

1574. Msj. Lotvcki. Venet.
MORALIA.

abr.

RZECZY, DO OBYCZAJOW,

NAVK, Y PRZESTROG W KAŻ

dym stanie żywota ludzkiego „

z Łacińskich, y z Polskich Przypowie „

ści, Oyczyстым krako napisane Wierszem

CZEŚĆ PIERWSZA

XIEGI PIERWSZEY.

Przekładania z Greckiego Języka **Erasma Roterodama.**

Joanna Hataczkiewicz

E^{dycje} pilotażowe w Laboratorium Edytorstwa Cyfrowego Uniwersytetu Jagiellońskiego

Uniwersytet Jagielloński w Krakowie, kontakt: j.hataczkiewicz@uj.edu.pl,
ORCID ID: 0000-0001-7143-789X

Laboratorium Edytorstwa Cyfrowego (LabEdyt) to interdyscyplinarny zespół badawczy działający w Jagiellońskim Centrum Humanistyki Cyfrowej w ramach projektu flagowego Digital Humanities Lab¹. Należą do niego zarówno doświadczeni edytorzy – wydawcy naukowi tradycyjnych edycji drukowanych, jak i historycy, językoznawcy czy informatycy. W poszczególne projekty angażują się ponadto studenci zainteresowani humanistyką cyfrową. Ważną rolę odgrywają też wspierający zespół pod względem technicznym architektki otwartego systemu TEI Publisher oraz członkowie społeczności e-editiones².

Głównym powodem, dla którego powstał LabEdyt, było pragnienie rzeczywistego, praktycznego rozwijania edytorstwa naukowego w paradygmacie cyfrowym. Choć od wielu lat w środowisku polskich filologów i wydawców pojawia się postulat tworzenia cyfrowych edycji dzieł literackich oraz źródeł historycznych, to wciąż, jak sądzę, pozostaje on w sferze deklaratywnej, koncepcyjnej, mimo coraz liczniej pojawiających się w internecie wyjątków³. Wydaje się zatem, że na przeszkodzie do

krzewienia cyfrowego edytorstwa naukowego nie stoi brak chęci naukowców do realizowania tego typu projektów, lecz raczej wciąż jeszcze niedobór odpowiednich kompetencji cyfrowych, niezbędnych do pracy w środowisku nieco innym niż klasyczny edytor tekstu. Potrzeba czasu, by wykształcić specjalistę swobodnie poruszającego się w gąszczu zagadnień teoretycznych i praktycznych edytorstwa naukowego, a przy tym orientującego się w problemach *stricte* informatycznych. Czas ten jest mierzony przede wszystkim liczbą podjętych prób edycji, gdyż doświadczenia zdobytego w trakcie pracy nad wydawanym tekstem, owego „misterium wtajemniczenia w arkana zawodu”, nie sposób zrekomensować lekturą podręczników. Ci jednak, którzy od lat z sukcesami trudnią się edytorstwem naukowym w Polsce, na ogół nie mają czasu na systematyczną, żmudną i samodzielną naukę kodowania (a to przecież zaledwie wstęp do zagadnienia). Wielu z tych, którzy nie tylko kodują, lecz także programują czy pracują z modelami sztucznej inteligencji, nie starcza z kolei humanistycznej erudycji niezbędnej do pracy z wydawanym tekstem. Być może więc właśnie obserwujemy, jak badaczom z młodego pokolenia, tzw. *digital natives*, którzy z tytułu urodzenia we właściwym momencie mieli niejako mimochodem ucyfrowić edytorstwo naukowe, nieoczekiwanie umyka spod nóg solidny grunt tradycji. Ta sytuacja pokazuje, jak ważna, ale też trudna jest systematyczna, interdyscyplinarna praca zespołowa nad cyfrową edycją naukową⁴.

Działalność LabEdytu skupia się przede wszystkim na praktyce e-edytorstwa – z całym potencjalnie negatywnym bagażem tej działalności, niewykluczającym porażek i błędów, będących wszakże naturalnym elementem procesu uczenia się. Intencją członków zespołu jest poszukiwanie metod efektywnego tworzenia edycji naukowych udostępnianych online, a dzięki temu – wypracowanie elastycznych, uniwersalnych schematów pracy, które można by zawrzeć w katalogu dobrych praktyk oraz w otwartych repozytoriach Git. Aby to osiągnąć, Laboratorium równolegle pracuje nad kilkoma edycjami pilotażowymi, spośród których każda różni się pewną istotną cechą, np. sposobem pozyskania i organizacji danych, co w efekcie przekłada się na odmienny sposób pracy z materiałem. To oczywiste, że tak powstałe wydania nie od razu objawiają się w swej idealnej, wzorcowej formie – takiej zresztą w edytorstwie nigdy nie było. Ważniejsze jednak, że naukowo opracowane teksty dzieł zaistnieją w Sieci, a przez to staną się punktem odniesienia, tematem do dyskusji. Już sam proces – od pliku wyjściowego do strony internetowej – jest wartościową lekcją dla zespołu.

Należy podkreślić, że Laboratorium Edytorstwa Cyfrowego nie aspiruje do stworzenia nowego dużego narzędzia czy

oprogramowania. Takie zadanie, jeśli traktować je poważnie, wymagałoby zatrudnienia wielu ekspertów i specjalistów, zapewne z kilku ośrodków, a przede wszystkim jednoznacznej odpowiedzi na pytanie o model finansowania kosztownej infrastruktury w dłuższej perspektywie – horyzont wyznaczony projektem flagowym nie sięga wystarczająco daleko. Częściową odpowiedzią na wskazane problemy wydaje się idea otwartego kodu źródłowego, dzięki której wokół konkretnych narzędzi mogą gromadzić się społeczności zainteresowane ich ulepszaniem. LabEdyt korzysta zatem z już istniejących programów i aplikacji, w większości otwartoźródłowych, testuje ich użyteczność w polskojęzycznych realiach, a pośrednio stara się wpływać na ich rozwój, proponując na forum międzynarodowym takie innowacje, które wydają się korzystne z perspektywy edytorów naukowych⁵. Pierwsze efekty są już widoczne: członkowie zespołu przyczynili się do rozwoju narzędzi Jinks oraz JinTap w ramach środowiska TEI Publishera. W miarę potrzeb i możliwości powstają też w LabEdycie skrypty oraz modele AI nadające się do wykorzystania w pracy z innymi tekstami. Mimo to nie ulega wątpliwości, że nadal nie istnieją wygodne w obsłudze narzędzia do kodowania i anotowania (komentowania) tekstów – zwłaszcza takich utworów, które zachowały się w wielu edytorsko ważnych przekazach, istotnych dla budowy aparatu krytycznego. Oxygen XML Editor, czyli najpopularniejszy program do tworzenia i edycji plików XML będących podstawą e-edycji, chociaż w porozumieniu z TEI Consortium wspiera obsługę znaczników TEI, a także zawiera wiele użytecznych funkcji, z pewnością nie posiada interfejsu przyjaznego użytkownikowi, którym jest nie zawodowy informatyk, lecz edytor literaturoznawca aspirujący zarazem do miana humanisty cyfrowego⁶.

Przedstawione w tym artykule edycje będą sukcesywnie udostępniane w Jagiellońskiej Platformie Cyfrowej (JPC) – serwisie internetowym agregującym projekty humanistyczne o charakterze cyfrowym realizowane w Uniwersytecie Jagiellońskim⁷.

Od statycznej strony internetowej do edycji cyfrowej: *Neolatina Sarmatica*

Wskreszenie cyfrowej biblioteki literatury staropolskiej i nowołacińskiej *Neolatina Sarmatica*⁸ było pierwszym zadaniem, jakie wyznaczył sobie zespół LabEdytu. Pierwotna wersja serwisu powstała w latach 2006–2011 na Wydziale

Polonistyki UJ w Pracowni Literatury Renesansu pod kierunkiem Grażyny Urban-Godziek. Ówczesnie realizowany projekt miał na celu opublikowanie edycji krytycznej dzieł łacińskich Jana Kochanowskiego wraz z tłumaczeniami – do tamtego czasu wydanych tylko raz: w 1884 roku, bez komentarza. Obok twórczości Kochanowskiego na starej stronie *Neolatiny* pojawiły się także skany dzieł innych autorów piszących po łacinie, m.in. Jana Dantyszka, Klemensa Janicjusza, Wawrzyńca Korwina, Andrzeja Krzyckiego, Piotra Roizjusza i Macieja Kazimierza Sarbiewskiego – z zamiarem późniejszego uzupełnienia ich o transkrypcję, tłumaczenie oraz komentarz.

Serwis opublikowany w subdomenie Biblioteki Jagiellońskiej został zbudowany z użyciem frameworku PHP Symfony. Abstrahując od нефunkcjonalnego, przestarzałego interfejsu witryny oraz wielu błędów technicznych (wynikających zapewne z upływu czasu, nie zaś z pierwotnych niedociągnięć), największym problemem edycji było formatowanie tekstu, który był pozbawiony struktury XML-owej, tj. oznakowania chociażby numeracji elegii łacińskich czy tytułów ód Jana Kochanowskiego, przez co z perspektywy komputera – ale także badacza chcącego wykorzystać możliwości cyfrowego przetwarzania danych tekstowych – był nieomal tożsamy z abstrakcyjnym ciągiem znaków⁹. Ponieważ stara wersja *Neolatiny* nie zapewniała żadnych cyfrowych przewag w rodzaju wygodnej wyszukiwarki, interaktywnych przypisów czy możliwości równoległej lektury transkrypcji, faksymile i tłumaczenia, w istocie była ona mniej użyteczna od tradycyjnego wydania książkowego. Odnosząc się do kryterium niedrukowalności prawdziwych edycji elektronicznych¹⁰, można stwierdzić, że *Neolatina* wprawdzie nie dawała się przenieść na papier, ale nie dlatego, że miała szczególne funkcjonalności, lecz dlatego, że była przygotowana w sposób niepozwalający jej wydrukować.

Zespół programistów działających w LabEdycie, posługując się techniką web scrapingu, pozyskał tekst z witryny, a następnie przekonwertował go do plików TEI XML. Zadanie to udało się zautomatyzować tylko częściowo, dlatego ważną rolę na obecnym etapie prac nad edycją odgrywają ręczne poprawianie struktury danych oraz korekta – wykonywane przez współpracowników projektu. Równolegle budowany jest interaktywny glosariusz zawierający krótkie objaśnienia postaci historycznych i mitologicznych występujących w utworach opublikowanych w *Neolatinie*.

Uwagi wymaga także powiązanie transkrypcji tekstów ze skanami podstaw wydania. Poprzedni serwis przechowywał

grafiki w osobnym katalogu plików strony, czyli wyłącznie na własne potrzeby magazynował duże zasoby danych. Współcześnie stosuje się inne rozwiązanie, które polega na tym, że wysokiej jakości skany pochodzące z serwerów bibliotek cyfrowych – np. z rodzimej Cyfrowej Biblioteki Narodowej Polona – są osadzone w witrynach czy aplikacjach za pomocą manifestów IIIF (*International Image Interoperability Framework*). Jeśli z kolei to my dysponowalibyśmy obrazami nieopublikowanymi nigdzie indziej w internecie, również byłibyśmy w stanie je udostępniać innym projektom z własnego serwera IIIF. Co jednak istotniejsze, standard ten umożliwia pracę z koordynatami, a zatem np. skany całych stron mogą być – w zależności od potrzeby – wyświetlane w edycji jako wycinki ograniczone przez koordynaty. Istnieje także możliwość oznaczania i anotowania fragmentów skanów, chociażby linijek tekstu, co doskonale sprawdza się w trudno czytelnych manuskryptach. W *Neolatinie* podgląd faksymile (aktualnie obsługiwany przez przeglądarkę OpenSeadragon) z początku nie zawsze działał właściwie, a problem ten wynikał z konieczności uzupełnienia brakujących powiązań z manifestami IIIF.

**Tworzenie objaśnień
do edycji naukowej za pomocą
klasycznego edytora kodu
nie jest zadaniem komfortowym**

Ponieważ pliki XML nie definiują wyglądu ani funkcjonalności edycji cyfrowej¹¹, muszą zostać odpowiednio przetworzone, aby ich zawartość była wyświetlana

w przeglądarce internetowej w formie czytelnej i przyjaznej dla użytkownika. Procedura transformacji XML do HTML może przebiegać np. z wykorzystaniem scenariuszy XSLT (*Extensible Stylesheet Language Transformations*) dostępnych w programie Oxygen XML Editor. Alternatywną ścieżką jest skorzystanie z narzędzia w rodzaju TEI Boilerplate (dla niedużych projektów cyfrowych, gdzie najważniejsze jest sprawne wygenerowanie minimalistycznych plików HTML), EVT czy TEI Publisher. Ostatnie z wymienionych rozwiązań zastosowane w *Neolatinie* dają największe możliwości wizualizacji danych oraz ich porządkowania, jednocześnie jednak – mimo szczerych chęci twórców – pozostaje skomplikowane dla niezaawansowanego użytkownika, który chciałby krok po kroku stworzyć własną edycję cyfrową, zaczynając od podstaw, a kończąc na publikacji w internecie¹².

Gdy nowa odsłona *Neolatiny* powstawała w środowisku TEI Publisher, twórcy tego oprogramowania pracowali nad jego dziewiątą wersją. Co istotne, w aktualizacji pojawił się ulepszony wizualny anotator – narzędzie do komentowania tekstu wyręczające badacza z konieczności wstawiania znaczników komentarza (np. <note>, <persName>, <orgName>,

<placeName>, <choice> itd.) bezpośrednio w plikach XML¹³. Bez wątpienia tworzenie objaśnień do edycji naukowej za pomocą klasycznego edytora kodu nie jest zadaniem komfortowym, zwłaszcza jeśli pewne komentarze mają być wykorzystane więcej niż raz. Najlepiej bowiem, aby wszystkie „przypisy”, uszeregowane w listę, znajdowały się w jednym miejscu (na początku lub na końcu dokumentu, względnie w osobnym pliku – rejestrze), a każdy z osobna miał własny niepowtarzalny identyfikator, który następnie będzie wywoływany obok właściwego fragmentu przez atrybut @corresp. Trudno wyobrazić sobie, że w przypadku edycji dużych dzieł można byłoby, postępując w ten sposób, zapanować nad lawinowo przyrastającym komentarzem – dlatego właśnie w Laboratorium Edytorstwa Cyfrowego projekty takie jak *Neolatina* korzystają ze swego rodzaju „protez” umożliwiających badaczom wygodne dodawanie przypisów przez formularz, a następnie tworzenie odwołań w strukturze źródłowych plików XML. Dla *Neolatiny* bazą do zarządzania komentarzem jest neoBytownia¹⁴ – rozwiązanie pierwotnie stworzone pod nazwą Bytownia na potrzeby cyfrowej edycji naukowej dzieł ks. Piotra Skargi¹⁵. Można jednak zakładać, że wkrótce anotorator TEI Publishera okaże się wystarczający dla większości cyfrowych edycji naukowych.

Lekcja płynąca z tego projektu jest jasna: chociaż wiele działań „wskrzeszających” stare witryny da się zautomatyzować za pomocą odpowiednich skryptów, to jednak dane – jeżeli nie miały wcześniej przyporządkowanej struktury – muszą być opracowane ręcznie. Pamiętanie o strukturze i przestrzeganiu standardów jest ważne, jeżeli efekty pracy badaczy mają być zachowane na przyszłość. Gdy projekt cyfrowy się zestarzeje – a za każdym razem jest to nieuniknione, gdyż technologie internetowe stale się rozwijają – o wiele łatwiej będzie go znów ożywić, jeśli będzie uporządkowany i właściwie zakodowany.

Od siedemnastowiecznego rękopisu do modeli AI: *Moralia* Wacława Potockiego

Inne wyzwania stawia przed zespołem edycja pierwszej centurii *Moralii* Wacława Potockiego¹⁶. To obszerne dzieło zachowane w rękopisie Biblioteki Narodowej nie doczekało się pełnego krytycznego wydania. Prolegomena filologiczne do twórczości Potockiego opublikował Leszek Kukulski¹⁷, natomiast jedyna edycja *Moralii* – przygotowana przez

Tadeusza Grabowskiego i Jana Łosia na początku XX wieku¹⁸ – nie zawierała komentarza; jej dodatkowym mankamentem były liczne błędy transkrypcji, drobniawo wskazane przez Kukulskiego. Jak pisali Stanisław Grzeszczuk, a w ślad za nim Janusz S. Gruchała, Potockiego może „pokonać” tylko zespół naukowców¹⁹. Wypada chyba dodać: zespół naukowców wspieranych nowoczesną infrastrukturą obliczeniową.

Praca nad tak obszernym dziełem stała się doskonałą okazją do przetestowania użyteczności programów HTR (*Handwritten Text Recognition*) w przygotowywaniu naukowych edycji utworów zachowanych w rękopisach. Transkribus – niegdyś program, obecnie aplikacja przeglądarkowa – służy do automatycznego generowania transkrypcji dyplomatycznych (a także do pewnego stopnia modernizujących) dokumentów powstałych w dowolnym języku, zarówno manuskryptów, jak i druków. Jego działanie opiera się na wykorzystaniu uczenia maszynowego do trenowania modeli rozpoznających każdy charakter pisma. Aby uzyskać model czytający rękę Potockiego, należało najpierw opracować dość dużą i bardzo szczegółową, niemalże bezbłędną próbkę transliteracji, a następnie przyporządkować ją do skanów rękopisu tak, aby komputer mógł na tej podstawie „nauczyć się” obrazów poszczególnych liter oraz znaków interpunkcyjnych. Pierwsza próbka obejmowała pięćdziesiąt stron rękopisu, jednak dopiero przy stu stronach udało się wyszkolić efektywny model mylący się w około trzech znakach na sto²⁰. Korekta wygenerowanej przez Transkribusa transliteracji całego zbioru²¹ już się zakończyła, obecnie trwają prace nad automatycznym przygotowaniem transkrypcji modernizującej pisownię.

Jak się okazało, dodatkową zaletą Transkribusa jest możliwość wstępnego otagowania tekstu metaznacznikami za pomocą przejrzystego w obsłudze interfejsu graficznego użytkownika. Wydawca podobnie jak w anotoratorze TEI Publishera nie musi się posługiwać nawiasami ostrymi, ponieważ cała „magia” kodowania dzieje się w tle i jest widoczna dopiero po wyeksportowaniu transliteracji do pliku XML. W ten prosty sposób można oznaczyć nie tylko postaci czy miejsca, lecz także cechy fizyczne rękopisu, miejsca nieczytelne, skreślone, zamazane, fragmenty niepewne oraz np. elementy struktury tekstu, takie jak tytuł, wiersz, strofa czy marginalia. Wprowadzenie tych znaczników przez badacza już na etapie pracy w Transkribusie sprawia, że późniejsze dostosowanie wyjściowych plików do potrzeb TEI Publishera (i w domyśle: całej edycji) jest łatwiejsze – to samo można powiedzieć o dziejącym się niejako przy okazji przyporządkowywaniu tekstu do miejsc na skanie.

Z Transkribusa oprócz dokumentów XML można pobrać niesformatowane pliki tekstowe (TXT) zawierające surowe dane. Aktualnie służą one do wytrenowania modelu AI, który niczym tłumacz z języka wyjściowego na język docelowy będzie generował transkrypcję modernizującą pisownię siedemnastowieczną według narzuconych mu reguł²². Takie narzędzie, o ile powstanie i będzie efektywne, mogłoby wspierać edytorów przy innych edycjach. Bez wątpienia zaletą modelu wykorzystywanego wielokrotnie, pracującego na jeszcze większych zbiorach tekstów, byłoby osiągnięcie w przyszłości dużego korpusu z ustandaryzowaną pisownią. Dlatego też LabEdyt bada możliwość zastosowania danych z elektronicznego korpusu tekstów barokowych KorBa²³ jako materiału treningowego. Procedura postępowania jak dotąd doprowadziła do stworzenia listy frekwencyjnej wyrazów i ich form w obrębie *Moralionów*, a także do zestawienia jej z formami transliterowanymi oraz modernizowanymi obecnymi w korpusie. Tak powstałe pary treningowe zostały uszeregowane według odległości Damerau-Levenshteina (od form najbardziej różniących się po zmodernizowaniu do par, w których różnice są małe lub żadne). Zadaniem edytorów jest obecnie określenie, czy sposób transkrybowania mniej lub bardziej konsekwentnie stosowany w KorBie spełnia wymogi przyjęte w edycji, a jeśli nie – sformułowanie odmiennych zasad modernizacji. Przed laty w Katedrze Edytorstwa i Nauk Pomocniczych UJ powstały szczegółowe zasady transkrypcji do planowanej edycji krytycznej *Moralionów* zredagowane przez Janusza S. Gruchałę. Podjęte w nich ustalenia są punktem wyjścia do budowy modelu, jednak najprawdopodobniej modernizacja wykonywana maszynowo będzie dalej posunięta, niż to zakładały pierwotne plany. Intuicja, aby pójść dalej w uwspółcześnianiu, wynika z tego, że w edycji cyfrowej obok faksymile pojawi się także transliteracja, dzięki czemu tekst „do czytania” mógłby zachowywać wyłącznie te cechy dawnej polszczyzny, które na poziomie zapisu prowadzą do odmiennej realizacji fonetycznej.

Osobnym zagadnieniem jest kwestia komentarza do *Moralionów*, który – opracowywany na razie w formie tradycyjnych przypisów – będzie wymagał adaptacji do wymogów środowiska cyfrowego. Jak wiadomo, Potocki swoje dzieło stworzył na podstawie lektury *Adagionów* Erazma z Rotterdamu. Znany jest egzemplarz²⁴, z którego korzystał poeta z Łużnej, o czym świadczą jego odręczne notatki – odnalazł go Leszek Kukulski. Interesujące byłoby pokazanie podobizny tego dzieła w planowanej edycji. Są to rzeczy z perspektywy technicznej łatwe do uzyskania, na dodatek w środowisku cyfrowym właściwie bezkosztowe, jednak trudność polega na umiejętnej

adaptacji komentarza filologicznego, który w edycji przeznaczony do wyświetlania w internecie powinien korzystać z dobrodziejstw hipertekstu.

Projekt opublikowania *Moralionów* w Jagiellońskiej Platformie Cyfrowej z początku zakładał pełne krytyczne wydanie pierwszych stu utworów w obrębie zbioru, jednak odkryte możliwości wdrożenia automatyzacji skłoniły wykonawców do opracowania całego tekstu – przynajmniej do poziomu transkrypcji modernizującej, a być może także tagowania semantycznego, które również, jak się wydaje, może być w dziele Potockiego zautomatyzowane za pomocą NER (*Named Entity Recognition*), czyli metody przetwarzania języka naturalnego stosowanej w celu oznaczenia specyficznych danych, takich jak imiona czy nazwy geograficzne²⁵.

Od pliku produkcyjnego książki do TEI: korespondencja Szymona Laksa z Krystyną i Czesławem Bednarczykami

Trzecia edycja pilotażowa bada możliwości przetworzenia już wydrukowanej edycji naukowej do wersji cyfrowej. Testowany jest tu schemat postępowania w przypadku, gdy edytor opracował tekst do druku i dysponuje plikami gotowej książki, a w szczególności – dokumentem programu InDesign. Można zakładać, że od wersji tekstu w Wordzie do wersji przed eksportem PDF-a drukarskiego zachodzą jeszcze korekty i uzupełnienia, dlatego najbardziej dopracowana, ostateczna postać edycji, która jednak w dalszym ciągu daje się w kontrolowany sposób przekonwertować do formatu XML, znajduje się w najpopularniejszym programie do składu z pakietu Adobe²⁶. Jako materiał do eksperymentu posłużyła obszerna edycja korespondencji Szymona Laksa z właścicielami londyńskiej Oficyny Poetów i Malarzy, przygotowana do druku przez absolwentkę edytorstwa Jadwigę Malik. Książka o objętości około pięciuset stron ukazała się w 2018 roku jako drugi tom serii „Materiały do Dziejów Oficyny Poetów i Malarzy” pod redakcją Janusza S. Gruchałę²⁷.

Listy bardzo często stają się przedmiotem cyfrowych edycji – dzieje się tak ze względu na ich konwencjonalną, wyrazistą strukturę, łatwą do zakodowania w TEI. Repertuar znaczników wykorzystywanych do kodowania korespondencji jest ograniczony, a przykłady dostępne w oficjalnej dokumentacji TEI wydają się nietrudne do zaaplikowania. Plików InDesigna jednak nie da się w prosty sposób przekonwertować

do jakiegokolwiek formy XML-a, gdyż język ten nie jest podstawowym formatem dokumentów w tym programie²⁸. Aby zapisywać pliki z rozszerzeniem „INDD” jako XML, za każdym razem najpierw należy je odpowiednio otagować, co oznacza, że trzeba przygotować dla nich zestaw reguł tagowania, które następnie posłużą jako wzór do eksportu plików XML. Zestaw ten definiuje się w plikach DTD (*Document Type Definition*), które określają, jakie elementy mogą występować w danym miejscu struktury dokumentu – czyli np., że formuła pożegnania wyrażona znacznikiem <salute> nie może się pojawić na początku listu. Mając tak opracowaną deklarację, można przystąpić do tagowania tekstu złamanego w InDesignie. Aby usprawnić ten proces, warto dodatkowo posłużyć się stylami akapitowymi oraz stylami obiektów, mapując je do uprzednio stworzonej siatki tagów.

Dotychczasowe próby przeniesienia edycji prymarnie drukowanych do formatów odpowiednich dla wydań cyfrowych nastroją optymistycznie, choć nie można jeszcze mówić o pełnym sukcesie. Występują nieliczne problemy, jak brak automatycznego indeksu osobowego w pliku korespondencji przekazanym zespołowi przez studio graficzne, w związku z czym nie sposób uzyskać automatycznego tagowania znacznikiem <persName> wszystkich nazwisk osób wymienionych w listach. Pracownia, która przygotowywała książkę do druku, z nieznanых przyczyn usunęła z tekstu znaczniki indeksu. Zadanie, które mogłoby zostać wykonane za pomocą prostego skryptu Petera Kahrela²⁹, być może będzie realizowane – tak jak w *Moraliach* – z wykorzystaniem metod przetwarzania języka naturalnego lub – co bardziej prawdopodobne – przez utworzenie pliku konkordancji z już istniejącego indeksu i próbę ponownego oznaczenia wystąpień w tekście za pomocą anotatora dostępnego w TEI Publisherze. Rzecz jasna taka automatyzacja nie poradzi sobie z metonimiami.

Projekty studenckie

Na koniec warto wspomnieć, że członkowie zespołu Laboratorium Edytorstwa Cyfrowego zajmują się również dydaktyką tworzenia cyfrowych edycji naukowych. W ramach kursów dla studentów drugiego stopnia edytorstwa udało się jak dotąd opracować niedużą edycję polskiego tłumaczenia pracy Scipione Amatiego o wyprawie posłów japońskich do króla hiszpańskiego w XVII wieku oraz przygotować model automatycznego rozpoznawania polskojęzycznych tekstów drukowanych szwabachą, trwają prace nad edycją dydaktyczną szesnasto- i siedemnastowiecznych tłumaczeń Księgi

Psalmów na język polski. Mikroedycja *Posłów japońskich* była pierwotnie hostowana na serwerze wirtualnym Mitr.us stworzonym przez Jakuba Mrugalskiego z myślą o pasjonatach IT, obecnie projekt znajduje się na serwerach uniwersyteckich i jest dostępny w Jagiellońskiej Platformie Cyfrowej. Wspomniany model rozpoznawania szwabachy – generujący transliterację – jest z kolei wkładem w rozwój Transkribusa. Został on upubliczniony, zatem może z niego korzystać każdy użytkownik aplikacji.

Key Words: Digital Scholarly Editions (DSEs), Jagiellonian Digital Platform, Jan Kochanowski (1530–1584), Wacław Potocki (1621–1696), the application of AI in the Humanities

Abstract: The Digital Scholarly Editing Laboratory (LabEdyt) is an interdisciplinary research team operating within the Jagiellonian Centre for Digital Humanities as part of the Digital Humanities Lab flagship project. Its primary goal is to develop workflow frameworks and best practices for DSEs.

LabEdyt pursues this objective through three pilot projects: *Neolatina Sarmatica* (a repository of critical editions of Latin texts authored by writers associated with the former Polish-Lithuanian Commonwealth), a digital edition of Wacław Potocki's *Moralia*, and a digital edition of correspondence. These efforts are complemented by the Jagiellonian Digital Platform, a service aggregating digital humanities projects at the Jagiellonian University.

This article presents the team's achievements in the ongoing projects and discusses the challenges and issues faced by editors seeking to publish the results of their textual studies in a digital environment.

¹ Projekt Flagowy Digital Humanities Lab jest realizowany w Uniwersytecie Jagiellońskim ze środków Inicjatywy Doskonałości – Uczelnia Badawcza za lata 2020–2026.

² E-editions, <https://www.e-editions.org>; TEI Publisher, <https://www.teipublisher.com>.

³ Od co najmniej kilku lat nie można już twierdzić, że nie istnieją żadne polskie cyfrowe edycje naukowe. Problem, który coraz wyraźniej się zarysowuje, polega na braku jednolitych standardów ich tworzenia, dokumentowania i recenzowania, a co za tym idzie – na zmniejszonej refleksji nad tym, jak w sposób zrównoważony realizować projekty odporne na starzenie się. Zob. np. przedsięwzięcia realizowane w Centrum Humanistyki Cyfrowej PAN, edycje tworzone przy wsparciu konsorcjum CLARIN-PL czy wyróżniony ostatnio projekt APOCRYPHA (*Staropolskie apokryfy Nowego Testamentu*), zrealizowany pod kierunkiem Doroty Rojszczak-Robińskiej, który jednak przez twórców jest określany mianem narzędzia, nie zaś edycji.

⁴ Takie wnioski można było wysnuć w trakcie dyskusji panelowej zorganizowanej z okazji zainaugurowania Jagiellońskiej Platformy Cyfrowej (6 grudnia 2024 roku). Na spotkaniu swoim doświadczeniem pracy w środowisku cyfrowym podzielił się Bogdan Hojdis (UAM), Agnieszka Kramkowska-Dąbrowska (IBL PAN), Wojciech Kruszewski (KUL) oraz Bartłomiej Szleszyński (IBL PAN). Por. także rozmowę przeprowadzoną przez Wojciecha Kruszewskiego i Dariusza Pachockiego z Januszem S. Gruchalą: „Pracy edytorskiej nie da się zrobić, siedząc w kawiarni”, „Sztuka Edycji. Studia Tekstologiczne i Edytorskie” 2022, nr 2 (22): *Książka bez granic*, red. M. Bizior-Dombrowska i P. Chłopek, s. 195–207.

⁵ W sierpniu 2024 roku przedstawiciele LabEdytu uczestniczyli w dorocznej konferencji Alliance of Digital Humanities Organizations w Waszyngtonie (zob. *DH2024 Book of Abstracts*, ed. by J. Karajikar, A. Janco, J. Otis, <https://zenodo.org/records/13834936>, dostęp: 28.02.2025); w październiku – w konferencji Texts, Languages and Communities. TEI 2024 w Buenos Aires (zob. *Texts, Languages and Communities: TEI 2024. Book of Abstracts*,

ed. by G. Del Rio Riande, <https://tei2024.tei-c.org/pdf/tei2024boa.pdf>, dostęp: 28.02.2025). We wrześniu 2025 roku Projekt Flagowy Digital Humanities Lab wspólnie z TEI Consortium przygotował na Uniwersytecie Jagiellońskim międzynarodową konferencję TEI Annual Conference and Members' Meeting 2025 (<https://tei2025.confer.uj.edu.pl>; materiały konferencyjne: <https://zenodo.org/communities/tei2025>, dostęp: 20.09.2025), z kolei nieco wcześniej – w lipcu – Kraków gościł prelegentów na konferencji organizowanej przez PF DHLab wspólnie z redakcją „Journal of Computational Literary Studies” (<https://jcls.io/site/ccsls2025/>, dostęp: 20.09.2025), poświęconej zastosowaniu metod obliczeniowych w literaturoznawstwie. Na krakowską humanistykę cyfrową czekają następne projekty – m.in. konferencja European Association for Digital Humanities 2026.

⁶Warto być może wymienić dostępne alternatywy, które wszakże nie rozwiązują problemu w pełni: Visual Studio Code z rozszerzeniem Scholarly XML, LEAF-Writer oraz JinTap. Zwłaszcza dwa ostatnie narzędzia zmierzają w kierunku kodowania WYSIWYM (*What You See Is What You Mean*) – ten rodzaj interfejsu jest szczególnie pożądany przez edytorów naukowych.

⁷Jagiellońska Platforma Cyfrowa, <https://labeledy.dhlab.uj.edu.pl> (dostęp: 28.02.2025).

⁸*Neolatina Sarmatica*, red. nauk. G. Urban-Godziek, red. tech. M. Turska, I. Grab-ska-Gradzińska, H. Bermúdez Sabel, A. Krzysik i J. Hałaczkiwicz, oprac. danych A. Rykowska i M. Wiater, koordynacja M. Komorowska, <https://neolatina.dhlab.uj.edu.pl> (dostęp: 28.02.2025).

⁹Jest to pewne uproszczenie. Teksty wyświetlane w oknie przeglądarki internetowej przeważnie są zakodowane w HTML-u, który również – jako język znaczników (ang. *Hyper-Text Markup Language*) – pozwala wielowymiarowo strukturyzować dane. Pierwotna wersja *Neolatiny* nie korzystała jednak z tej możliwości, co można dostrzec, analizując źródło strony. W klasach oraz stylach nadanych lokalnie znacznikom widać wyraźnie, że formatowanie zostało dość przypadkowo zaimportowane z plików programu Word; D. Schmidt, *True Interoperability for Digital Scholarly Editions*, „Umanistica Digitale” 2021, No. 10.

¹⁰J. S. Gruchała, *Wirtualny wydawca i użytkownik edycji elektronicznej*, w: *Europejski kanon literacki*, red. E. Wichrowska, Warszawa 2011. Z tą koncepcją w zasadzie należało by polemizować. Pliki XML, stanowiące podstawę większości tworzonych obecnie cyfrowych edycji naukowych, nie są wprawdzie tożsame z edycjami, ale posługując się terminologią nauk przyrodniczych, można by je porównać do macierzystych komórek totipotencjalnych: dzięki transformacjom są zdolne przekształcić się zarówno w edycję cyfrową, jak i w tradycyjne, drukowane wydanie. Istotą cyfrowych edycji naukowych byłoby raczej otwarcie się na inne teksty za pomocą API, NLP oraz różnego rodzaju ontologii. Taka płynna wymiana danych tekstowych jest niemożliwa do osiągnięcia w monadycznym świecie druku.

¹¹Znane ze specyfikacji TEI atrybuty @rend, @rendition oraz @style, którym można przypisać wartości określające chociażby wyśrodkowanie tekstu czy jego kolor, tylko pozornie zdają się odpowiadać za wygląd edycji cyfrowej – w rzeczywistości służą one do opisu cech wizualnych podstawy wydania. Mylne wrażenie wynika z widocznej w edycjach cyfrowych dokumentacyjnej skłonności do odwzorowywania rozwiązań typograficznych podstawy źródłowej; TEI: *Guidelines for Electronic Text Encoding and Interchange*, P5 Version 4.9.0, <https://tei-c.org/release/doc/tei-p5-doc/en/html/index.html> (dostęp: 28.02.2025).

¹²O ile zbudowanie edycji działającej lokalnie (na komputerze użytkownika) już po utworzeniu bazy eXist-db nie sprawia większych trudności osobie znającej podstawy języków HTML i CSS, o tyle prawdziwym wyzwaniem okazuje się udostępnienie efektów tych prac w internecie. Tu już niezbędna staje się chociażby minimalna umiejętność obsługi serwerów typu VPS (*Virtual Private Server*) za pomocą wiersza poleceń oraz zrozumienia architektury kontenerów Docker; wiele ułatwia także znajomość XQuery, języka służącego w TEI Publisherze do konfigurowania plików ODD (*One Document Does it all*) odpowiadających za „tłumaczenie” XML-a między innymi na HTML.

¹³Edytor anotacji z graficznym interfejsem użytkownika istnieje w TEI Publisherze od wersji 7.1.0 (2021), jednak dopiero jego najnowsza wersja spełnia potrzeby bardziej skomplikowanego komentarza. Oprócz dodawania odwołań do zewnętrznych baz kontroli autorytatywnej, takich jak GND, GeoNames czy Wikidata, wspierających budowę sieci semantycznej (*Semantic Web*), wersja 9.0 umożliwia np.łączanie edytorskich objaśnień.

¹⁴Opracowane przez Iwonę Grabską-Gradzińską oraz Piotra Grabskiego-Gradzińskiego.

¹⁵Edycja powstaje w ramach grantu NPRH kierowanego przez Magdalenę Komorowską: „Naukowa edycja pism Piotra Skargi. Część 2: Żywoty świętych, pisma polemiczne i katechetyczne”.

¹⁶W. Potocki, *Moralia*, red. nauk. J. S. Gruchała i A. Gorzkowski, red. tech. M. Turska, I. Grabska-Gradzińska, H. Bermúdez Sabel, A. Krzysik i J. Hałaczkiwicz, koordynacja M. Komorowska, opracowanie danych L. Nowak, Z. Ponulak i M. Warzocha.

¹⁷L. Kukulski, *Prolegomena filologiczne do twórczości Wacława Potockiego*, Wrocław 1962.

¹⁸W. Potocki, *Wacław Potockiego „Moralia” 1688*, t. 1–3, wyd. T. Grabowski i J. Łoś, Kraków 1915–1918.

¹⁹J. S. Gruchała, *Wacław Potocki – problem edytorski*, w: *Wacław Potocki. W 400-lecie urodzin poety*, red. K. Koehler i D. Chemperek, Warszawa 2023, s. 17; S. Grzeszczuk, *O potrzebie i programie badań nad twórczością Wacława Potockiego*, w: *Wśród zagadnień*

polskiej literatury barokowej, cz. 2: *Motywy – inspiracje – recepcja*, red. Z. J. Nowak, Katowice 1980, s. 419.

²⁰Szczegóły trenowania modelu opisałam w artykule pt. *Zastosowanie sztucznej inteligencji w edytorstwie naukowym – przykład „Moralii” Wacława Potockiego*, „Napis. Pismo poświęcone literaturze okolicznościowej i użytkowej” 2024, t. 30, s. 215–232.

²¹Nie generowano transliteracji dla końcowych stron kodeksu przepisanych obcą ręką; nie zawierają one zresztą tekstu *Moralii*, lecz fragment innego dzieła.

²²Opracowaniem modelu od strony koncepcyjnej oraz technicznej zajmuje się Michał Warzocha.

²³Instytut Języka Polskiego PAN, *Elektroniczny korpus tekstów polskich z XVII i XVIII w.*, www.korba.edu.pl (dostęp: 24.09.2025).

²⁴E. Roterodamus, *Adagiorvm Chiliades Des. Erasmi Roterodami Qvatvor Cvm Dimidia Ex Postrema Avtoris Recognitione: In hac aeditione, prioribus tribus Indicibus subiunctus est quartus nouus* [...], <https://rcin.org.pl/dlibra/publication/74578/edition/66983/content> (dostęp: 28.02.2025).

²⁵Funkcja ta jest dostępna w TEI Publisherze od wersji 8.0 i korzysta z biblioteki SpaCy. Chociaż istnieje w niej model NER dla języka polskiego (przygotowany w Instytucie Podstaw Informatyki PAN), zapewne znów należałoby opracować próbkę danych, a następnie wytrenować własny model uwzględniający specyfikę siedemnastowiecznej polszczyzny; W. Meier, *Names Sell: Named Entity Recognition in TEI Publisher*, <https://www.e-editiones.org/posts/names-sell-named-entity-recognition-in-tei-publisher/> (dostęp: 28.02.2025).

²⁶Przyjmuje się, że korekta autorska na składzie powinna ograniczać się do absolutnego technicznego minimum, jednak praktyka pokazuje, że autorzy, mając przed oczami rozkładki książki, odkrywają błędy wcześniej niedostrzeżone; pojawia się ponadto pokusa dodawania drobnych uzupełnień – tego wszystkiego nie znajdziemy w zredagowanych plikach programu Word.

²⁷S. Laks, K. i C. Bednarczykowie, *Korespondencja*, wstęp i oprac. J. Malik, Kraków 2018 („Materiały do Dziejów Oficyny Poetów i Malarzy”, t. 2).

²⁸W przeciwieństwie do plików tekstowych z pakietu Office, zapisywanych od 2007 roku w formacie DOCX (litera „X” oznacza tu XML). Konwersja takich dokumentów, będących w istocie skompresowanymi plikami ZIP, nie sprawia większych trudności, ponieważ może przebiegać z wykorzystaniem schematów transformacji XML do TEI, dostępnych chociażby w edytorze Oxygen.

²⁹P. Kahrel, *Indexes and Concordances*, https://creativepro.com/files/kahrel/indexsign/lists_indexes.html (dostęp: 28.02.2025).