



Entwicklung der RePlay-DH-Werkzeuge

Motivation

- Komplexe Forschungsprozesse werden häufig nur skizzenhaft dokumentiert.
- Bisher war ein hoher Mehraufwand für eine nachhaltige Dokumentation erforderlich.
- Dokumentation der Prozesse findet erst, wenn überhaupt, nach dem Abschluss eines Vorhabens statt.
- Versionierungslösungen aus dem Bereich der Software-Entwicklung erlauben zu viele Freiheiten für den Nutzer und sind komplex in der Handhabung.

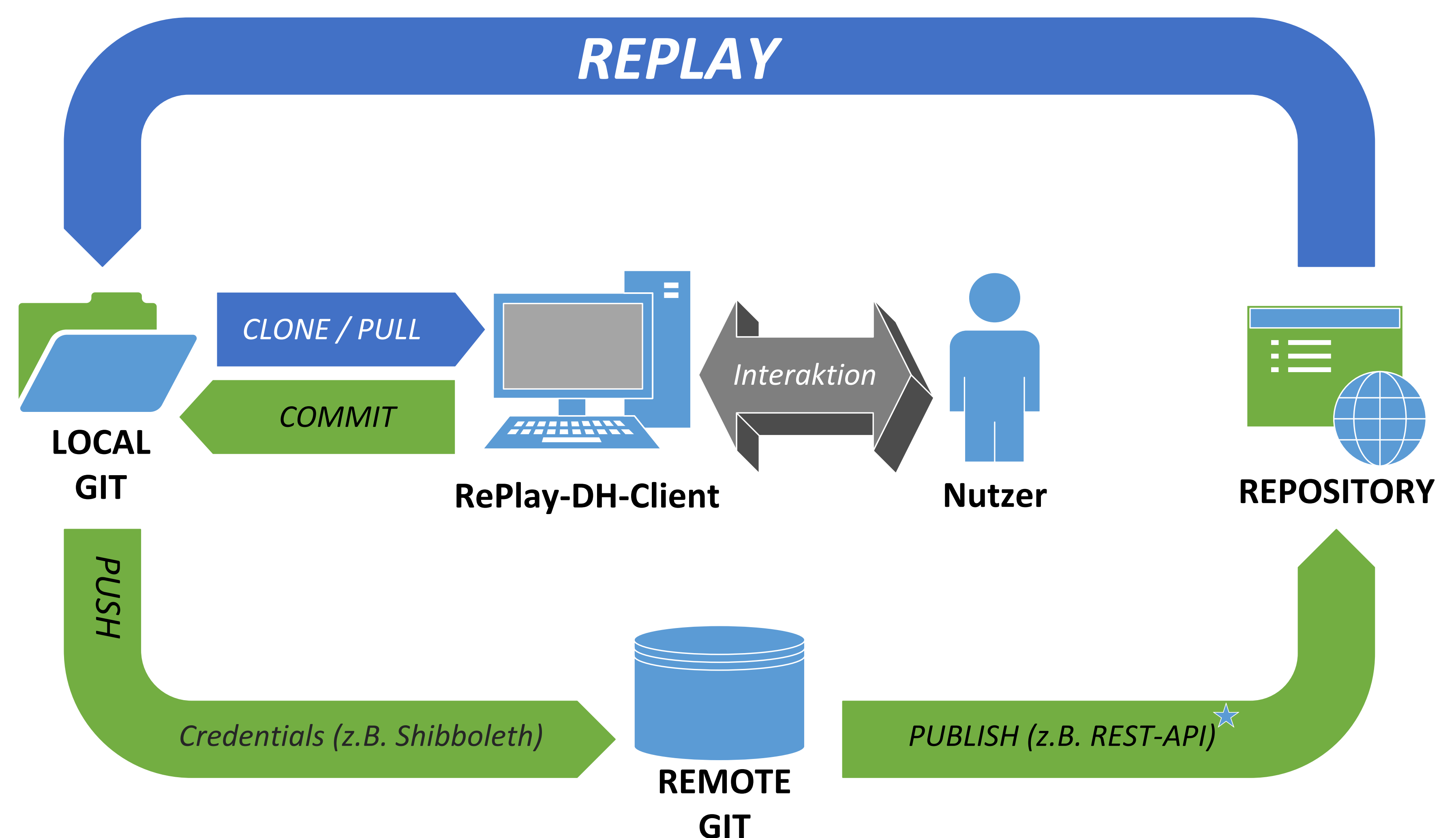
Ziel

Dokumentation von Zwischenständen bereits während des wissenschaftlichen Arbeitsprozesses mit der Option zur Archivierung, Veröffentlichung und Nachnutzung von Forschungsdaten am Beispiel der Digital Humanities.

RePlay-DH Client

Der RePlay-DH-Client ist das Herzstück der RePlay-DH-Idee. Er soll für den Nutzer eine einfache Möglichkeit bieten, die Vorteile komplexer Versionierungssoftware im DH-Kontext nutzen zu können:

- RePlay verfolgt den Ansatz, dem Nutzer nur die für seine Zwecke notwendigen Funktionen anzubieten, um so die Komplexität zu reduzieren und eine einfache grafische Aufbereitung anzubieten (GUI).
- Der Client wird während des Arbeitsprozesses selbst meist für den Nutzer unsichtbar sein. Wenn eine für die Archivierung, Veröffentlichung oder Nachnutzung relevante Interaktion ansteht, die nicht automatisiert erfasst werden kann, bekommt der Nutzer eine Benachrichtigung.
- Der Nutzer muss sich über Archivierung und Veröffentlichung kaum Gedanken machen: Die Daten werden bereits so erfasst dass eine Archivierung, Veröffentlichung und Nachnutzung einfach möglich ist.



Local Git

RePlay-DH-Werkzeuge und Git-Implementierung

Der in Java umgesetzte und somit plattformunabhängige Client erlaubt die Nutzung der RePlay-DH-Werkzeuge ohne gesonderte Software-Installation, mit Ausnahme einer Java-Laufzeitumgebung.

Abbildung des Workflows

Damit der Arbeitsprozess abgebildet werden kann, wird für jedes eigenständige RePlay-DH-Projekt vom Client ein (lokales) Git-Repository angelegt. Je nachdem, welches Projekt vom Wissenschaftler gerade im Client aktiviert ist, werden die Aktivitäten im entsprechenden Git-Ordner überwacht.

Mehrwehrt für den Wissenschaftler

Die Funktion der Verzweigung im Git (Branches), ermöglicht es dem Wissenschaftler nach gescheiterten Versuchen einfach zu einem kritischen Punkt seiner Forschung zurückzukehren und Alternativen zu verfolgen. Der Fehlversuch bleibt – wenn gewünscht – dokumentiert.

Remote Git

Anbindung an externe Server (Remote Git)

Der Nutzer kann im Client eine Anbindung zu einem Remote Git (z.B. Landesdienst) definieren. Nach erfolgreicher Authentifizierung können Nutzer Inhalte im Remote Git ablegen und teilen.

Mehrwert

- **BACKUP:** Das Remote Git dient als geeignetes Backup der Forschungsdaten innerhalb einer vertrauenswürdigen Einrichtung.
- **ARCHIV:** Das Remote Git kann – je nach Betreiber – als Forschungsdaten-Archiv im Sinne der GWP-Aufbewahrungsfristen der DFG dienen.
- **KOLLABORATIVES ARBEITEN:** Im Remote Git kann z.B. bestimmten eingeschränkten Nutzergruppen Zugriff auf bestimmte Repositorien gewährt werden. So kann im Forschungsprojekt gemeinsam an Inhalten gearbeitet werden.

Repository

Anbindung an Veröffentlichungsplattform (Repository)

Häufig sind Repositorien mit Schnittstellen zur automatisierten Datenablieferung versehen (z.B. SWORD, REST).

- RePlay-DH hält die im Projekt SARA verfolgten Ziele zur Anbindung eines Git-Repositorys an eine entsprechende Veröffentlichungs-Plattform im Blick.
- Die Nutzer sollen auswählen können, welche Inhalte veröffentlicht werden: z.B. „komplettes Repo“ (RePlay) / nur Commits, keine Dateien (Workflow) / nur der letzte Commit (Ergebnis).

Veröffentlichung von Forschungsdaten

Neben der Idee der Reproduzierbarkeit und Zitierbarkeit wissenschaftlicher Ergebnisse, ist ein zentraler Aspekt der RePlay-Idee die möglichst einfache Nachnutzung von veröffentlichten (Teil-) Ergebnissen. Durch die Anbindung an ein Repository kann der Wissenschaftler direkt innerhalb des RePlay-DH-Clients eine Veröffentlichung anstoßen.