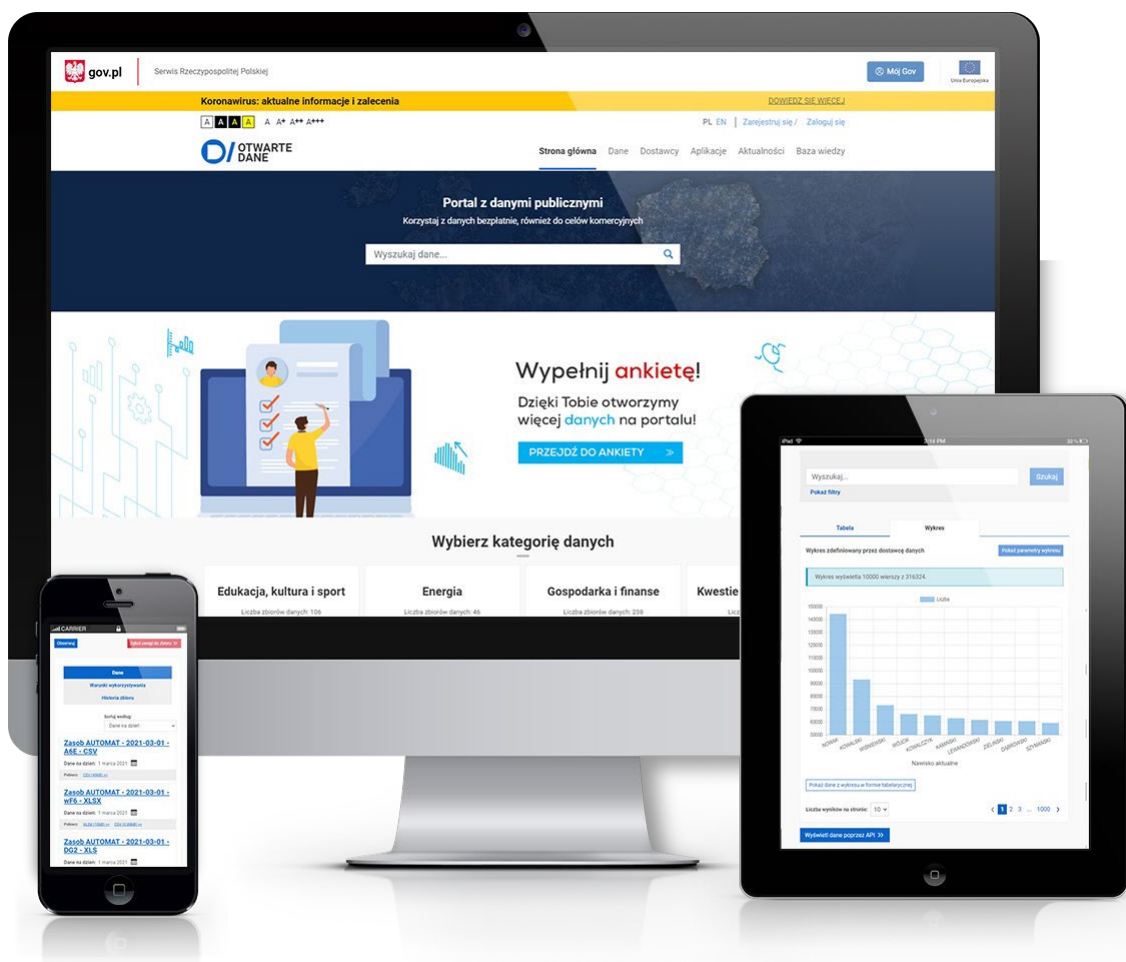


Przewodnik

Automatycznego dodawania danych xml



Wersja 1.02

Spis treści

I. Wstęp.....	4
II. Czynności przygotowawcze - Dostawca	4
Krok 1	4
Krok 2	4
Krok 3	5
Krok 4	5
Krok 5	5
III. Sprawdzanie poprawności pliku XML	6
IV. Przygotowanie pliku xml przez dostawcę.....	7
Krok 1	7
Krok 2	8
Krok 3	8
Krok 4	9
Krok 5	9
V. Aktualizacja danych i zbiorów danych za pomocą pliku .xml – przez dostawcę	10
1. Aktualizacja nowo utworzonych zbiorów i danych za pomocą pliku .xml .	10
Krok 1	10
Krok 3	10
2. Aktualizacja ręcznie dodanych danych	10
Krok 1	10

Krok 2	10
Wskazówka	10
3. Usuwanie zaimportowanych danych – podmiana pliku zasobów (danych)....	11

I. Wstęp.

System Otwarte Dane udostępnia dostawcom funkcjonalność o nazwie **Importer XML** pozwalającą na publikację danych (zbiorów danych, danych) z pliku XML. Przygotowany przez dostawcę plik XML z danymi powinien mieć strukturę zgodną ze zdefiniowanym schematem. Rozwiązanie, które zostało stworzone ułatwia dostawcom aktualizowanie danych na portalu Otwarte Dane.

Zbiory danych oraz dane na portalu mogą być automatycznie zasilane za pomocą informacji zawartych w pliku XML. Plik XML jest przygotowywany przez dostawcę dla pojedynczego źródła danych.

Każdy plik XML zawiera dane o wszystkich zbiorach danych i danych, które dostawca w ramach źródła informacji opublikował lub ma zamiar opublikować na portalu.

II. Czynności przygotowawcze – Dostawca.

Krok 1

Dostawca przygotowuje plik XML, który zawiera informacje o wszystkich zbiorach danych i danych, które dostawca w ramach źródła udostępnił, lub ma zamiar udostępnić na Portalu. Struktura pliku ma być zgodna ze schematem XSD dostępnym pod adresem https://www.dane.gov.pl/static/xml/otwarte_dane_latest.xsd

Krok 2

Dostawca nadaje nazwę plikowi XML zgodnie z przyjętą konwencją.

(nazwa bez spacji zakończona rozszerzeniem .xml).

Plik XML będzie miał ustaloną nazwę podaną jako ciąg znaków bez spacji. Każdy plik XML przygotowywany przez dostawcę w ramach tego samego źródła będzie posiadał tę samą nazwę.

Przykład:

Plik z danymi dla KPRM dla źródła danych CEPiK – Cepik.xml

Krok 3

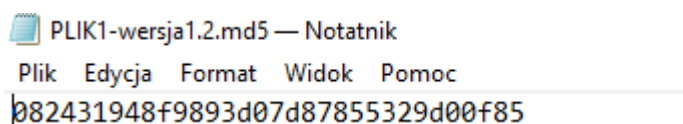
Dostawca przygotowuje plik, zawierający sumę kontrolną pliku XML, wyliczoną algorytmem MD5. Na podstawie pliku .xml generujemy sumę kontrolną (hash) i jej zawartość wklejamy do pliku .md5

Przykład:

Plik z danymi - KPRMCepik.xml

Plik z sumą kontrolną - KPRMCepik.md5

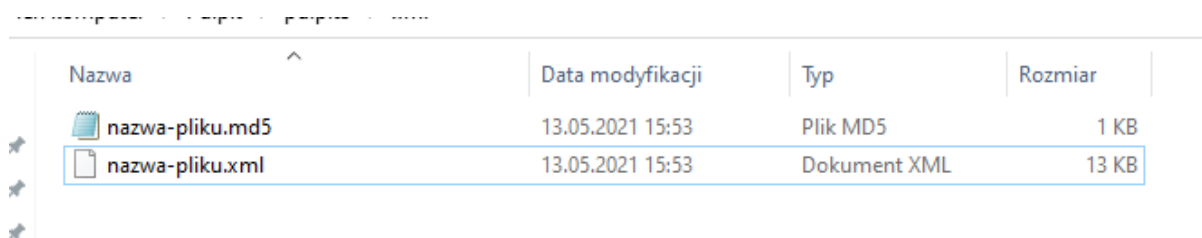
Przykład pliku md5:





Krok 4

Dostawca danych umieszcza oba pliki w ustalonej lokalizacji jest to serwer, na którym przechowuje pliki, w celu późniejszego pobrania pliku przez portal Dane.gov.pl.

Przykład folderu:



Nazwa	Data modyfikacji	Typ	Rozmiar
 nazwa-pliku.md5	13.05.2021 15:53	Plik MD5	1 KB
 nazwa-pliku.xml	13.05.2021 15:53	Dokument XML	13 KB

Krok 5

Dostawca wysyła wniosek o automatyczne zasilanie danych w portalu Dane.gov.pl do administratora portalu na adres kontakt@dane.gov.pl. Wniosek powinien zawierać następujące informacje:

- Imię i nazwisko osoby zgłaszającej wniosek
- Nazwa dostawcy

- c. Adres URL określający lokalizację pliku XML zawierające informacje o wszystkich zbiorach danych i danych dla danego źródła oraz pliku MD5
- d. Dostawca określa częstotliwość z jaką dane mają być aktualizowane:
 - i. Co dzień
 - ii. Co tydzień
 - iii. Co miesiąc
 - iv. Co kwartał

Przykład:

- a. Jan Kowalski
- b. Kancelarii Prezesa Rady Ministrów
- c. <https://mc.gov.pl/plikiXML/Cepik.xml>
- d. Co tydzień

III. Sprawdzanie poprawności pliku XML.

Jest to proces składający się z kilku kroków, który wykonywany jest po naciśnięciu na przycisk **Sprawdź poprawność pliku**. Tą czynność wykonuje administrator podczas dodawania pliku. Jeżeli plik jest źle przygotowany pojawi się komunikat. Wówczas dostawca powinien poprawić błędy pojawiające się w pliku.

Kroki procesu są następujące:

1. Sprawdzanie nazwy pliku:

- Nazwa pliku nie powinna zawierać spacji i powinna być zakończona rozszerzeniem .xml

Przykład poprawnej nazwy: nazwaPliku.xml

- Wszystkie pliki powinny być udostępniane na standardowych portach, czyli **80** w przypadku protokołu http i **443** w przypadku protokołu https.

2. Sprawdzenie nagłówka Content-Type:

- Pod adres url pliku wysyłane jest żądanie **HTTP** typu **HEAD**, a następnie sprawdzany jest nagłówek **Content-Type** odpowiedzi. Musi on zawierać fragment **text/xml** lub **application/xml**.

3. Sprawdzenie hasha **MD5**:

- Najpierw pobierana jest zawartość pliku z hashem przygotowanego przez dostawcę.

Adres URL pliku z hashem MD5 powinien być taki sam jak pliku **XML**, ale z rozszerzeniem **.md5**.

Przykład:

Adres pliku XML: `http://adres.pl/plik.xml`

Adres pliku MD5: `http://adres.pl/plik.md5`

Przykład:

Adres pliku XML: `http://adres.pl/plik`

Adres pliku MD5: `http://adres.pl/plik.md5`

- Następnie system pobiera plik XML, wylicza z niego hash i porównuje z hashem dostarczonym przez dostawcę.
- **UWAGA!** Na obecną chwilę **wielkość liter w hashu** ma znaczenie - powinny być to małe litery.

4. Sprawdzenie zgodności pliku XML z odpowiednią wersją schematu XSD:
Struktura pliku XML ma być zgodna ze schematem XSD dostępnym pod adresem https://www.dane.gov.pl/static/xml/otwarte_dane_latest.xsd

IV. Przygotowanie pliku xml przez dostawcę.

Krok 1

Dostawca danych przygotowuje informacje o swoich zbiorach danych oraz danych w pliku XML zgodnie z najaktualniejszym schematem XSD. Plik XML zawiera informacje o wszystkich zbiorach danych oraz danych, które dostawca w ramach jednego źródła danych udostępnił lub chce udostępnić na portalu. (np. może być to jeden zbiór danych).

Krok 2

Dostawca przygotowuje plik MD5 na podstawie pliku XML (generuje hash).

Przykładowy sposób generowania hasha dla pliku MD5:

- w przeglądarce internetowej wyszukujemy narzędzie do generowania hasha dla pliku MD5 (w Internecie znajduje się wiele darmowych narzędzi)
- wybieramy plik .xml
- pobieramy rezultat (czyli hash)
- hash wklejamy do pliku .md5, który znajduje się w tej samej lokalizacji co plik .xml

Krok 3

Czynność przygotowania całościowego pliku XML oraz pliku MD5 należy powtarzać z podaną w definicji częstotliwością.

Krok 4

Zbiory danych oraz dane na portalu, których identyfikatory dostawcy nie znajdują się w pliku XML oraz posiadają odpowiedni identyfikator definicji zasilania są usuwane i nie będą dostępne na portalu dla użytkowników.

Krok 5

Dla każdego nowoutworzonego danych na portalu następuje pobranie zawartości danych oraz ich standardowa walidacja. Lokalizację danych określa adres URL. Adres URL znajduje się w polu <URL> dla elementu <Resource> w pliku XML. Jeżeli danymi jest plik to w polu <Availability> znajduje się informacja o jego sposobie publikacji.

Przykład xml:

```
<resource status="published">

    <extId>dane_extId_dane_1</extId>

    <url>https://6f71c616ab06.ngrok.io/XLSX.xlsx</url>

    <title>

        <polish>DANE csv REMOTE</polish>
```



```
<english>ENGLISH TITLE - RESOURCE 1</english>

</title>

<description>

    <polish>Opis danych opublikowane z XMLA - aktualizacja</polish>

    <english>English description of first resource</english>

</description>

<availability>remote</availability>

<dataDate>2021-10-10</dataDate>

<lastUpdateDate>2020-12-08T00:00:00.000Z</lastUpdateDate>

</resource>
```

V. Aktualizacja danych i zbiorów danych za pomocą pliku .xml – przez dostawcę.

Po utworzeniu definicji zasilania zbiorów i danych, dostawca ma możliwość dodawania usuwania i modyfikacji obiektów utworzonych za pomocą pliku .xml.

1. Aktualizacja nowo utworzonych zbiorów i danych za pomocą pliku .xml.

Krok 1

Dane jak i zbiory danych są identyfikowane przez wymaganą w pliku **XML** wartość **extIdent** (**external identifier** - identyfikator zewnętrzny) będącą identyfikatorem zbioru danych i danych nadanym przez dostawcę.

Każdy zbiór danych i dane mają przypisany swój unikalny identyfikator **extIdent**. W przypadku zmiany tego identyfikatora system uzna ten obiekt jako nowy – co spowoduje usunięcie danych / zbioru i utworzenie nowego z nowym identyfikatorem.

W przypadku zmiany metadanych zbiorów i danych (typu title, opis, daty), nie jest wymagana zmiana identyfikatora **extIdent**. Wartość **extIdent** dotyczy tylko zbiorów i danych w przypadku ich usuwania i dodawania.

Krok 3

W przypadku zmiany wartości **extIdent** danych, zostanie zasób usunięty (do kosza), a następnie powstaną nowe dane z podanymi w pliku wartościami.

2. Aktualizacja ręcznie dodanych danych.

Krok 1

Dostawca chce w pliku XML zaktualizować zbiory danych i dane które zostały już opublikowane w sposób tradycyjny na portalu (ręcznie).

Krok 2

W takim przypadku dostawca może skorzystać z opcjonalnej wartości **intIdent** (**internal identifier** - identyfikator wewnętrzny) będącej liczbowym id zbioru/danych na portalu, widocznym między innymi w adresie **URL** podczas przeglądania danych.

Aby zaktualizować konkretne dane lub zbiór za pomocą Importera XML, wystarczy w pliku XML odwoływać się do niego przez **intIdent**.

Wskazówka

- **intIdent** ma priorytet w stosunku do **extIdent**, więc w przypadku podania obu tych identyfikatorów, aktualizowany będzie tylko obiekt z odpowiadającą wartością **intIdent**.
- Dobrą praktyką wydaje się użycie **intIdent** tylko podczas pierwszej aktualizacji obiektu, gdyż po niej będzie on już miał nadany identyfikator **extIdent**, po którym będziemy mogli się do niego później odwoływać.

```
<ns2:datasets xmlns:ns2="urn:otwarte-dane:harvester:1.0-rc1">
  <dataset status="published">
    <extIdent>plik2_zbiór_1</extIdent>
    <intIdent>1753</intIdent>
    <title>
      <polish>Zbiór testowy MNISW po aktualizacji int identem</polish>
      <english>ENGLISH DATASET - EN TITLE - po aktualizacji</english>
    </title>
    <description>
      <polish>Opis w wersji PL - opis testowy do testów UAT - po aktualizacji</polish>
      <english>ENGLISH DATASET DESCRIPTION - UAT PHASE TEST - po aktualizacji</english>
    </description>
    <url>https://www.youtube.com</url>
    <updateFrequency>monthly</updateFrequency>
    <categories>1</categories>
    <conditions>
      <source>false</source>
      <modification>false</modification>
      <responsibilities>Warunki dla pola 1 - po aktualizacji</responsibilities>
      <dbOrCopyrighted>Warunki dla pola 2 - po aktualizacji</dbOrCopyrighted>
      <personalData>Warunki dla pola 3 - po aktualizacji</personalData>
    </conditions>
    <resource status="published">
      <extIdent>plik2_zasob1_1</extIdent>
      <url>https://c03c2e84333.agrok.io/plik2.xml</url>
      <title>
        <polish>ZASOB 1 - nowy XML</polish>
        <english>ENGLISH DATASET - EN TITLE - po aktualizacji</english>
      </title>
      <description>
        <polish>Opis zasobu opublikowane z XMLA - aktualizacja</polish>
        <english>CSV file with technical data of vehicles</english>
      </description>
      <availability>local</availability>
      <dataDate>2001-01-01</dataDate>
      <lastUpdateDate>2017-01-01T00:00:00.000Z</lastUpdateDate>
    </resource>
  </dataset>
</tags>
```

3. Usuwanie zaimportowanych danych – podmiana pliku zasobów (danych).

Zbiory danych i dane zaimportowane z pliku XML, przestaną występować na portalu w momencie usunięcia danych z pliku XML, lub po zmianie identyfikatora ExIdent. W przypadku chęci podmiany pliku danych zgodnie z procesem należy usunąć stare dane i utworzyć nowy. W przypadku importera xml aby utworzyć nowy obiekt należy zmienić wartość atrybutu exIdent.

Przykład 1

Użytkownik dodał dane za pomocą importera xml oraz opcji local (plik jest zapisany na naszym folderze lokalnym aplikacji). Użytkownik chce zamienić plik danych na aktualny.

Opis:

W takim przypadku użytkownik powinien zmienić wartość atrybutu exIdent na inny. Dzięki temu importer zidentyfikuje nowe dane o nowym exIdent i usunie dane o starym exIdent.

Przykład 2:

Użytkownik dodał dane za pomocą importera xml za pomocą opcji remove (Plik jest udostępniany zdalnie z serwera dostawcy).

Opis:

W takim przypadku użytkownik nie musi dodawać nowe dane (zmienić exIdent ponieważ plik znajduje się w lokalizacji zdalnej dostawcy). Zawartość pliku jest aktualizowana przez dostawcę.