

2. Generierung und Bereitstellung von semantischen Thumbnails aus heterogenen Daten für Project-Zoom (Dominic Bräunlein)

Wer Daten speichert, möchte diese Informationen zu einem späteren Zeitpunkt zur Verfügung haben. Der Speicherplatz, der dabei zur Verfügung steht, steigt kontinuierlich. Einblicke in den dSchool Prozess zeigten, dass dieser Speicherplatz oft mit Daten gefüllt wird, die weder vorgefiltert noch einheitlich strukturiert wurden. So verwundert es die wenigsten, dass immer größere Mengen der gespeicherten Daten redundant oder unwichtig für den späteren Projektverlauf sind. Die anschließende Suche nach den relevanten Daten in dieser Datenflut stellt eine Herausforderung für Programmierer wie Nutzer von Project-Zoom dar.

Eine zentrale Rolle spielt hier die Miniaturansicht der Daten. Je mehr Informationen einer einzelnen Datei zu erkennen sind, umso besser kann der Nutzer den Inhalt der Datei einschätzen. Im Gegensatz dazu steht die Forderung, so wenig Bildschirmfläche wie möglich für ein einzelnes Bild zu benötigen, um so dem Nutzer zu ermöglichen schnell einen Überblick über alle Daten zu bekommen. Ein Ansatz der beide Anforderungen adressiert, ist das Erstellen von Schlagwortwolken von Textdokumenten. Steht nur sehr wenig Bildschirmfläche zur Verfügung, muss sich die Schlagwortwolke auf nur wenige Wörter beschränken, wohingegen bei einer großen Bildschirmfläche sehr viele Wörter dargestellt werden können.

In Project-Zoom sammelt ein Aggregator im Backend heterogene Daten aus verschiedenen Datenquellen und stellt sie dem Nutzer zur Verfügung. Der Nutzer soll die für ihn relevanten Daten schnell finden und miteinander in Verbindung bringen können. Um den Zeitaufwand der Suche nach den Daten zu minimieren, werden semantische Thumbnails generiert. Semantisch bedeutet zum einen, dass abhängig vom Dateitypen verschiedene Algorithmen herangezogen werden, um intelligente Vorschaubilder zu erzeugen. Zum anderen bedeutet es, dass pro Datei mehrere Vorschaubilder in verschiedenen Auflösungen erstellt werden, um später die vorhandene Bildschirmgröße mit möglichst viel Inhalt befüllen zu können. Die Vorschaubilder werden direkt nach dem Aggregieren einer Datei erstellt, sodass diese bei einer Nutzeranfrage ohne Zeitverzögerung dargestellt werden können.

Verschiedene Medientypen erfordern verschiedene Algorithmen. Das Verfahren, das verwendet wird, um aus Textdokumenten Vorschaubilder zu erzeugen, unterscheidet sich zum Beispiel erheblich von dem für Fotos oder Filmsequenzen. Für Project-Zoom ist es geplant, die Medientypen Bild, Video und Textdokument zu unterstützen. Für die Verarbeitung eines Medientyps wird hierzu jeweils ein erweiterbares und autonomes Plugin entwickelt und an das Eventsystem angehängt. Jedes dieser drei Plugins hat wiederum selbst Plugins, die die unterstützten Dateiformate eines Medientypen in ein jeweils gemeinsames Format bringen. Wurden die Daten in dieses proprietäre und Medientyp-spezifische Format gebracht, können unabhängig vom ursprünglichen Dateiformat die Miniaturansichten mit dem Medientyp-spezifischen Algorithmus erzeugt werden (siehe Abbildung 1).

Nachdem die intelligenten Vorschaubilder erstellt wurden, ist es wichtig dem Nutzer einen

komfortablen Finder für alle aggregierten Daten zur Verfügung zu stellen. Gruppierungen nach beispielsweise Quelle oder Medientyp und Sortierungen wie Dateiname oder Erstellungsdatum sollen beim Finden der Artefakte helfen.

Weitere Herausforderungen birgt das Bereitstellen solcher semantischen Thumbnails in einer Webapplikation wie Project-Zoom. Die erste Herausforderung sind die verschiedenen Technologien wie SVG und HTML, die im Frontend benutzt werden. Trotz der unterschiedlichen Modelle hinter den Technologien wird ein einheitliches Verhalten bei der Darstellung der Thumbnails erwartet. Die zweite Herausforderung ist, dass der Nutzer unter anderen im Process Overview stufenlos zoomen kann. Zoomt ein Nutzer stärker in den Graphen hinein, können auch mehr Inhalte der Datei dargestellt werden, da der verfügbare Platz auf dem Bildschirm für eine Miniaturansicht größer ist. Diese Problemstellungen werden mit einer Thumbnailkomponente adressiert, die nur noch in den unterschiedlichen Views eingebunden werden muss. Einmal eingebunden, sorgt sie dafür, dass der Nutzer automatisch und immer an den verfügbaren Bildschirmplatz angepasst das Thumbnail in der semantisch richtigen Zoomstufe dargestellt bekommt.

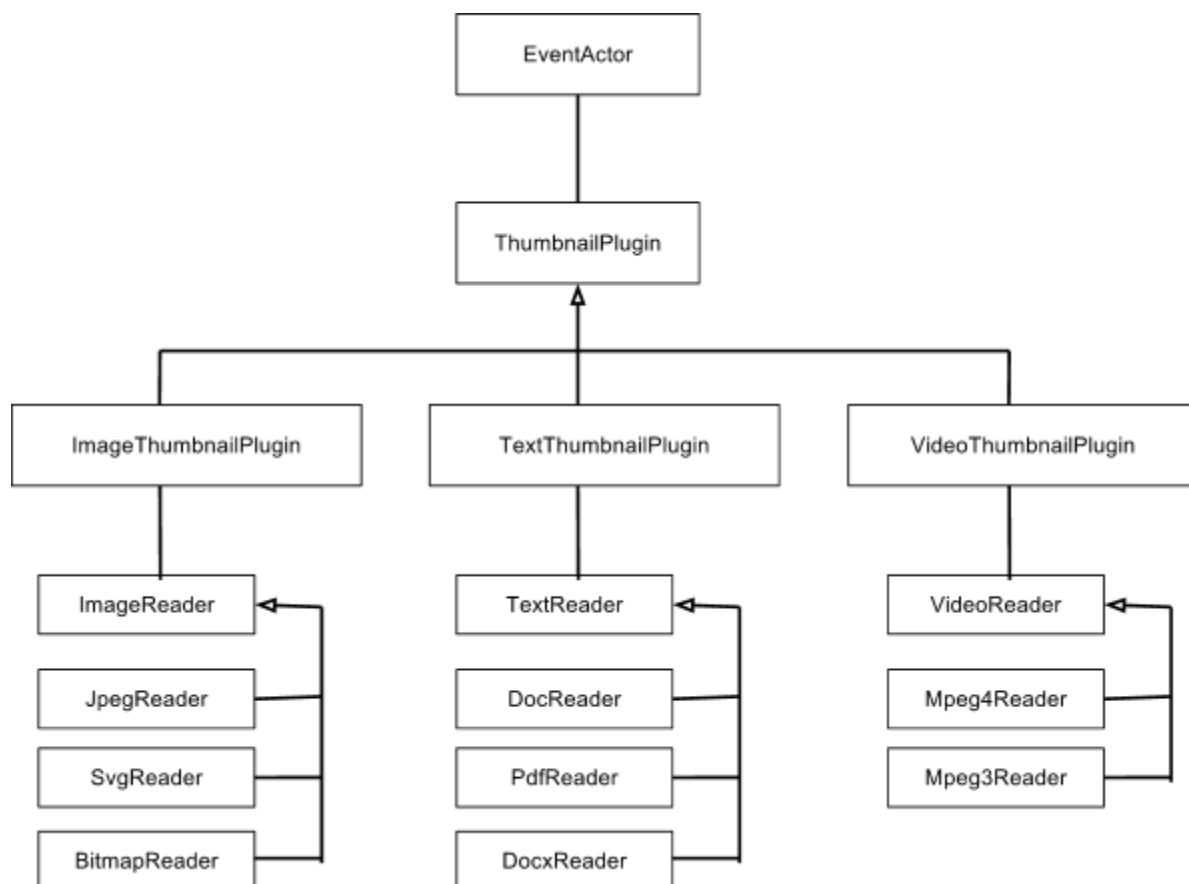


Abbildung 1: Beispielhafte Architektur der Thumbnail-Plugins