



Karol Król

Uniwersytet Rolniczy w Krakowie
Wydział Inżynierii Środowiska i Geodezji
Katedra Gospodarki Przestrzennej i Architektury Krajobrazu
k.krol@onet.com.pl

ANALIZA PORÓWNAWCZA WYBRANYCH TECHNIK INTERAKTYWNEJ PREZENTACJI DANYCH LICZBOWYCH

Streszczenie: W obecnych czasach różnorakie podmioty publiczne, w tym rządowe i samorządowe, a także jednostki działające na rynku komercyjnym, gromadzą, przechowują i przetwarzają coraz większe ilości danych, również mikro i makroekonomicznych. Coraz częściej ich analizy upowszechniane są w Internecie w formie różnorodnych wykresów tworzonych za pomocą różnych technik i narzędzi projektowych. W artykule przedstawiono charakterystykę oraz wyniki analizy porównawczej wybranych technik prezentacji danych w formie interaktywnych wykresów generowanych w oknie przeglądarki internetowej. W konkluzji wykazano, że opisywane techniki różnią się m.in. oprawą graficzną, możliwościami projektowymi, technikami programistycznymi, na których bazują, a także licencją oraz dostępem do wsparcia technicznego.

Słowa kluczowe: wizualizacja danych, interaktywne wykresy, testy eksploracyjne.

JEL Classification: O33.

Wprowadzenie

Doznania wzrokowe stają się w natłoku informacji coraz istotniejsze. Odbiorcy poszukują nowych, atrakcyjnych form prezentacji danych, które ułatwią szybkie przyswojenie informacji [Leszkowicz, 2011]. Przekaz oparty na obrazie przemawia do wyobraźni i jest uniwersalny. Obraz staje się wszechobecny, oddziałuje na myślenie odbiorcy, wywołuje emocje i zapada w pamięć [Kampka, 2011, Siennicka, 2015] – „ma większą moc oddziaływania i skuteczniej komunikuje idee niż liczby zapisane za pomocą rzędów cyfr” [Piekarski, 2015, s. 12]. Na znaczeniu zyskują techniki wizualizacji danych, które w efektywny i zarazem efektowny sposób prezentują wyniki przeprowadzonych analiz [McCandless, 2012].

Do niedawna wizualizacja danych była przedmiotem zainteresowania nielicznej grupy analityków, badaczy i projektantów. Dopiero dostęp do Internetu i narzędzi przetwarzania danych umożliwił jej upowszechnienie [Piekarski, 2015]. Wizualizacja to informacja w postaci obrazu, która za pomocą kształtów, linii, hierarchii i układów komunikuje treść przekazu [Paradowski, 2011]. Celem wizualizacji jest przede wszystkim graficzna prezentacja danych [Furmankiewicz, Sołtysik-Piorunkiewicz, Ziuziański, 2014], jednak z powodzeniem może być ona wykorzystana jako narzędzie badawcze pozwalające eksplorować zbiory danych [Piekarski, 2015].

Coraz powszechniejszy dostęp do różnorodnych urządzeń mogących łączyć się z Internetem oraz dynamiczny rozwój technik i narzędzi projektowych sprawiły, że pojawiają się nowe formy i formaty prezentacji danych, w szczególności statystycznych i mających odniesienie przestrzenne. Wynika to m.in. z rosnących oczekiwań odbiorców, którzy poszukują zwięzłego, przejrzystego, atrakcyjnego i jednocześnie interaktywnego przekazu. Wszystkie te kryteria mogą być spełnione przez wizualizacje przyjmujące formę wykresów generowanych dynamicznie w oknie przeglądarki. Dostawcy usług sieciowych oddali do dyspozycji użytkowników liczne narzędzia i techniki programistyczne umożliwiające samodzielne tworzenie oraz publikowanie zaawansowanych i atrakcyjnych wizualnie wykresów, które przyjmować mogą różnorakie formy, począwszy od tradycyjnych wykresów słupkowych przez futurystyczne miraży wzajemnie splecionych linii i obiektów, skończywszy na kokpitach menadżerskich. Techniki i narzędzia te są często udostępniane nieodpłatnie przez pasjonatów i społeczności użytkowników, stanowią formę upowszechniania technologii lub samego dostawcy (promocja marki), lub też wstęp do bardziej zaawansowanych i rozbudowanych usług płatnych. Znajdują również szerokie zastosowanie w wizualizacji danych ekonomicznych [Eurostat, 2016; Google Finance, 2016].

Celem pracy jest analiza porównawcza wybranych technik interaktywnej prezentacji danych w formie wykresów prezentowanych w oknie przeglądarki internetowej, ze szczególnym uwzględnieniem wizualizacji danych liczbowych.

1. Materiały i metody

Analizie porównawczej poddano możliwości projektowe wybranych technik i narzędzi programistycznych oraz funkcjonalność i użyteczność utworzonych za ich pomocą wykresów (tab. 1). Podstawą analizy były testy eksploracyjne typu ad-hoc, które są określane również jako *expert testing* lub *monkey testing* [Agruss, Johnson, 2000; Kölling, Patterson, 2004; Afzal, Torkar, Feldt, 2009; Chhabra, 2012]. Testowane techniki dobrano tak, aby podstawą każdej z nich była inna technologia, co da-

ło pewien przegląd alternatyw projektowych. Ponadto kryteriami wyboru były: dostępność, aktualność oraz możliwość ich nieodpłatnego wykorzystania, ze szczególnym uwzględnieniem projektów komercyjnych.

Tabela 1. Techniki oraz narzędzia projektowe poddane testom ad-hoc

Narzędzie (pisownia oryginalna)	Techniki projektowe	Źródła danych	Licencja
RGraph	JavaScript, HTML5 Canvas, CSS	DataTable (tablica danych zapisana w dokumencie hipertekstowym), CSV or XML files, json	Dual-licensing scheme: Open Source General Public License and Non-GPL licensing
jqPlot	JavaScript, jQuery, HTML, CSS	DataTable (tablica danych zapisana w dokumencie hipertekstowym), AJAX JSON Data Renderer – json data (dane w pliku tekstowym)	MIT License and Open Source General Public License (GPL version 2.0 license)
Google Charts Visualization API: Geomap	Adobe Flash, JavaScript, Google Visualization API, HTML, CSS,	Google Chart libraries (jsapi and loader) DataTable (tablica danych zapisana w dokumencie hipertekstowym)	Google Visualization API Terms of Service
Google Charts Visualization API	SVG, VML, JavaScript, Visualization API, HTML, CSS	Google Chart libraries (loader) DataTable (tablica danych zapisana w dokumencie hipertekstowym), Google Spreadsheets, Google Fusion Tables	
CSS, Cascading Style Sheets	HTML, CSS	Brak źródła danych liczbowych	Ogólnodostępny język programowania

Źródło: Opracowanie własne.

1.1. Testy eksploracyjne ad-hoc

Testy eksploracyjne ad-hoc to jedna z najmniej formalnych metod testowania oprogramowania. Badania eksploracyjne (rozpoznawcze) są wykonywane bez zaplanowanych wcześniej przypadków użycia (test cases), jednak z wyznaczonym celem, mają także charakter wstępny [Kaner, Falk, Nguyen, 1999; Itkonen, Mäntylä, 2014]. Często towarzyszy im improwizacja, przez co mogą być trudne do odtworzenia. Testy ad-hoc poprzedzają zwykle testy zasadnicze, wspomagają ich planowanie lub z uwagi na ograniczenia zasobów lub czasowe, całkowicie je zastępują.

Zaletą testów ad-hoc jest ich funkcja poznawcza. Pozwalają one zapoznać się ze specyfiką testowanej aplikacji, która nie musi być dokładnie opisana w dokumentacji projektowej. Scenariusz tak przeprowadzanych testów przewi-

duże obserwację i odnotowanie zachowań programu w typowych warunkach, w jakich pracuje, tj. podczas realizacji założeń projektowych, bądź też czynności wykonywanych przez użytkowników [Król, 2016].

W opisywanym przypadku testy ad-hoc przeprowadzono na gruncie funkcjonalności i możliwości projektowych, z punktu widzenia przeciętnego użytkownika, posługując się techniką samooceny. Badania polegały zatem na rejestracji doświadczeń i obserwacji poczynionych w trakcie tworzenia i wdrażania aplikacji internetowych utworzonych według przyjętych założeń projektowych.

1.2. Główne założenia projektowe

Głównym założeniem projektowym było utworzenie modelowej wizualizacji danych statystycznych. Do interaktywnej prezentacji wybrano 9 wskaźników oddających poziom realizacji celów założonych w dokumencie Strategia 2020, który opisuje strategię Unii Europejskiej w zakresie wzrostu gospodarczego i zatrudnienia na okres od 2010 do 2020 r. Strategia ta zakłada wzrost wydatków na inwestycje w badania i rozwój oraz zatrudnienia osób w wieku 20-64 lat, osiągnięcie wybranych celów z zakresu klimatu i energii, poprawę wykształcenia, a także ograniczenie liczby osób zagrożonych ubóstwem. Realizacja celów jest monitorowana przez Eurostat – urząd statystyczny Unii Europejskiej [Rozporządzenie Parlamentu Europejskiego i Rady UE, 2015], w oparciu o zestaw wskaźników oddających ogólny postęp w tworzeniu inteligentnej, ekologicznej gospodarki sprzyjającej włączeniu społecznemu, o wysokiej stopie zatrudnienia, wydajności i spójności społecznej [Strategia, 2010]. Na wykresach zaprezentowano dane z lat 2013-2015 oraz jako punkty odniesienia – dane z roku 2008 i założenia przyjęte jako docelowe w roku 2020.

W projekcie przyjęto założenie, że prezentacja ma mieć multimedialny i interaktywny charakter oraz ma stanowić komponent witryny internetowej. Ponadto głównym nośnikiem informacji mają być wykresy liniowe (*line charts*), słupkowe (*bar charts*) lub kołowe (*pie charts*), których forma graficzna oraz zasada działania mogą być różne, w zależności od zastosowanych technik projektowych. Dodatkowo podjęto próbę prezentacji wykresów w postaci kokpitu menadżerskiego (*management dashboard*), tj. interaktywnej prezentacji różnych wykresów jednocześnie, z możliwością sortowania danych [Ziuziański, 2014] oraz zobrazowania danych na mapach cyfrowych, tj. w odniesieniu przestrzennym [Król, Prus, 2016].

2. Charakterystyka analizowanych technik projektowych

RGraph to biblioteka JavaScript udostępniana nieodpłatnie (również do użytku komercyjnego), która w sposób dynamiczny generuje w oknie przeglądarki różnorakie, interaktywne wykresy, także animowane. Wykresy są tworzone w ramach obiektu „canvas” (HTML5 <canvas> tag), który stanowi przestrzeń dla JavaScript. W obiekcie „canvas” możliwe jest generowanie grafik rastrowych, od prostych figur geometrycznych do różnego rodzaju krzywych, gradientów, wykresów i transformacji, a także wyświetlanie animacji.

Bibliotekę RGraph można wykorzystywać zgodnie z warunkami licencji GPL (General Public License). Kod źródłowy biblioteki jest ogólnodostępny i można go modyfikować wedle uznania (zgodnie z wytycznymi licencji GPLv2.0).

Biblioteka jqPlot umożliwia generowanie w oknie przeglądarki różnorakich wykresów, w tym liniowych, słupkowych i kołowych. Pozwala zaprogramować animacje oraz interakcyjność, np. przybliżanie widoku danych, czy też prezentację wartości liczbowych na życzenie użytkownika. Narzędzie bazuje na jQuery JavaScript i jest udostępniane nieodpłatnie w ramach otwartych licencji Open Source – MIT lub GPL (v2.0). Licencja MIT umożliwia używanie, kopiowanie, modyfikowanie i rozpowszechnianie (w tym sprzedaż) oryginalnego lub zmodyfikowanego programu, z zachowaniem warunków licencyjnych i informacji o autorze wersji podstawowej [Rosen, 2005].

Pierwsze wydanie jQuery zostało udostępnione w 2006 r. Biblioteka szybko zyskała uznanie i stała się jedną z najpopularniejszych technologii wykorzystywanych do tworzenia aplikacji internetowych. Do zalet jQuery zaliczyć można relatywnie prostą implementację w strukturze dokumentu hipertekstowego, łatwość przekształcania i modyfikacji oraz funkcjonalność [Bennett, O’Neill, Kammerer, 2014; Król, Salata, 2013; Król, Szomorova, 2015; Verens, 2012].

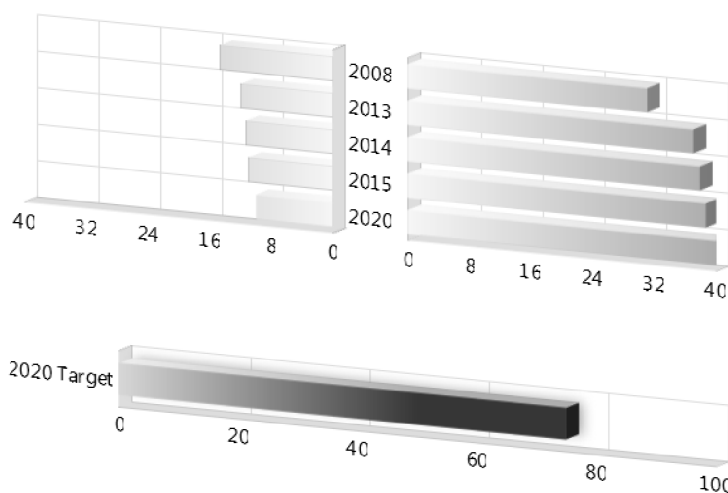
Google Visualization API to jedno z najbardziej popularnych w świecie i jednocześnie ogólnodostępnych narzędzi, które pozwala prezentować dane w formie interaktywnych map i wykresów [Zhu, 2012]. Visualization API (Visualization Application Programming Interface) stanowi zbiór klas JavaScript, wywoływanych i prezentowanych w strukturze dokumentu hipertekstowego (API to zestaw procedur, protokołów i narzędzi umożliwiających wymianę informacji pomiędzy oprogramowaniem komputerowym; Król, 2015). Wykresy Google Charts są narzędziem prezentacji danych w oknie przeglądarki, podczas gdy Visualization API jest techniką ich generowania [Król, 2016]. Warunki korzystania z Google API są szczegółowo opisane w postanowieniach licencyjnych (Google APIs Terms of Service).

Kaskadowe arkusze stylów (Cascading Style Sheets, CSS) to język opisujący formę prezentacji różnorodnych obiektów wyświetlanych w oknie przeglądarki, związany przede wszystkim z definiowaniem „wyglądu” stron internetowych. Arkusz stylów CSS jest listą atrybutów i ich wartości (tzw. reguł) definiujących sposób wyświetlania przez przeglądarkę konkretnych obiektów. Style CSS, użyte jako podstawowa technika projektowa, są wystarczające do utworzenia prezentacji danych w formie wykresów.

3. Interaktywna prezentacja danych

Implementacja biblioteki RGraph w strukturze dokumentu hipertekstowego jest relatywnie prosta. Zróżnicowane graficznie wykresy można tworzyć, wzorując się na przykładach udostępnionych na stronie internetowej projektu. Dane prezentowane w formie wykresu (rys. 1) domyślnie zapisywane są w macierzy, która stanowi integralny fragment kodu JavaScript. Technika ta dopuszcza również stosowanie zewnętrznych źródeł danych.

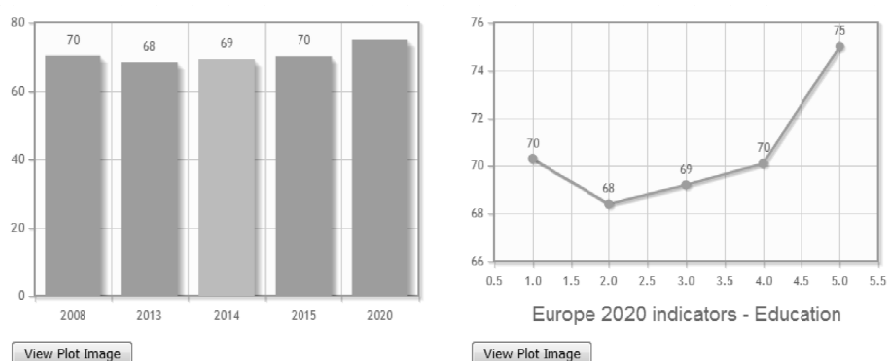
RGraph nie umożliwia prezentacji danych w odniesieniu przestrzennym, dodawania lub usuwania prezentowanych zmiennych (modyfikacji zakresu prezentacji danych) oraz nie udostępnia zaawansowanych funkcji nawigacji i ustawień wykresu na życzenie użytkownika w czasie rzeczywistym (zmiana np. kolorów, linii siatki, legendy; tab. 3).



Rys. 1. Prezentacja w oknie przeglądarki internetowej wartości wskaźnika zatrudnienia w latach 2008, 2013-2015 oraz jego wartości zamierzonej w roku 2020 (tzw. zrzut ekranu)

Źródło: Opracowanie własne za pomocą RGraph na podstawie danych Eurostat [2016].

Podobnie jak RGraph, biblioteka jqPlot nie umożliwia prezentacji danych w odniesieniu przestrzennym, modyfikacji zakresu prezentowanych zmiennych oraz tworzenia wykresów trójwymiarowych (3D). Umożliwia jednak zaprogramowanie zaawansowanych funkcjonalności nawigacji i ustawień wykresu, dostępnych dla użytkownika (tab. 3). Pomijając te ograniczenia, biblioteka jest rozbudowana i daje wiele możliwości projektowych (rys. 2). Pewnych trudności może przysporzyć implementacja skryptów odpowiedzialnych za generowanie wykresów, ponieważ są one powiązane z typem wykresu.



Rys. 2. Wartości wskaźnika zatrudnienia w latach 2008, 2013-2015 oraz w roku 2020, prezentacja w oknie przeglądarki internetowej (tzw. zrzut ekranu)

Źródło: Opracowanie własne za pomocą jqPlot na podstawie danych Eurostat [2016].

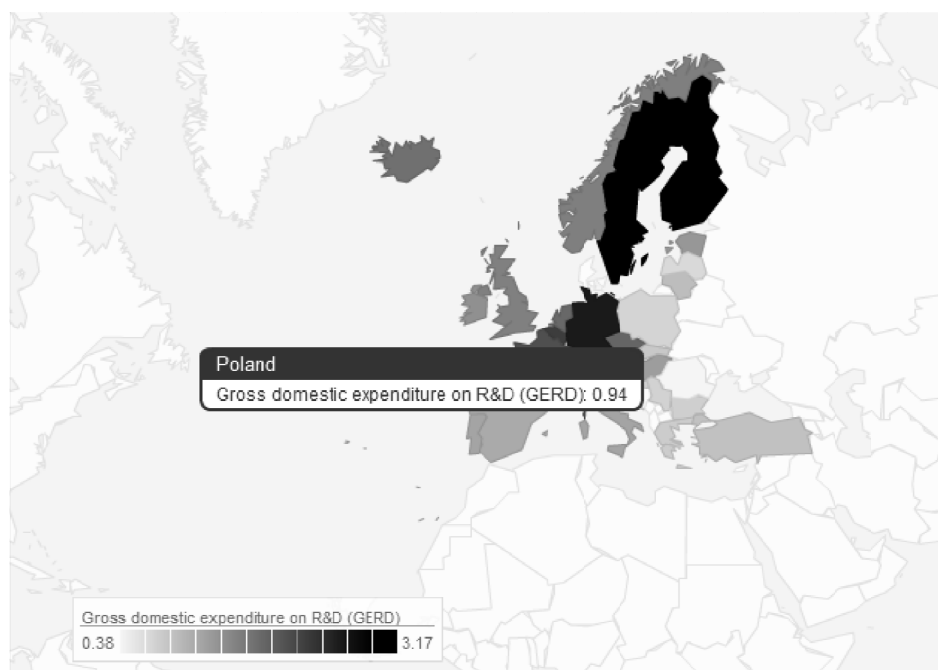
Podstawowym źródłem danych dla wykresów Google Charts jest dwuwymiarowa tabela (macierz) złożona z wierszy i kolumn, osadzona bezpośrednio w dokumencie hipertekstowym (tab. 2). Ponadto wykresy mogą być generowane na podstawie danych umieszczonych w chmurze – Google Spreadsheets lub Google Fusion Tables.

Tabela 2. Macierz danych Google Visualization API

```
var data = google.visualization.arrayToDataTable([
  ['Indicator', 'Headline indicator', '2008', '2013', '2014', '2015', '2020 Target'],
  ['Employment', 'Employment rate, total (% of the population aged 20-64)', 70.3, 68.4, 69.2, 70.1, 75.0],
  ['R&D', 'Gross domestic expenditure on R&D (% of GDP)', 1180.0, 1106.2, 1061.2, 'none', 1086.0],
  ['Climate change & energy', 'Final energy consumption (Million tonnes of oil equivalent)', 1.85, 2.03, 2.03, 'none', 3.0],
  ['Education E', 'Early leavers from education & training, total (% of population aged 18-24)', 14.7, 11.9, 11.2, 11.0, 10.0],
  ['Education T', 'Tertiary educational attainment, total (% of population aged 30-34)', 31.1, 37.1, 37.9, 38.7, 40.0]
]);
```

Źródło: Opracowanie własne z wykorzystaniem Visualization API.

Za pomocą Visualization API utworzyć można zarówno tradycyjne wykresy, w tym kołowe, słupkowe lub liniowe, jak i mapy zjawisk przestrzennych (rys. 3), przez co mogą one np. znaleźć zastosowanie w geomarketingu. Wykresy są generowane w oknie przeglądarki w technologii Adobe Flash lub w formacie SVG czy VML. Ponadto Visualization API pozwala programować dodatkowe funkcjonalności w postaci narzędzi sterujących zakresem danych prezentowanych na wykresie – suwaków i rozwijanych list wyboru (*toolbars*), a także animacji danych (rys. 4). Rozwiązania oparte o Visualization API są relatywnie proste w implementacji i nie wymagają dodatkowych rozszerzeń przeglądarki, jednak mają zamkniętą formę, tzn. użytkownik ma jedynie dostęp do zdefiniowanego zakresu prezentacji oraz wybranych atrybutów graficznych. Brak tu możliwości rozszerzenia funkcjonalności choćby o możliwość chwytania i przeciągania obszaru wykresu lub mapy poprzez np. implementację skryptu czy dopisanie modułu. Jest to jednak rekompensowane wysoką jakością oprawy wizualnej i bezproblemową obsługą.



Rys. 3. Wydatki brutto krajów UE na badania i rozwój, prezentacja w oknie przeglądarki internetowej (tzw. zrzut ekranu)

Źródło: Opracowanie własne za pomocą Charts API na podstawie danych Eurostat [2016].



Rys. 4. Interaktywna prezentacja wskaźników Strategii Europa 2020 za pomocą Google Charts – kokpit menadżerski (tzw. zrzut ekranu)

Źródło: Opracowanie własne za pomocą Visualization API na podstawie danych Eurostat [2016].

Tabela 3. Właściwości narzędzi projektowych odnotowane w trakcie obserwacji poczynionych podczas testów eksploracyjnych

Możliwości projektowe	RGraph	jqPlot	Google Charts, Visualization API	CSS
Wymaga zewnętrznych bibliotek	+	+	+	-
Implementacja zewnętrznych źródeł danych	+	+	+	-
Wiele wykresów na jednej stronie	+	+	+	+
Prezentacja danych w odniesieniu przestrzennym	-	-	+	-
Tworzenie kokpitów (<i>dashboard</i>)	+	+	+	-
Animacje/cieniowanie	+/+	+/+	+/+	-/+
Dynamiczna zmiana rozmiaru wykresu	+	+	+	-
Dynamiczna prezentacja i podświetlanie obiektów	+	+	+	+
Dodawanie lub usuwanie prezentowanych zmiennych	-	-	+	-
Zaawansowane funkcje nawigacji i ustawień wykresu (zmiana koloru, siatki, legendy itp.)	-	+	+	-
Przybliżanie widoku danych, wybór zakresu prezentacji (<i>zooming</i>)	+	+	-	-
Wykresy 3D	+	-	+	-
Możliwość użycia grafik w tle wykresu	+	-	-	+

Źródło: Opracowanie własne.

W strukturze kodu HTML/CSS, wywołującego w oknie przeglądarki wykres utworzony jedynie za pomocą arkuszy stylów, brak jest rzeczywistego odniesienia do danych liczbowych. Technika ta bazuje na pewnego rodzaju „mirażu” generowanych obiektów, które mają zdefiniowany rozmiar i kolor oraz nakładają się na siebie, wywołując wrażenie czy też odwzorowanie wykresu. W rzeczywistości słupki wykresu są obiektami o określonych atrybutach i mogą być zakodowane jako np. tabela (<table>), lista punktowana () lub lista defi-

nicji (<dl>). Wykresy nie są zatem generowane w oparciu o macierz danych lub dane zgromadzone w zewnętrznych plikach, a do ich utworzenia nie są wymagane dodatkowe biblioteki i języki skryptowe. Trudno jednak stwierdzić, że technika ta jest najprostszą z omawianych, bowiem wymaga relatywnie dobrej znajomości kaskadowych arkuszy stylów. Rozszerzenie tak utworzonych wykresów o zaawansowane funkcjonalności, animacje oraz przestrzenne odniesienie danych jest możliwe, jednak po zastosowaniu zewnętrznych bibliotek i języków skryptowych.

Interaktywne wykresy prezentowane w oknie przeglądarki internetowej umożliwiają eksponowanie (również poprzez elementy wizualne, w tym animacje i podświetlanie lub zmianę kontrastu) i porównywanie danych liczbowych (dodawanie lub usuwanie zmiennych). Pozwalają również dobierać zakres prezentowanych informacji, filtrować, sortować, porządkować dane lub prezentować je na życzenie użytkownika, a także tworzyć wykresy z zaawansowanymi funkcjami nawigacji, takimi jak np. możliwość przybliżania lub oddalania widoku danych (*zooming*) oraz przeciągania wykresu w płaszczyźnie poziomej (*panning*), a także zmiany typu wykresu.

Wykorzystanie udostępnianych w Internecie narzędzi powinno być poprzedzone studiami dokumentacji techniczno-projektowej oraz postanowień licencji w zakresie warunków użycia. Wybrane usługi mogą być limitowane. Dla przykładu, licencja Google Visualization API nie przewiduje pobierania i zapisywania kodu niezbędnego do wywoływania wykresów bezpośrednio na dysku twardego i używania go w trybie bez dostępu do Internetu (*offline*).

Podsumowanie

Współcześnie od menadżerów, dziennikarzy danych, marketerów, wymaga się coraz częściej nie tylko wiedzy i umiejętności organizacyjnych, ale również znajomości obsługi oprogramowania branżowego oraz podstaw wybranych technik i narzędzi komputerowych, w tym programistycznych. Jest to związane z gromadzeniem, przetwarzaniem oraz wizualizacją danych, które są nieodzowne w pracy analitycznej.

Opisane w pracy techniki i narzędzia projektowe umożliwiające interaktywną prezentację danych to jedynie wybrane z wielu dostępnych narzędzi. W sieci odnaleźć można kilkadziesiąt innych, przygotowanych i udostępnionych z myślą o tworzeniu i publikowaniu wykresów w oknie przeglądarki internetowej.

Przedstawione techniki i narzędzia projektowe służą przede wszystkim prezentacji danych liczbowych. Wyjątek stanowi Google Visualization API, za po-

mocą którego można prezentować dane w odniesieniu przestrzennym. Ponadto mają one charakter modułarny, przez co są uniwersalne, tj. można je stosować łącznie i rozdzielnie, jako komponent, rozszerzenie dowolnego dokumentu hipertekstowego.

Analizowane techniki i narzędzia są dopracowane i relatywnie proste w użyciu. Ich wykorzystanie w podstawowym zakresie wymaga jedynie wiedzy z zakresu osadzania obiektów w strukturze dokumentu hipertekstowego. Do użytkownika należy konfiguracja parametrów wykresu, wybór formy prezentacji oraz uzupełnienie bazy danych, która przyjmuje zwykle postać macierzy i stanowi integralną część kodu.

Trudno jest jednoznacznie wskazać, które z testowanych narzędzi jest najlepsze, choć za najbardziej profesjonalne i użyteczne spośród testowanych można wskazać Google Charts. Różnice między nimi wynikają m.in. z oprawy graficznej, możliwości projektowych, technik programistycznych, a także postanowień licencyjnych oraz dostępu do wsparcia technicznego. Wszystko to nadaje każdemu z nich unikalny charakter.

Wizualizacja danych nie powinna być celem samym w sobie. Z technicznych możliwości prezentacji danych, w szczególności animacji i interaktywności, należy korzystać z umiarem, tak aby nie doprowadzić do przerostu formy nad treścią.

Literatura

- Afzal W., Torkar R., Feldt R. (2009), *A Systematic Review of Search-based Testing for Non-functional System Properties*, "Information Software Technology", No. 51(6), s. 957-976.
- Agruss C., Johnson B. (2000), *Ad Hoc Software Testing: A Perspective on Exploration and Improvisation*, "Florida Institute of Technology", s. 68-69, <http://cite.seerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.93.2070> (dostęp: 05.03.2018).
- Bennett R., O'Neill E.T., Kammerer K. (2014), *assignFAST: An Autosuggest-Based Tool for FAST Subject Assignment*, "Information Technology and Libraries", No. 33(1), s. 34-43.
- Chhabra M.N. (2012), *Introduction To Adhoc Testing*, "International Journal of Scientific and Technology Research", No. 1(7), s. 66-67.
- Eurostat (2016), <http://ec.europa.eu/eurostat> (dostęp: 16.08.2016).
- Furmankiewicz M., Sołtysik-Piorunkiewicz A., Ziuziański P. (2014), *Zaawansowane techniki graficznej analizy danych epidemiologicznych na kokpicie menedżerskim*, „Informatyka Ekonomiczna”, nr 2(32), s. 64-77.

- Google Finance (2016), *Google Finance: Stock Market Quotes, News, Currency Conversions*, <https://www.google.com/finance> (dostęp: 16.08.2016).
- Itkonen J., Mäntylä M.V. (2014), *Are Test Cases Needed? Replicated Comparison Between Exploratory and Test-Case-Based Software Testing*, "Empirical Software Engineering", No. 19(2), s. 303-342.
- Kampka A. (2011), *Retoryka wizualna. Perspektywy i pytania*, „Forum Artis Rhetoricae”, nr 1, s. 7-23.
- Kaner C., Falk J., Nguyen H.Q. (1999), *Testing Computer Software*, Dreamtech Press, New York.
- Kölling M., Patterson A. (2004), *Going Interactive: Combining Ad-hoc and Regression Testing. In Extreme Programming and Agile Processes in Software Engineering*, Springer, Berlin–Heidelberg.
- Król K. (2015), *Ocena wybranych technik tworzenia interaktywnych map lokalizacji obiektów przestrzennych*, „Acta Scientiarum Polonorum Formatio Circumiecus”, nr 14(4), s. 49-59.
- Król K. (2016), *Data Presentation on the Map in Google Charts and jQuery JavaScript Technologies*, "Geomatics, Landmanagement and Landscape (GLL)", No. 2, s. 91-106.
- Król K., Prus B. (2016), *The Comparative Analysis of Selected Interactive Data Presentation Techniques on the Example of the Land Use Structure in the Commune of Tomice*, "Polish Cartographical Review", nr 48(3), s. 115-127.
- Król K., Salata T. (2013), *Gromadzenie, przetwarzanie oraz wizualizacja danych przestrzennych za pomocą interaktywnych aplikacji internetowych na potrzeby rozwoju obszarów wiejskich*, „Infrastruktura i Ekologia Terenów Wiejskich”, nr 1(IV), s. 195-207.
- Król K., Szomorova L. (2015), *The Possibilities of Using Chosen jQuery JavaScript Components in Creating Interactive Maps*, "Geomatics, Landmanagement and Landscape (GLL)", No. 2, s. 45-54.
- Leszkowicz M. (2011), *Infografika jako forma edukacji w kulturze wzrokocentrycznej*, „Nedidagmata”, nr 31/32, s. 37-55.
- McCandless D. (2012), *Information is Beautiful*, Collins, London, UK.
- Paradowski M.B. (2011), *Wizualizacja danych – dużo więcej niż prezentacja* [w:] M. Kluza (red.), *Wizualizacja wiedzy. Od Biblia Pauperum do hipertekstu*, Materiały konferencyjne, Lublin, s. 40-42.
- Piekarski K. (2015), *Siedem rzeczy, które musisz wiedzieć, zanim zabierzesz się do wizualizacji danych* [w:] K. Piekarski (red.), *Metody badania i odkrywania miasta oparte na danych*, Medialab, Katowice, s. 12-21.
- Rosen L. (2005), *Open Source Licensing: Software Freedom and Intellectual Property Law*, Prentice Hall, Upper Saddle River, New Jersey.
- Rozporządzenie Parlamentu Europejskiego i Rady UE 2015/759 z dnia 29 kwietnia 2015 r. zmieniające rozporządzenie (WE) nr 223/2009 w sprawie statystyki europejskiej.

- Siennicka A. (2015), *Retoryka i wiedza – wielogłos. Retoryka i wizualność*, „Res Rhetorica”, nr 1, s. 74-75.
- Strategia (2010), EUROPA 2020. Strategia na rzecz inteligentnego i zrównoważonego rozwoju sprzyjającego włączeniu społecznemu, Bruksela, 3.3.2010. KOM(2010) 2020.
- Verens K. (2012), *Projektowanie systemów CMS przy użyciu PHP i jQuery*, Wydawnictwo Helion, Gliwice.
- Zhu Y. (2012), *Introducing Google Chart Tools and Google Maps API in Data Visualization Courses*, “IEEE Computer Graphics and Applications”, No. 32(6), s. 6-9.
- Ziuziański P. (2014), *Kokpit menedżerski jako efektywne narzędzie do wizualizacji danych w organizacji* [w:] Z.E. Zieliński (red.), *Rola informatyki w naukach ekonomicznych i społecznych. Innowacje i implikacje interdyscyplinarne*, Wydawnictwo Wyższej Szkoły Handlowej, Kielce, s. 60-70.

COMPARATIVE ANALYSIS OF CHOSEN TECHNIQUES OF NUMERICAL DATA PRESENTATION

Summary: Nowadays, state authorities and private companies collect, store and process large amount of micro- and macroeconomic data. Quite often their analyses are published on the Internet in a form of charts created with various techniques and tools. The paper presents characteristics and the results of comparative analysis of chosen presentation techniques (interactive charts generated in a browser). Conclusion stresses that the techniques differ in many aspects such as: graphics, design limits, coding techniques, licences and technical support.

Keywords: data visualisation, interactive charts, exploratory testing.