

Primena softvera otvorenog koda kod vizualizacije podataka

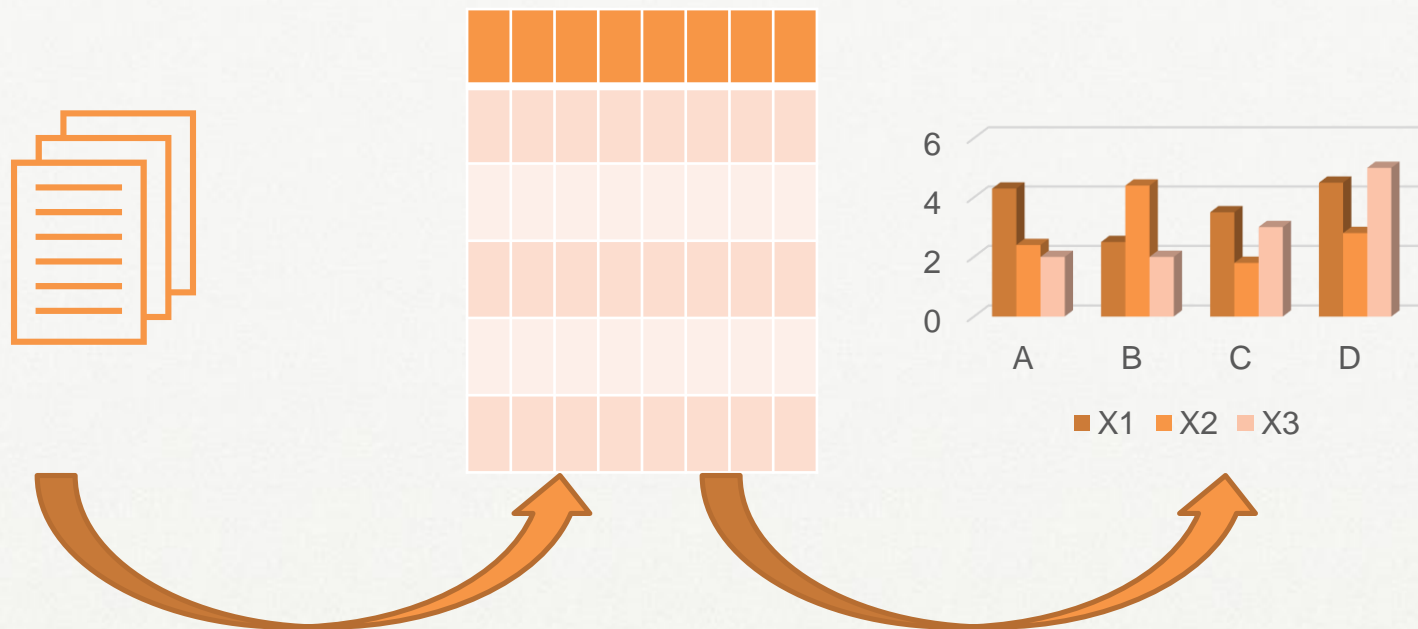


Vladimir Otašević
Biljana Kosanović

Računarski centar Univerziteta u Beogradu

- Vizualizacijom podataka bi se mogla definisati kao tehnika kojom se informacije i podaci grafički predstavljaju.
- Postoji veći broj alata za vizualizaciju podatka:
 - Veliki broj takvih alata predstavljaju biblioteke bazirane na javascript jeziku.
- U nastavku opisane su sledeće javascript biblioteke:
 - D3.js [\[1\]](#)
 - Chart.js [\[2\]](#)
- Navedene biblioteke pored mogućnosti generisanja grafičkih komponenti, nude i određen broj dodatnih funkcija koje su često neophodne prilikom rada sa podacima.

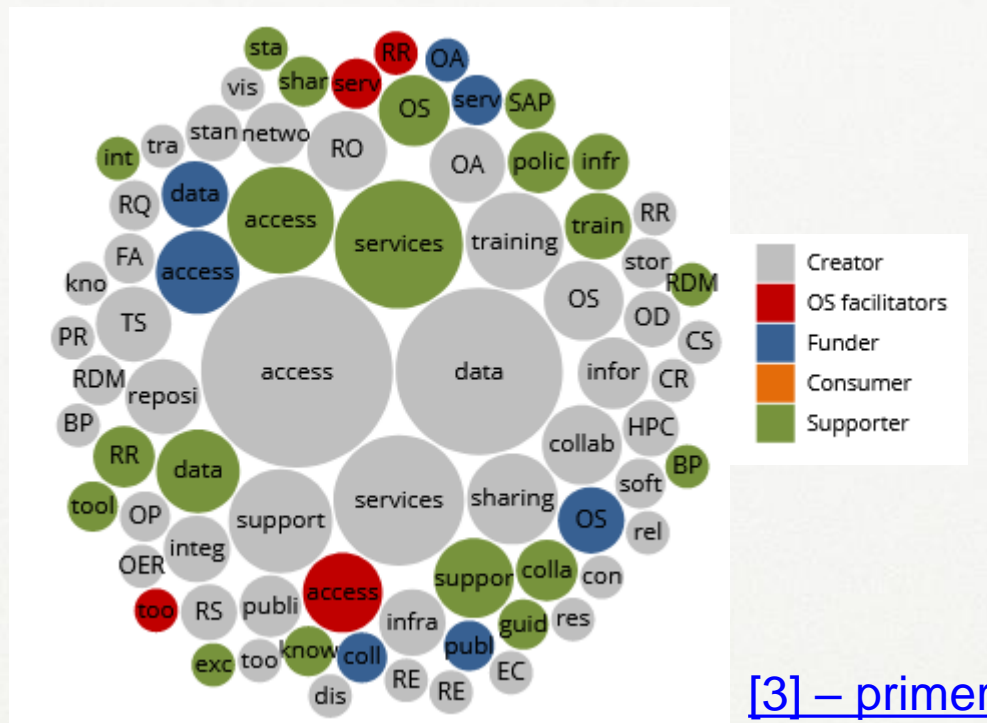
- Prve tehnike vizualizacije podataka se pojavljuju početkom 16. veka.
- Dugo se vizualizacija posmatrala samo kao jedan korak u procesu obrade podataka.
- Danas postoje eksperti za vizualizaciju podataka, a da pri tome ti isti eksperti ne moraju biti stručni za oblast čije podatke obrađuju.



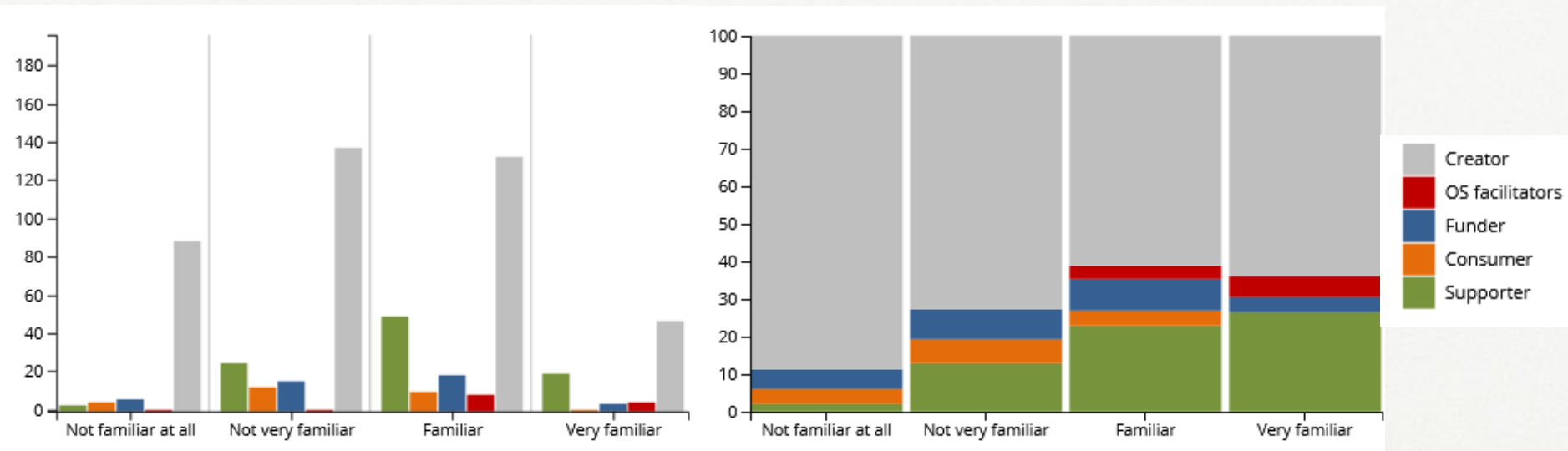
Ilustracija procesa razvoja vizualizacije podataka

- Neki od elemenata koji su sastavni deo vizualizacije su:
 - Boja
 - Oblik
 - Nijanse
 - Interaktivnost
 - ...
- Isti podaci se mogu predstaviti na različite načine u zavisnosti od postavljenih vrednosti parametara.
- Isti podaci se mogu predstaviti na različite načine kako bi se istakle različite osobine podataka ili istakle osobine koje nisu lako uočljive.

- Boja se može koristiti kao tehnika klasifikacije.
- Različitim bojama, se mogu jasno ukazati na postojanje razlike među podacima koji pripadaju različitim klasama.

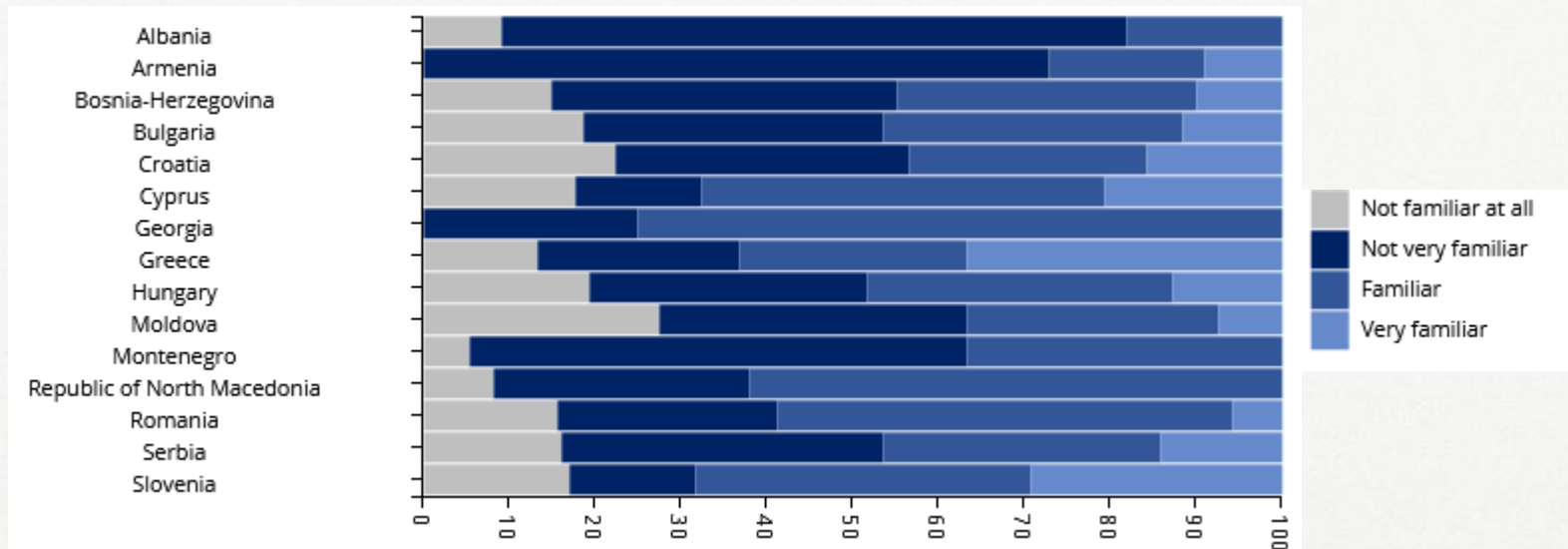


- Kao i boje oblici se mogu koristiti za prikazivanje klasne pripadnosti podataka.
- Za razliku od boja, oblici se najviše koriste kako bi se isti podaci prikazali na različite načine i tako istakle različite osobine podataka.

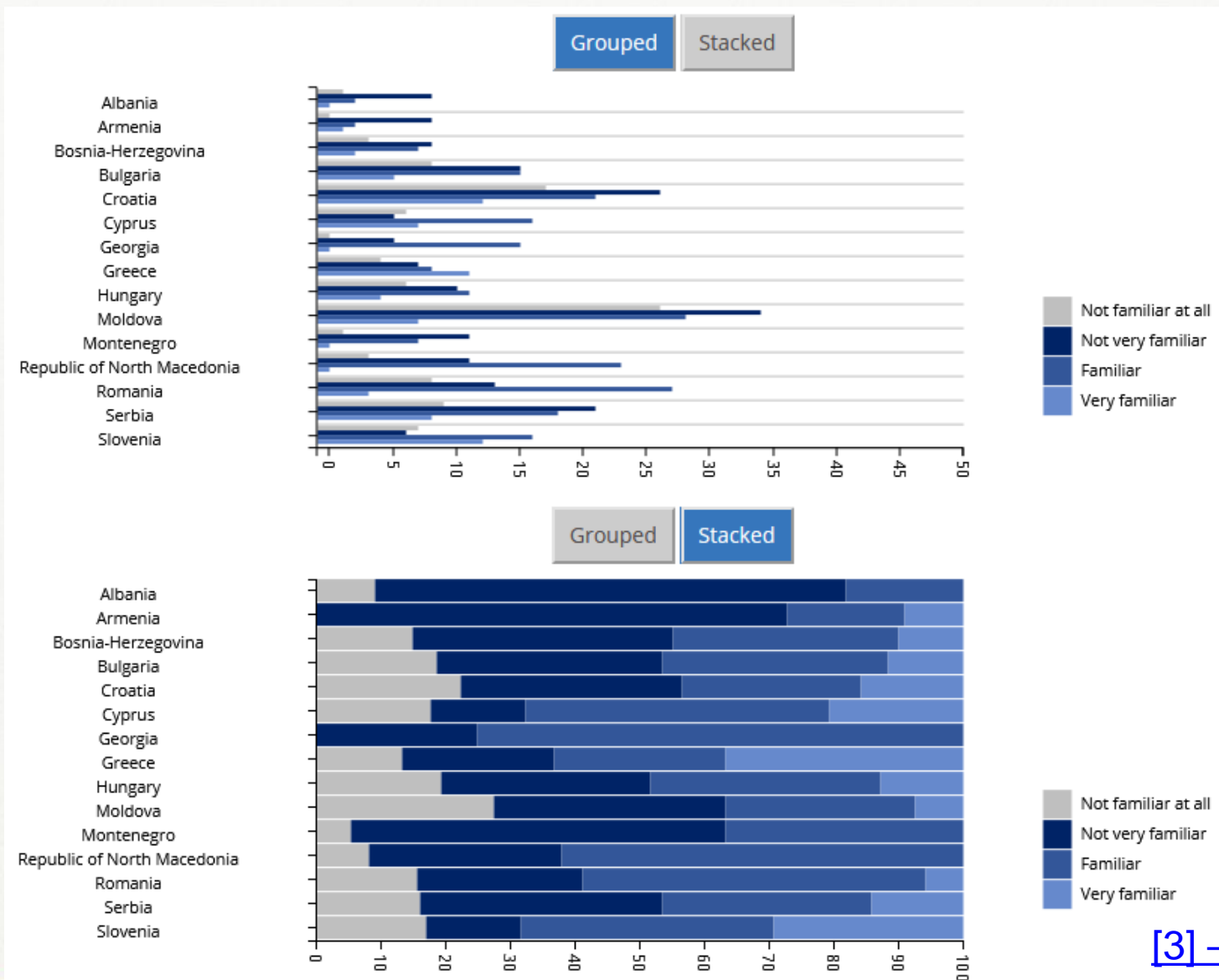


[\[3\] – primer](#)

- Nijanse se mogu koristiti poput boja za isticanje klasne pripadnosti.
- Najčešće tehnika nijansi koristi kada je potrebno istaći da među podacima postoji razlika ali da se vrednosti klasa mogu rangirati po npr. važnosti ili intenzitetu.
- Dobra praksa jeste da skup boja bude jasan i bude u skladu sa okruženjem.



[\[3\] – primer](#)



[3] – primer

- eng. Data-Driven Documents je biblioteka koja se može primeniti kada je potrebno kreirati složeniji oblik grafičke interpretacije podataka.
- Sadrži veliki broj pratećih funkcionalnosti:
 - npr. čitanje i obrada formata fajlova kao što su JSON, csv, psv, ...
- Može se koristiti i sa klijentske strane i sa serverske strane.
- Biblioteku karakterišu 3 osobine i to:
 - Kompatibilnost
 - Mogućnost za obradu grešaka
 - Performanse

D3.js

Zahteva dobre programerske veštine

Vrlo bogata biblioteka

Visok stepen prilagođavanja problemu i podacima

- Posедуje jednostavan interfejs za implementaciju.
- Veliki broj implementiranih primera i jedino je potrebno pripremiti podatke.
- Mogućnost transformacije grafikona.
- Problem: korisnik želi grafikon koji nije predviđen od strane same biblioteke.
- Značajna prednost chart.js biblioteke je jednostavna integracija sa radnim okruženjem kao što su:
 - Angular
 - React

Chart.js

Poseduje manji skup definisanih grafikona

Teže je prilagoditi grafikon konkretnoj implementaciji

Jednostavna za primenu

Lako se integrira sa postojećim JavaScript kodom

- Iako se primenjuju sa istim ciljem, navedene biblioteke se znatno razlikuju.
- Postoje elementi vizualizacije koje nezavisno od konkretnog odabira biblioteke treba pažljivo definisati.
- Navedene biblioteke su samo neka od rešenja koja se mogu primeniti. Ako se doslovno prate pravila otvorenosti, ove biblioteke su svakako rešenja koja treba razmotriti.
- Korisni linkovi:
 - [1] <https://d3js.org/> - D3.js biblioteka
 - [2] <https://www.chartjs.org/> - Chart.js biblioteka
 - [3] Primeri su dostupni adresi <https://ni4os.eu/survey-results/>



Pitanja ?

Vladimir Otašević
vladimir.otasevic@rcub.bg.ac.rs

Biljana Kosanović
biljana@rcub.bg.ac.rs