. Erstere verwenden typischerweise eine relationale Datenbank, letztere basieren meist auf sogenannten *Digital Object Repositories* (DOR). Vielfach lassen sich Daten sowohl datensatz- als auch dateiorientiert darstellen. Eine Entscheidung zwischen beiden Optionen fällt meist im Spannungsfeld zwischen langfristiger Archivierung von Daten (dateiorientiert) und interaktivem Arbeiten mit den Daten (datensatzorientiert).

Weiterhin kann man Einzelsysteme von verteilten Systemen unterscheiden. Zu letzteren gehören *Grid*-basierte Systeme und zunehmend auch *Cloud*-basierte Ansätze. Die folgenden Abschnitte führen jeweils einige Vor- und Nachteile der unterschiedlichen Ansätze an und geben Beispiele für derartige Systeme.

Für die Implementierung einer Benutzungsoberfläche für DMS in der „*Collaboration Domain*“ kommen häufig pragmatische Ansätze zum Tragen, etwa die Verwendung von Wikis oder *Content Management* Systeme wie z. B. Plone oder Drupal (ein Beispiel hierfür ist Islandora[11]).

11. http://islandora.ca/ [Zugriff am 13.08.2011].

2.7.3.1 Datenbanken

Datenbanken, und hier besonders relationale Datenbanken, bieten sich für strukturierte Daten an. Der größte Vorteil von relationalen Datenbanksystemen sind die seit Jahren bewiesene Leistungsfähigkeit und Zuverlässigkeit der Software sowie die vorhandene umfangreiche Erfahrung mit ihnen. Funktionen wie Transaktionalität, Clusterfähigkeit und hohe Geschwindigkeit sowie etablierte Werkzeuge zur Datensicherung und -wiederherstellung sprechen für Datenbanken. Wichtig ist darüber hinaus die Möglichkeit, auch aus sehr großen Datenmengen über *Queries* beliebige Teilmengen zu extrahieren. Beispiele für derartige Systeme sind *Data Warehouses* in der Bioinformatik[12] oder das *World Data Center* WDC-MARE[13], das Daten aus dem Bereich der Meereskunde vorhält.

Die Skalierbarkeit von Datenbanken in den Terabyte-Bereich erfordert allerdings ein großes technisches Verständnis und ist mit hohem administrativem Aufwand verbunden. Eine weitere Herausforderung von Datenbank-basierten Systemen ist die Nachverfolgung von Änderungen, die Versionierung und die Erfassung von *Provenance*-Daten. Dies ist nicht technisch bedingt, sondern resultiert aus der meist pragmatisch erfolgten Datenmodellierung, die diese Aspekte in der Regel nicht beachten. Gerade bei aufwändig aufgebauten Datensammlungen kann diese Unterlassung den Wert der Daten, insbesondere hinsichtlich einer Nachnutzung, einschränken. Buneman spricht hier von der Notwendigkeit sogenannter „*Curated Databases*“ (Buneman, Cheney, Tan & Vansummeren, 2008), die vom Design und der Implementierung her aber deutlich aufwändiger sind und Erfahrung mit den grundlegenden funktionalen Anforderungen (s.o.) erfordern.

Neben relationalen Datenbanksystemen können auch spezialisierte Systeme zum Einsatz kommen, z. B. XML-Datenbanken oder sogenannte No-SQL-Datenbanken[14]. Letztere skalieren sehr gut und lassen sich einfach verteilen, bieten aber nur eingeschränkte Konsistenz („*eventual consistency*“). Das *Compact Muon Solenoid* (CMS) Experiment, Teil des Large *Hadron Colliders* am CERN, nutzt zum Beispiel für sein Datenmanagement *Couch*-DB[15]. In beiden Fällen handelt es sich aber um relativ neue Technologie, zu der noch wenig Erfahrung existiert.

12. Beispiele für derartige Systeme sind die Lipase *Engineering Database* (http://www.led.uni-stuttgart.de/ [Zugriff am 13.08.2011]) bzw. die CYP450 *Engineering Database* (http://www.cyped.uni-stuttgart.de/ [Zugriff am 13.08.2011]) am Institut für Technische Biochemie an der Universität Stuttgart.

13. http://www.wdc-mare.org/ [Zugriff am 13.08.2011].

14. Beispiele für No-SQL-Datenbanken sind Cassandra (http://cassandra.apache.org/ [Zugriff am 13.08.2011]) oder *Couch*-DB (http://couchdb.apache.org/ [Zugriff am 13.08.2011]).

15. http://www.couch.io/case-study-cern [Zugriff am 13.08.2011].

2.7.3.2 Digital Object Repositories

In vielen Fällen liegen Daten aber nicht strukturiert, sondern semi- oder unstrukturiert als Dokumente oder Dateien vor. Hier sind Schema-basierte Ansätze umständlich und unflexibel. *Digital Objekt Repositories* unterstützen die dateiorientierte Speicherung und haben sich, ausgehend von der Bibliothekswelt, zunehmend auch im Bereich der Forschungsdaten etabliert. Bekannte Vertreter sind etwa EPrints[16], aDORe[17] und Fedora[18] (*Flexible Extensible Digital Object Repository Architecture*) zusammen mit darauf aufsetzenden Systemen wie etwa eSciDoc[19] oder EASY[20]. Die meisten dieser Systeme orientieren sich am OAIS-Referenzmodel *Consultative Committee for Space Data Systems* (CCSDS) (2002). Grundsätzlich können auch herkömmliche Dokumentenmanagement-Systeme wie z. B. Alfresco[21] oder *Content Repositories* wie JackRabbit[22] zum Einsatz kommen, die allerdings nur Teile der oben genannten Anforderungen erfüllen und damit nur eingeschränkt für das wissenschaftliche Datenmanagement eignen.

Vorteile des dateiorientierten Ansatzes sind Schemafreiheit, Unterstützung beliebiger Dateiformate für den *Content* und ausgezeichnete horizontale Skalierbarkeit. Ein *Repository* verhält sich – stark vereinfacht – wie ein Webserver: es ist sehr einfach, weitere Webserver (bzw. *Repositories*) hinzuzufügen und Daten auf die verschiedenen Instanzen aufzuteilen. Die Sicherung oder Replizierung der Daten kann auf Dateisystemebene erfolgen, was bis für mittlere *Repository*-Größen einfach zu bewerkstelligen ist. Wächst die Anzahl der Dateien zu stark an, kann sich das aber insbesondere hinsichtlich der Sicherung und vor allem der Wiederherstellung ins Gegenteil verkehren. Durch technische Maßnahmen (*Snapshot*-fähige Dateisysteme, Verteilung der Daten auf diverse Dateisysteme, Einsatz von *Cloud*- oder *Grid*-Technologie) lassen sich diese Probleme umgehen.

2.7.3.3 Grid und Cloud

Eine weitere Architekturoption für DMS stellen *Grid*-Systeme dar, also Systeme zum verteilten Rechnen und zur Speicherung von Daten über mehrere (räumlich verteilte) Knoten. Dabei stellt eine *Middleware* (z. B. *Globus Toolkit*, gLite oder UNICORE) Dienste zur Verfügung, die aus Sicht des Anwenders die

16. http://www.eprints.org/ [Zugriff am 13.08.2011].
17. http://african.lanl.gov/aDORe/projects/adoreArchive/ [Zugriff am 13.08.2011].
18. http://www.fedora-commons.org/ [Zugriff am 13.08.2011].
19. http://www.escidoc.org/ [Zugriff am 13.08.2011].
20. https://easy.dans.knaw.nl/dms [Zugriff am 13.08.2011].
21. http://www.alfresco.com/ [Zugriff am 13.08.2011].
22. http://jackrabbit.apache.org/ [Zugriff am 13.08.2011].

Komplexität der *Grid*-Infrastruktur hinter einer Dienste-Schicht verbirgt. *Grid*-Systeme können sowohl Rechen- wie auch Speicherressourcen zur Verfügung stellen. Je nach Schwerpunkt spricht man auch von *Computational Grids* und *Storage* oder *Data Grids*, wobei letzterer Begriff eher aus der Industrie kommt und momentan durch (*Private*) *Storage Clouds* verdrängt wird. In *Grid*-Systemen bieten Nutzer von Diensten bzw. Ressourcen meist auch eigene Ressourcen an. In Deutschland koordiniert *D-Grid* (Gentzsch, 2006) den Aufbau und den Betrieb einer solchen Infrastruktur. Im Bereich der Geisteswissenschaften zeigt das Projekt *TextGrid* (Gietz, et al., 2006), dass *Grid*-Technologie durchaus auch für die langfristige Speicherung umfangreicher Textkorpora (als Primärdaten z. B. für Linguisten) geeignet ist. Einen etwas generischeren Ansatz verfolgt *Diligent* (A *Digital Library Infrastructure on Grid Enabled Technology*)[23].

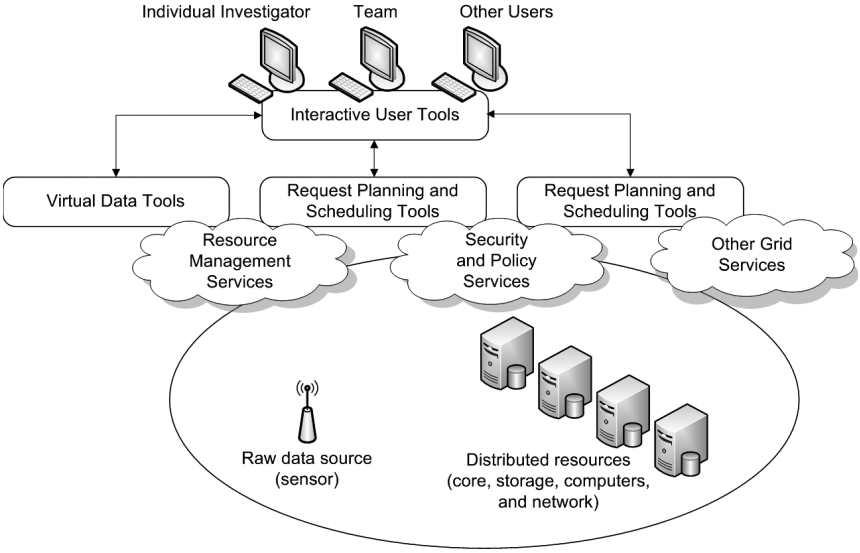

Abb. 3: Beispiel für den grundlegenden Aufbau eines *Data Grids* am Beispiel des *Grid Physics Network* (Quelle: nach Foster, 2003)

Cloud Computing basiert ebenfalls auf verteilten Diensten; hier bietet aber im Gegensatz zum *Grid* ein zentraler Anbieter Kunden seine Infrastruktur an. Neben kommerziellen Angeboten bauen Firmen und Universitäten inzwischen vielfach sogenannte „*Private Clouds*“ auf, bei denen die Ressourcen auf eigenen Systemen vor Ort vorgehalten werden. Hier lässt sich eines der größten Hemmnisse für den breiten Einsatz von *Grid*- und *Cloud*-Technologien umgehen,

23. http://diligent.ercim.eu/ [Zugriff am 13.08.2011].

nämlich das notwendige Vertrauen in die Sicherheit, Vertraulichkeit, Zuverlässigkeit und Langfristigkeit der angebotenen Leistungen.

Duraspace, die Organisation hinter Fedora und *DSpace*, arbeitet an einem *Cloud*-basierten *Storage Layer* für *Digital Object Repositories* namens *Duracloud*[24].

2.7.3.4 Mischformen

In der Praxis kommen meist Mischformen der vorgestellten Systeme zum Einsatz, also z. B. die Kombination von DOR und relationalen Datenbanken. Sei es, dass die Metadaten in einem RDBMS, die Daten aber im Dateisystem liegen, oder dass eine Mischung von strukturierten und unstrukturierten Daten vorliegt. Auch die Kombination von DOR und *Grid*-Technologie via SRB oder iRODS kommt vor, insbesondere im Umfeld von Fedora, für das entsprechende Plugins bereitstehen. Das oben erwähnte *TextGrid*-Projekt arbeitet an einer solchen Kombination. Hier empfehlen sich pragmatische Entscheidungen für eine geeignete Architektur für ein DMS, die sich insbesondere an den Anforderungen der Wissenschaftler, der Strukturiertheit der Daten und den Anforderungen an *Provenance* und Langzeitarchivierung orientieren sollte.

2.7.3.5 Ausblick

Die Heterogenität der Daten- und Dateiformate, der Größe und Menge der Daten, die Vielzahl der unterschiedlichen intendierten Anwendungsfällen eines DMS, die Verschiedenheit der disziplinspezifischen Anforderungen machen es unmöglich, eine übergreifende Architektur zu beschrieben. Neue Technologien wie z. B. NoSQL-Datenbanken erweitern die möglichen Komponenten einer DMS-Architektur. Insofern werden Mischformen zunehmend an Bedeutung gewinnen, in denen unterschiedliche technische Lösungen für die verschiedenen Datentypen und Anforderungen kombiniert und zu einem System zusammengefasst werden – hoffentlich transparent für Wissenschaftler, der seine Daten möglichst einfach verwalten, archivieren, publizieren und mit Mitarbeitern und Kollegen austauschen möchte.

---

24. http://www.duraspace.org/duracloud.php [Zugriff am 13.08.2011].

Literaturhinweise

Berners-Lee, T. Hendler, J. & Lassila, O., 2001. The Semantic Web. *Scientific American,* 284(5), S. 34–43.

Buneman, P. Cheney, J. Tan, W.-C. & Vansummeren, S, 2008. Curated databases. *Proceedings of the twenty-seventh ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*. Vancouver, Kanada 9.-12. Juni 2008. New York, NY: ACM, S.1–12.

Campbell, L. M. & MacNeill, S., 2010. *The Semantic Web, Linked and Open Data*. Bolton: JISC CETIS. Online: http://wiki.cetis.ac.uk/images/1/1a/The_Semantic_Web.pdf [Zugriff am 27.01.2011].

Caplan, P. & Guenther, R. S., 2005. Practical Preservation: The PREMIS Experience. *Library Trends*, 54 (1), S. 111–124.

CCSDS (Consultative Committee for Space Data Systems), 2002. *Reference Model for an Open Archival Information System (OAIS)*. Washington, DC: National Aeronautics and Space Administration.

DFG (Deutsche Forschungsgemeinschaft), 1998. *Vorschläge zur Sicherung guter wissenschaftlicher Praxis: Empfehlungen der Kommission „Selbstkontrolle in der Wissenschaft" (Denkschrift).* Weinheim: Wiley-VCH.

Foster, I., 2003. The Grid: A New Infrastructure for 21st Century Science. In: F. Berman, G. Fox & T. Hey, ed. 2003. *Grid Computing: Making the Global Infrastructure a Reality.* Chichester: John Wiley & Sons. doi: 10.1002/0470867167.ch2.

Gentzsch, W., 2006. D-Grid, an E-Science Framework for German Scientists. *ISPDC '06: Proceedings of the Proceedings of The Fifth International Symposium on Parallel and Distributed Computing*. Timisoara, Rumänien 6.-9. Juli 2006. Los Alamitos, Calif.: IEEE Computer Society, S. 12–13.

Giaretta, D. et al., 2010. *Riding the wave – How Europe cain gain from the rising tide of scientific data. Final report of the high level expert group on scientific data*. Online: http://cordis.europa.eu/fp7/ict/e-infrastructure/docs/hlg-sdi-report.pdf [Zugriff am 18.12.2010].

Gietz, P. et al., 2006. TextGrid and eHumanities. *Second IEEE International Conference on e-Science and Grid Computing.* Amsterdam, Niederlande 4.-6. Dez. 2006. Los Alamitos, Calif.: IEEE Computer Society, S. 133–141.

ICSU (International Council for Science), 2008. *Ad hoc Strategic Committee on Information and Data – Final Report to the ICSU Committee on Scientific*

Planning and Review. Online: http://www.icsu.org/Gestion/img/ICSU_DOC_DOWNLOAD/2123_DD_FILE_SCID_Report.pdf [Zugriff am 12.12.2010].

Kroll, S. & Forsman, R., 2010. A Slice of Research Life: Information Support for Research in the United States. *Report commissioned by OCLC Relesearch in support with the RLG partnership.* Online: http://www.oclc.org/research/publications/library/2010/2010–15.pdf [Zugriff am 10.02.2010].

Moreau, L. et al., 2007. *The Open Provenance Model*. Online: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.148.7394 [Zugriff am 23.11.2010].

Rajbhandari, S. Hedges, M. & Fabiane, S., 2010. BRIL – Capturing Experiments in the Wild. *5th International Conference on Open Repositories.* Madrid, Spanien 6.-9. Juli 2010. Online: http://or2010.fecyt.es/Resources/documentos/GSabstracts/BRIL.pdf [Zugriff am 15. 01.2011].

Razum, M. et al., 2010. Research Data Management in the Lab in the Lab. *5th International Conference on Open Repositories.* Madrid, Spanien 6.-9. Juli 2010. Online: http://or2010.fecyt.es/Resources/documentos/GSabstracts/ResearchDataManagamentIinTheLab.pdf [Zugriff am 15.01.2011].

Razum, M. Schwichtenberg, F. Wagner, S. & Hoppe, M., 2009. eSciDoc Infrastructure: A Fedora-Based e-Research Framework. In M. Agosti et al., ed., Research and Advanced Technology for Digital Libraries, *13th European Conference, ECDL 2009.* (LNCS 5714). Korfu, Griechenland, 27. Sept.-2. Okt. 2009. Berlin: Springer, S. 227–238.

Recordon, D. & Reed, D., 2006. OpenID 2.0: a platform for user-centric identity management. *Proceedings of the second ACM workshop on Digital identity management*. Alexandria, VA, USA 30. Okt.-3. Nov. 2006. New York, NY: ACM, S. 11–16.

Scavo, T. & Cantor, S., 2005. Shibboleth Architecture – Technical Overview. *Working Draft: draft-mace-shibboleth-tech-overview-02*.

Stern, R. & McEwen, S., 2009. FITS – The File Information Tool Set. Poster. *4th International Conference on Open Repositories.* Atlanta, USA 18.-21. Mai 2009. Online*:* http://hdl.handle.net/1853/28508 [Zugriff am 27.01.2011].

TIB (Technische Informationsbibliothek) Hannover, FIZ Chemie Berlin & Universität Paderborn, 2010. *Konzeptstudie „Vernetzte Primärdaten-Infrastruktur für den Wissenschaftler-Arbeitsplatz in der Chemie“*. Online:

http://www.fiz-chemie.de/fileadmin/user_upload/PDF_DE/abstract_Konzeptstudie_Forschungsdaten_Chemie.pdf [Zugriff am 13.08.2011].

Treloar, A. Groenewegen, D. & Harboe-Ree, C., 2007. The Data Curation Continuum. *D-Lib Magazine*, 13(9/10). http://dx.doi.org/ doi:10.1045/september2007-treloar.