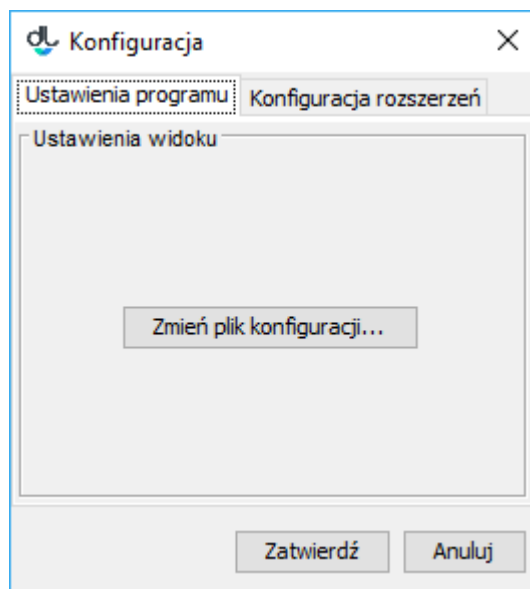


07. Konfiguracja aplikacji redaktora

generalConfig.propertiesWstęp

Po wybraniu opcji Konfiguracja z menu Program pojawi się okno konfiguracji programu ([rysunek poniżej](#)).



Okno konfiguracji programu

W oknie konfiguracji znajdują się zakładki odpowiadające za konfigurację różnych cech programu. W kolejnych podrozdziałach omówione są wszystkie zakładki. Wartości domyślne poszczególnych parametrów znajdują się w załączniku konfiguracja domyślna.

Ustawienia programu

Parametry konfiguracyjne mogą być pobierane z pliku konfiguracyjnego. Aby zmienić plik konfiguracyjny aplikacji wybierz przycisk Plik konfiguracyjny... i wskaż w otwartym oknie katalog lub URL pliku konfiguracyjnego. W przypadku podania katalogu, musi się w nim znaleźć plik o nazwie `generalConfig.properties`. Plik konfiguracyjny aplikacji zawiera zaawansowane parametry konfiguracyjne aplikacji. Jest to plik właściwości, gdzie klucze oddzielone są od wartości znakiem równości. Znajdują się w nim następujące parametry konfiguracyjne:

- `gui.lookandfeel` - wartość to nazwa klasy języka programowania Java, która odpowiada za sposób wyświetlania/zachowania elementów interfejsu graficznego użytkownika. Jeśli brak tej wartości przyjmowany jest domyślny wygląd odpowiedni dla systemu operacyjnego na którym uruchomiono aplikację.
- `messages.refresh.period` - wartość to czas podawany w milisekundach, który mówi o tym co ile milisekund powinna być odświeżana (aktualizowana z serwera) lista wiadomości dla redaktora. Domyślna wartość to 600000 czyli 10 minut.
- `messages.message.period` - wartość to czas podawany w milisekundach, który mówi o tym co ile milisekund powinien być wyświetlany monit użytkownikowi w przypadku, gdy redaktor ma nieodczytane wiadomości. Domyślna wartość to 1800000 czyli 30 minut.
- `newPublication.title.template` - wartość określa szablon tytułu tworzonego elementu (np. publikacji). Szablon ten jest szablonem [velocity](#). W szablonie w miejscu odpowiednio nazwanych zmiennych wprowadzane są wartości atrybutów. Zmienne powinny mieć nazwę odpowiadającą nazwie RDF atrybutu. Domyślna wartość tego parametru to

```
!${Creator}#if ( ($Date || $Title) && $Creator ), #end$!Date#if ( $Title && $Date ), #end$!Title
```

Szablon ten powoduje ustawienie jako nazwy wartości atrybutów o nazwach RDF, w kolejności: `Creator`, `Date` oraz `Title`. Wartości te są oddzielane przecinkiem, a jeśli którejś brakuje, jest pomijana.

- `miniature.maxWidth` - ograniczenie (w pikselach) na szerokość miniatury tworzonej automatycznie z plików publikacji w kreatorze nowej publikacji. Domyślnie szerokość miniatury nie jest ograniczona.
- `miniature.maxHeight` - ograniczenie (w pikselach) na wysokość miniatury tworzonej automatycznie z plików publikacji w kreatorze nowej publikacji. Domyślnie wysokość miniatury nie jest ograniczona. Jeśli ustawiono ograniczenie zarówno na wysokość jak i szerokość miniatury, ostatecznie wykorzystywane jest to ograniczenie, które skutkuje mniejszym rozmiarem miniatury.

- `searchDuplicates.roles.url` - wartość to adres URL pod którym znajduje się lista ról Dublin Core atrybutów, które mają być wykorzystywane w mechanizmie wyszukiwania duplikatów. Domyślna wartość to <http://fbc.pionier.net.pl/owoc/ajax.xml?id=GetDCRolesForDups>.
- `searchDuplicates.results.url` - wartość to początek adresu URL, do którego zostaną dodane parametry wyszukiwawcze w procesie wyszukiwania duplikatów. Domyślna wartość to <http://fbc.pionier.net.pl/owoc/ajax.xml?id=GetDups>.
- `sortTokens` - wartość to lista ciągów znaków oddzielonych symbolem |. Podczas alfabetycznego sortowania elementów w drzewie katalogów i na liście obiektów, podane ciągi znaków są ignorowane, jeśli występują na początku nazwy elementu. Domyślna wartość to "|([[. .]] | the | the".
- `attributesViews.url` - wartość to adres URL pod którym znajduje się opis widoków atrybutów, które mają być dostępne w aplikacji redaktora. W przypadku pliku na dysku lokalnym należy pamiętać, aby poprzedzić ścieżkę frazą `file:///`, a kolejne foldery oddzielać ukośnikiem /. Dodatkowo, każdy dwukropek musi być poprzedzony ukośnikiem {. Na przykład dla pliku `widoki.xml`, położonego w katalogu `C:\dlibra`, para klucz-wartość w pliku konfiguracyjnym przyjmie postać:

```
attributesViews.url=file:///C:/dlibra/widoki.xml
```

Domyślnie wykorzystywany jest zestaw widoków wbudowany bezpośrednio w aplikację redaktora.

Należy pamiętać o zaznaczeniu pola Używaj konfiguracji z podanego źródła. W przeciwnym razie plik konfiguracyjny nie zostanie wczytany.

Przykładowy plik konfiguracji mógłby wyglądać następująco (jeśli w pliku nie ma jakiegoś klucza to przyjmowana jest wartość domyślna):

```
sortTokens=[ ]|/
```

Pobieranie domyślnej konfiguracji z zewnętrznego źródła

W bibliotece cyfrowej może pojawić się potrzeba, by wszyscy redaktorzy pracowali z konkretnym zbiorem ustawień, innym niż ustawienia domyślne. Na przykład, wykorzystanie niestandardowego schematu metadanych może spowodować, że dla wygodnej pracy redaktorów trzeba będzie przygotować specjalne widoki atrybutów, a także zmienić atrybuty wykorzystywane przy tworzeniu domyślnej nazwy publikacji. W tej sytuacji odpowiedni plik konfiguracyjny najlepiej jest umieścić Internecie.

Aby redaktorzy nie musieli samodzielnie wprowadzać adresu takiego pliku, można tak ustawić aplikację czytelnika (www), aby aplikacja redaktora automatycznie otrzymywała odpowiedni adres i wczytywała konfigurację przy każdym uruchomieniu, zastępując domyślne opcje. Funkcja ta jest również wygodna wtedy, gdy w późniejszym czasie zajdzie potrzeba zmiany pewnych ustawień u wszystkich redaktorów. Oczywiście, każdy redaktor może nadpisać pobrane w ten sposób ustawienia na swoim komputerze, stosując metody wymienione w poprzednim podrozdziale.

Adres pliku konfiguracyjnego aplikacji redaktora należy wprowadzić w pliku `components.xml`, znajdującym się w katalogu konfiguracyjnym aplikacji czytelnika (WEB-INF). Wewnątrz znacznika `<component name="pl.psnc.dlibra.web.comp.pages.components.JnlpComponent">` domyślnie znajduje się zakomentowana właściwość w postaci:

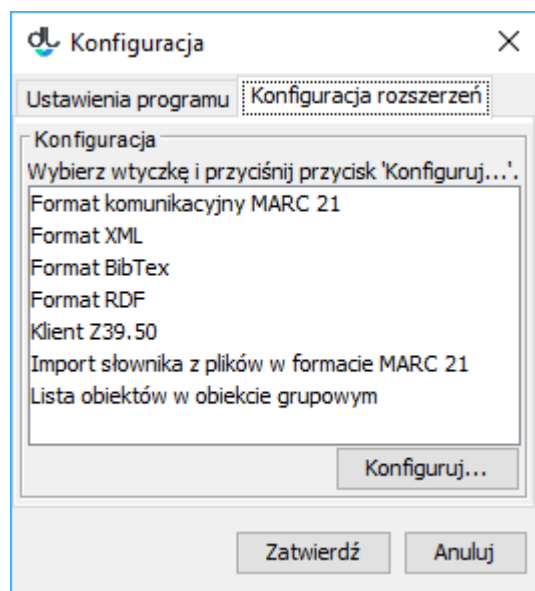
```
<property>
  <name>app.config.default.url</name>
  <value>http://example-library.org/dlibra/editor/generalConfig.properties</value>
</property>
```

Aby uruchomić funkcję automatycznego pobierania dokumentacji, należy odkomentować ten fragment i zmienić przykładowy adres na własny. Zmiana ta nie wymaga restartu serwera www (ale aplikacja redaktora musi być jeszcze raz pobrana ze strony biblioteki cyfrowej).

Konfiguracja rozszerzeń

Zakładka Konfiguracja rozszerzeń umożliwia dostosowanie ustawień poszczególnych rozszerzeń. Na panelu ([rysunek poniżej](#)) znajduje się lista rozszerzeń, które można konfigurować. Aby zmienić ustawienia konkretnego rozszerzenia należy zaznaczyć je na liście i wybrać przycisk Konfiguruj.... Domyślnie w aplikacji znajdują się następujące rozszerzenia:

- format wymienny MARC 21 (szczegóły konfiguracji znajdują się w rozdziale [Import metadanych z formatu MARC](#)).
- format XML (konfiguracja domyślna pozwala na import z formatu RDF oraz MASTER, szczegóły konfiguracji znajdują się w rozdziale [Import metadanych z formatu XML](#)).
- format Bibtex (szczegóły konfiguracji znajdują się w rozdziale [Import metadanych z formatu BibTeX](#)).
- format RDF (szczegóły konfiguracji znajdują się w rozdziale [Eksport w formacie RDF](#)).
- klient Z39.50 (lokalizacja metadanych, szczegóły konfiguracji znajdują się w rozdziale [Pobieranie metadanych poprzez protokół Z39.50](#)).
- import słownika z plików w formacie MARC (szczegóły konfiguracji znajdują się w rozdziale [Importowanie grup wartości z plików MARC](#)).
- lista publikacji w publikacji grupowej (szczegóły konfiguracji znajdują się w rozdziale [Generowanie listy publikacji z publikacji grupowej](#)).



Konfiguracja - zakładka Konfiguracja rozszerzeń