Next Geberation Repositories: Wie kann die Interoperabilität hergestellt werden?

Bei der Weiterentwicklung von Repositorien (Next Generation Repositories[1]) spielt die Interoperabilität eine gewichtige Rolle. Diese ist Grundlage, um automatisierte maschinelle Methoden, wie Text- und Datamining, auf den Inhalt von verteilten Repositorien anzuwenden. Für die Bereitstellung und das Abrufen von Metadaten von Einträgen in Repositorien hat sich die OAI-PMH[2] Schnittstelle als Standard mit großem Erfolg etabliert und ist in allen gängigen Softwarelösungen implementiert. Auf der einen Seite hat man hier eine stabile und lange erprobte Schnittstelle, auf der anderen Seite entwickelt sich die Technologie weiter und Daten werden in schneller Zeit, größeren Umfang und im Volltext benötigt. Hier bietet sich als zusätzliche Technologie ResourceSync[3] an. Dabei wird der Fokus auf eine einzelne Ressource (Publikation) gelegt. Um diese eindeutig zu identifizieren, dienen Sitemaps[4]. Sitemaps sind eine Standardtechnologie für Webseiten, um Crawlern die Struktur der Seite mitzuteilen. Dabei werden in einem xml-Format die URLs einer Website mit zusätzlichen Daten wie etwa der letzten Änderung der Resource zur Verfügung gestellt. Dies wird nun um entsprechende Namespaces (rs:) bestehend aus zusätzlichen Metadaten (rs:md) und Verlinkungen (rs:ln) erweitert. Aus diesem Vorgehen ergeben sich unterschiedliche Einsatzmöglichkeiten.
Bei den PULL Szenarien holt sich das Synchronisationsziel die Daten von den Datenlieferanten ab. Hier unterscheidet man folgende Anwendungsfälle, welche in den Metadaten des jeweiligen Sitemap-Files angegeben werden. Man kann sich die gesamten Ressourcen auflisten lassen (capability=“resourcelist“). Dies ist v.a. für den Beginn einer Synchronisation notwendig, aber auch für eine Überprüfung im laufenden Betrieb. Außerdem besteht die Möglichkeit, sich den Link zu einem zip-File liefern zu lassen (capability=“resourcedump“). Es enthält die jeweiligen Bitstream-Dateien und ein xml-File (manifest.xml) zur Beschreibung dieser. Weiter kann man sich Änderungen einzelner Dateien anzeigen lassen (capability=“changelist“). Hier werden für ein angegebenes Zeitintervall die Metadaten der Änderungen in einer standardisierten Form angegeben. Analog zu einem ResourceDump ist es möglich, den Link zu einem zip-File der Änderungen, aufgeteilt in ein manifest.xml-File und die Bitstream-Dateien, zu erhalten (capability=“changedump“).
Bei den PUSH-Szenarien informiert der Datenlieferant das Synchronisationsziel über Änderungen. Hierbei sollen ständige PULL-Nachfragen über Veränderungen sowie wiederholte Anfragen nach ResourceLists und der Beschreibung der ResourceSync-Schnittstelle (ResourceSyncDescription) vermieden werden. Zusätzlich wird hier die Latenzzeit für die Einspielung von Änderungen deutlich verringert. Grundsätzlich unterscheidet man bei den PUSH-Benachrichtigungen zwischen Änderungen von einzelnen Ressourcen und Änderungen der Abrufmöglichkeiten. Diese werden mittels WebSub[5] implementiert.
Ergänzt wird das Ganze noch mit einem Discovery-Mechanismus, um die einzelnen Sitemap-Files zu finden, und einer Archivierungsstruktur.
Insgesamt stellt das Framework eine ideale Ergänzung der bestehenden OAI-Technologie zur Synchronisierung von Repositorien dar.

1 Vgl. Confederation of Open Access Repositories (COAR) Next Generation Repositories, https://ngr.coar-repositories.org/
2 Siehe https://www.openarchives.org/
3 Siehe http://www.openarchives.org/rs/toc
4 Siehe https://www.sitemaps.org/
5 Siehe https://www.w3.org/TR/websub/