Distributed Research Data Management - Plädoyer für eine verteilte Forschungsdaten-Infrastruktur

Reiko Kaps[1]

1 Leibniz Universität Hannover

Zusammenfassung. Klassisches Forschungsdatenmanagement sieht die Forschenden eher als Kunden denn als Mitstreiter. Beratung und Angebote etwa zur Veröffentlichung von Forschungsdaten orientieren sich an Service-Konzepten, die Informationen zentral verteilen und vorhalten. Diese Einstellung spiegelt sich auch bei Forschungsdaten-Repositorien wider.

Diese Publikationsplattformen eignen sich damit zwar gut als Schaufenster, weniger oder oft gar nicht als Arbeits- und Austauschplattform für Forschungsdaten. Zudem sind Forschungsdaten-Repositorien für die dringendsten Probleme der Forschenden nur unzureichend gerüstet: Dazu zählen die Sicherung der Urheberschaft, die Wahrung der Unversehrtheit von veröffentlichten Daten sowie die dauerhafte Verknüpfung mit Metadaten und Lizenzinformationen. Herkömmliche IT-Dienste gewährleisten solche Zusicherungen nur innerhalb ihres Systems. Verlassen die veröffentlichten Daten dieses Refugium, können Informationen zu Urheberschaft und Lizenz in Gefahr geraten.

Jenseits des klassischen Client-Server-Ansatzes existieren andere Verfahren, um Daten und Dienste im Internet zu verbreiten und deren Authentizität sicherzustellen. Diese dezentralen Ansätze setzen auf Peer-to-Peer-Techniken und neuerdings auch auf Blockchain-Verfahren, die die mathematische Grundlage für Kryptowährungen bilden. Blockchain-Verfahren sichern Transaktionen, stellen "Einigkeit" unter den beteiligten Knoten her und eignen sich so für den sicheren, vertrauenswürdigen und nachvollziehbaren Austausch beliebiger Daten – also auch von Forschungsdaten.

Inzwischen stehen Anwendungen bereit, die Dateien dezentral verteilen (IPFS), Daten strukturiert ablegen (BigchainDB) und untrennbar mit Metainformationen zu Urheber, Lizenz etc. verbinden (Mediachain/IPDB). Diese Programme benötigen keine zentrale Instanz, sichern die Datenintegrität, verfolgen Änderungen und verteilen effektiv Datenbestände. Jeder kann damit nicht nur Daten anbieten und abrufen, sondern auch auf dieser Basis eigene Anwendungen entwickeln.

Das folgende Papier stellt diese Ansätze vor, zeigt deren Vorteile gegenüber klassischen Client-Server-Angeboten und skizziert, wie sie das Datenmanagement für veröffentlichte Forschungsdaten verbessern und vereinfachen können.

Datarefuge

Das Musterland des Forschungsdatenmanagement erlebt derzeit eine einzigartige Bewegung. Seit Monaten treffen sich fast jedes Wochenende ganze Scharen von Wissenschaftlern, Programmierern und Aktivisten in Universitätsbibliotheken, Hackerspaces und anderen Orten, um öffentlich zugängliche Forschungsdaten zu retten. Dabei handelt es sich um Daten aus der Klimaforschung öffentlicher Bundeseinrichungen wie der Umweltbehörde EPA und der NASA, die diese auf ihren eigenen Servern bislang zum Download anbieten.

Abbildung 1. Datarefuge auf Twitter (Screenshot 2017)

Diesen Daten droht durch den neuen US-Präsidenten Gefahr, denn bereits im Wahlkampf äußerte sich Donald Trump sehr ablehnend gegen die Forschung zum Klimawandel und deren Erkenntnisse[1]. Das noch unter seinem Vorgänger wichtige Thema verschwand sehr bald nach Trumps Amtsübernahme von einigen offiziellen Webseiten und ist nun nur noch im Web-Archiv des Weißen Hauses zu finden[2].

Seit der Wahl versuchen zahleiche US-Wissenschafler daher, diese drohenden Verluste abzuwenden: Dazu müssen sie die Daten mühsam von den Behörden-Webservern laden und auf externen, oft in Kanada stehenden Servern auslagern.

Da die Daten oft nicht einfach kopiert werden können, sondern sich hinter aufwendigen Webanwendungen verstecken, ist diese Rettungsaktion kein leichtes Unterfangen. Die an #Datarefuge Beteiligten müssen zu diesem Zweck Webseiten und Daten analysieren und Skripte für den Download und die Konvertierung programmieren.

Client-Server-Dilemma

Warum müssen die US-Wissenschalfter und -Aktivisten jedoch diesen Aufwand treiben? Neben den politischen Rahmenbedingungen drängt sich dabei eine grundsätzliche Frage auf: Liegt der eigentliche Grund für diesen massiven Rettungsaufwand womöglich in der Art, wie wissenschaft-

1 http://www.stern.de/politik/ausland/donald-trump--wissenschaftler-retten-klima-daten-vor-seinem-amtsantritt-7241850.html
2 http://www.zeit.de/politik/ausland/2017-01/donald-trump-website-weisses-haus-klimawandel

liche Einrichtungen in der Vergangenheit und aktuell ihre Forschungsdaten speichern und veröffentlichen?

Wissenschaftliche Einrichtungen veröffentlichen derzeit Forschungsdaten und deren Ergebnisse über Plattformen, die dem Client-Server-Modell entsprechen. Das heißt, ein oder auch mehrere zentrale Server verteilen auf ihnen vorgehaltene Daten an Interessenten (Clients). Diese Architektur ist vergleichsweise anfällig für Ausfälle (single point of failure). Wer dabei auf Redundanz und Verfügbarkeit wert legt, muss sich beide Punkte teuer erkaufen. Das erschwert die Skalierbarkeit, die ausschließlich der Server-Betreiber sicherstellen kann. Das gilt besonders dann, wenn der bereitgestellte Dienst tatsächlich erfolgreich ist: Steigt die Nachfrage nach diesen Daten, müssen sie an jeden Client einzeln ausgeliefert werden: Abhängig von den vorhandenen Netzwerkressourcen droht der Dienst an seinem eigenen Erfolg zu scheitern.

Gleichzeitig forciert der Client-Server-Ansatz Diensteinseln. Interoperabilität und leichter Datenaustausch mit anderen bleiben nachrangige Aufgaben oder sie müssen durch umfangreiche Regelwerke sichergestellt werden (RFC/IETF, ISO). Diese Diensteinseln orientieren sich zudem an kommerziellen Angeboten, behandeln ihre Nutzer fast immer als Kunden und betrachten ihre Daten als Ausstellungsgegenstand.

Auf dem Client-Server-Modell aufsetzende Dienste haben in der Vergangenheit massgeblich zum Wachstum des Internets beigetragen. Die steigende Verbreitung des Internets und neue Herausforderungen bei Datenhaltung und -verteilung lassen dieses Konzept jedoch an seine Grenzen stoßen.

Neue Herausforderungen

Datacenter- und Infrastruktur-Betreiber stehen vor der Aufgabe, immer größere Datensätze bereitzustellen und an immer mehr Teilnehmer zuverteilen. So hat sich der Datenverkehr am deutschen Internet-Knoten DE-CIX in den vergangenen 5 Jahren mehr als verdoppelt[3].

Während im kommerziellen Bereich dabei besonders Videodaten die Entwicklung vorantreiben, sind es in der Wissenschaft digitale Forschungsdaten: Zu den Messreihen in überschaubaren Textdateien haben sich längst Bilder, Videos, Simulationen, Visualisierungen und Social-Media-Logs gesellt, die den Umfang digitaler Forschungsdaten massiv nach oben treiben. Anders als bei anderen Daten unterliegen Forschungsdaten jedoch Ansprüchen, die weit über das Speichern und Verteilen hinausgehen.

Im Kern handelt es sich dabei um die Forderungen der öffentlichen Förderer, die Forschungsdaten für wenigstens 10 Jahre erhalten und sie für die Öffentlichkeit nutzbar machen wollen. Forschungsdaten müssen daher ausreichend dokumentiert sein sowie in Formaten vorliegen, deren Aufbau offengelegt ist. Außerdem muss ihre Herkunft gesichert sein und ihre Entstehungs- und Bearbeitungswege nachvollzogen werden können (Integrität, Versionierung, Verknüpfung). Ohne diese Randbedingungen sind veröffentlichte Forschungsdaten kaum oder nur mit hohen Aufwand durch Interessierte nutzbar – selbst wenn diese der derselben Fachdisziplin angehören.

3 https://www.de-cix.net/en/locations/germany/frankfurt/statistics

Auswege

Die skizzierten Probleme lassen sich bereits mit vorhandenen und erprobten Techniken lösen, die sowohl auf Peer-to-Peer-Prinzipien als auch das tradierte Client-Server-Modell setzen.

Unter dem Begriff Peer-to-Peer (P2P) versammeln sich Techniken, die ohne die zentralen Instanzen des Client-Server-Modells auskommen. In Netzwerken arbeiten solche Systeme gleichzeitig als Server und als Client, sodass sich ein Netz aus (grundsätzlich gleichberechtigten) Knoten aufspannt. Diese Knoten verteilen die für die Peer-to-Peer-Anwendung nötigen Informationen.

Eine der bekanntesten Peer-to-Peer-Anwendungen dürfte das im Jahr 2001 entwickelte Bittorrent-Protokoll sein, das große Dateien zuverlässig und effektiv verteilt. Im Unterschied zu den von Standardisierungsgremien gepflegte HTTP oder FTP nutzt Bittorrent auch die ansonsten ungenutzten Upload-Kapazitäten der jeweiligen Download-Knoten, sodass eine Datei nicht nur vom Anbieter selbst sondern auch von denjenigen verteilt wird, die diese Datei herunterladen[4]. Anhand von Prüfsummen stellt Bittorrent dabei sicher, dass die übertragene Datei dem Orginal entspricht. Dank dieser Fähigkeiten erlangte Bittorrent einerseits eine durchaus zweifelhafte Berühmtheit als Dateitauschbörse für digitale Medien wie Musik und Spielfilme, andererseits stellte das Protokoll damit auch sein Fähigkeiten als effektiver Dateiverteiler unter Beweis[5].

Für den Umgang mit Dateien und Inhalten steht inzwischen die Dateiversionierungssoftware Git bereit, die ebenfalls auf Peer-to-Peer-Techniken setzt. Versionsverwaltungen wie Git protokollieren Änderungen an Dateiinhalten, erlauben verteilte und nicht-lineare Arbeitsabläufe und gewährleisten damit Nachvollziehbarkeit. Viele Funktionen von Git lassen auch ohne permanente Internet-Verbindung einsetzen.

Eine weitere Technikentwicklung der vergangenen Jahre erlaubt eine lückenlose Buchführung von Aktionen: Die Blockchain ist eine Datenbank mit mathematischem Integritätsansatz. Sie speichert den Hashwertes (Integritätsgarant) eines Datensatzes im Hashwert des jeweils nachfolgenden. Die Technik ähnelt dem Journal in der Buchführung und stellt damit die Grundlage für Kryptowährungen bereit[6]. Mittels einer Blockchain lassen sich sowohl die Transaktionsicherheit als auch die Nachvollziehbarkeit in verteilten Systemen erheblich vereinfachen und verbessern.

IPFS

Jede der bereits genannten Peer-to-Peer-Techniken löst nur Teile der eingangs geschilderten Probleme. Allerdings steht seit dem Jahr 2014 das Interplanetary File System (IPFS) als quelloffene Implementierung bereit, die Web-, Bittorrent-, Git- und Blockchain-Funktionen in einem verteilten Dateisystem vereint[7]. Das IPFS-Konzept und die in Go geschriebene Referenzimplementierung stammen von Juan Benet[8].

IPFS stellt ein vollständig verteiltes Netzwerk-Dateisystem bereit. Darin arbeiten alle Knoten sowohl als Server als auch als Client und jeder Knoten ist mit jedem anderen verbunden (siehe

4 https://de.wikipedia.org/wiki/BitTorrent
5 https://torrentfreak.com/bittorrent-dominates-internet-traffic-070901/
6 https://de.wikipedia.org/wiki/Buchf%C3%BChrung#Journal_.28Grundbuch.29
7 https://ipfs.io/
8 https://github.com/ipfs/papers/raw/master/ipfs-cap2pfs/ipfs-p2p-file-system.pdf, https://github.com/ipfs/ipfs, https://twitter.com/juanbenet

Abbildung 1). IPFS adressiert Inhalte über Hashes, macht Dateien über lesbare Namen auffindbar (IPNS) und dedupliziert Dateien innerhalb seines Dateisystems. Abgerufene Inhalte anderer IPFS-Knoten hält IPFS in einem lokalen Cache vor - bei Bedarf sogar dauerhaft (Pinning). Lädt IPFS fremde Inhalte, versucht es sie von möglichst vielen und nahe gelegenen Knoten abzuholen. Jeder angefragte Knoten liefert dabei Teilstücke einer Datei an den Anfragenden aus. Ähnlich wie Git protokolliert IPFS Änderungen an seinen Dateien respektive Inhalten.

Research Data Federation

IPFS stellt damit Funktionen bereit, aus denen Inhalteproduzenten und -anbieter, Forschende, Infrastrukturbetreiber und Bibliotheken sowie digitale Archive Profit ziehen können. Mit IPFS und verwandten Techniken lässt sich eine globale Infrastruktur für Forschungsdaten aufbauen, bei der jeder Nutzer zum Teil der Infrastruktur wird. Anders als Client-Server-Konzepte zeichnet sie sich durch eine hohe Zensurresistenz und Ausfallsicherheit aus. Sie nutzt Netzwerkressourcen optimal, verbessert die Netzsicherheit und erlaubt eigene Anwendungen, die auf IPFS aufsetzen.

Beispielsweise kann IPFS dabei helfen, bessere und günstigere Lösungen für das Problem der Zweitkopie zu finden. Forschungsdaten-Repositorien und andere digitale Archive verhindern mit einer Zweitkopie die ungewollte Alterung und Verfälschung ihrer Archivobjekte, die etwa durch Bitfehler verursacht werden können. Diese Maßnahme verdoppelt jedoch die Kosten für jede gespeicherte Datei, denn es sind dafür zusätzliche Speichermedien und Standorte nötig. Gerade für kleinere wissenschaftliche Einrichtungen dürfte der Betrieb mehrerer Systeme an unterschiedlichen Standorten ein nur schwer lösbares Problem sein.

In einer auf IPFS aufsetzenden Forschungsdaten-Infrastruktur können jedoch andere Teilnehmer die Aufbewahrung der Zweitkopie über das beschriebene Pinning übernehmen. Das eröffnet große Freiheiten beim Speicherort und kann Kosten senken. Diese Zweitkopie-Delegation lässt sich etwa mittels des Gegenseitigkeitsprinzips (Peering) oder über Mietmodelle vertraglich regeln.

Testbed: Call for Participation

Angesichts der genannten Argumente für ein verteiltes Forschungsdaten-Dateisystem und der Notwendigkeit Forschungsdaten effizient vorzuhalten und zu verteilen, schlagen wir einen Feldversuch in Form eines Testbeds vor: Dieses Experiment soll IPFS und ähnliche Techniken auf ihre Praxistauglichkeit untersuchen und Möglichkeiten der Zusammenarbeit mittels dieser Techniken erproben.

Wir wollen dabei sowohl Anwendungen testen, die Forschende direkt an die beschriebene Infrastruktur anbinden und sich damit besser in den wissenschaftlichen Workflow einfügen, als auch Infrastruktur-Konzepte erproben, mit denen Wissenschaftsorganisationen Forschungsdaten leicht und kostengünstig aufbewahren und verteilen können. Zu diesen Punkten können Interessenten jederzeit weitere Themen beitragen.

Dank der Peer-to-Peer-Struktur liegen die Hürden für die Teilnahme niedrig: Neben den klassischen Rechenzentren und Diensteanbietern wie Bibliotheken kann jeder mitmachen, der einen Rechner im Internet betreibt. Das sind im Idealfall ständig laufende Server, können aber auch PCs

und Notebooks sein, die nur temporär arbeiten. Darüber hinaus benötigen die Teilnehmer jedoch den Willen, sich mit den Interna von Peer-to-Peer-Techniken wie IPFS auseinanderzusetzen und dieses Wissen mit anderen zu teilen.
Weitere Fragen beantwortet der Autor gern per E-Mail oder Twitter[9].

9 E-Mail: kaps@luis.uni-hannover.de; Twitter: https://twitter.com/reik_kaps