https://doi.org/10.2352/EI.2023.35.1.VDA-398

Multi-layer visualization for media planning

Marina LJUBOJEVIC, Mihai MITREA; Institut Mines-Telecom, Telecom SudParis, SAMOVAR Laboratory, Palaiseau, France

Abstract

This study deals with data visualization for media planning. Media planning database is a collection of heterogeneous content (image/video/graphics, text, data analytics, logical expressions, ...) to be aggregated, managed, and displayed together. The main contribution is the multi-layer visualization architecture that allows any type of visualization element to be created by its most appropriate libraries. Such visualization ensures spatio-temporal synchronization of displayed content, as well as the proper evolution with the user interaction. The illustrations are provided on two real (yet anonymized) media plans and show how a complex, 7 steps interaction workflow with the media plan can be dealt with.

1. Introduction

Recently, the term MarTech has been coined to cover the business and research efforts situated at the cross-roads of MARketing and digital TECHnologies. The evolution of marketing from analogous to digital, and from mono and multi-channels to omni channels, raised incremental and complementary challenges in the large panoply of tools the marketers should dispose of for achieving their tasks: planning, execution, impact assessment, decision-making and action orientation.

Under the MarTech framework, media planning is the process of strategizing and purchasing advertisement placements by determining the best combination of media to achieve its marketing campaign goals [1]. This concept is very broad and covers multiple scientific fields, from marketing and sociology to information technologies.

From an intuitive point of view, the deployment of a media plan refers to three levels: conceptual formulation, marketing provisioning, and software instantiation. First, conceptual formulation refers to the logical and structured description of the heterogeneous information required for the media plan: the digital advertising content itself, the distribution channels (also referred to as *vehicles* – TV, radio, newspaper, social networking, web adds, …), the time intervals, the distribution targets (audience typology, age segment, *etc.*), the various technical means required for content delivery (network, display devices, *etc.*). Secondly, marketing refers to the financial optimization of the media plan, specifically to the maximization of the return of the advertising campaign under the constraint of fixed total financial cost. Finally, specific software tools are required to allow end-to-end usage of a media plan. This includes the management of the advertising content itself, the interfaces making it possible to carry out this management, and the interactive and adaptative data visualization.

Data visualization in media planning software comes across with incremental constraints. First, it requests heterogeneous content (image/video/graphics, text, data, logical expressions, …) to be aggregated and managed together, as illustrated in Figure 1. Secondly, this data collection targets different professional users with different usage routines and expectations. Finally, the media planner community already has well-established working culture, generally served by fixed environment computing solutions (desktops) in use for several years.

The present paper presents a research study on the possibility of designing a data visualization solution answering the media planning requirements and matched to the mobile device usage peculiarities.

The paper is structured as follows. After the *Problem Statement* (in Section 2), the *State-of-the-art* (in Section 3) presents current software solutions related to data visualization. Then, we advance a solution referred to as *Multi-Layer Visualization* (in Section 4) that allows dynamic and application-oriented composition of content generated by different sources. *Experimental Validation* (in Section 5) is carried out on an anonymized database filled-in with real data. *Conclusion are* drawn and *Perspective* are opened in the last section (Section 6).

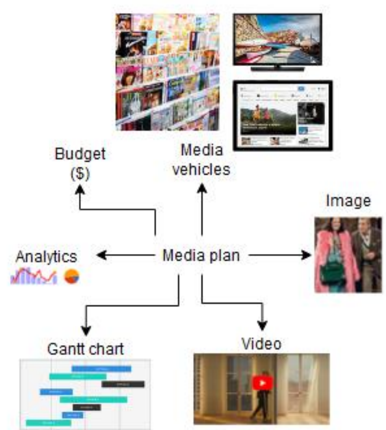

***Figure 1**: Media plan visualization requires heterogeneous data to be timely and spatially aligned according to strict business rules*

2. Problem statement

In a nutshell, media planning is the process of strategizing and purchasing advertisement placements by determining the best combination of media to achieve its marketing campaign goals. While such a definition may be fuzzy or even controversial [1], a mathematical model for media planning was already provided by marketing researchers [2], [3] some 35 years ago.

According to this model, a media planning is a two-folded problem, composed of the media plan itself and of the way in which the media plan can be used. Thus, the media plan is a non-linear function of at least 5 variables (advertising content, distribution channel, distribution time span, price, return on investment). On the top of the media plan, an expert system (e.g. *frame-based*, *rule-based*, …) translating the marketing business culture computes the extremum (or extrema) of the media plan function.

Of course, the computation complexity beneath such a model strongly depends on the accuracy of both media plan and of the expert system rules. To pragmatically estimate this complexity, several methodological solutions coexist, as for instance a multi-object deterministic optimization approach advanced in [4]. It is this brought to light the problem has a combinatory complexity, among the types of elements composing the media plan (number of advertising contents, distributions channels, targets, *etc*.). While the tree representation is intuitively suitable for a media plan, matrix representations for media planning are also available [5].

The present paper deals with media plan visualization in mobile environments. When processing and visualizing this type of data, the challenges are incremental, as they stem from the conceptual, usage, and technical realms.

First, at the conceptual level, the multiple dependencies among variables, as well as the sheer amount of data are required to be represented by complex representations, combining basic data (text, numerical values, image, video) with various types of time-series visualization (plots, Gantt charts, sparklines, *etc*.). According to a rule of thumb in the business culture, a media plan should allow to "*overview first, zoom and details on demand*" [6]. This principle assumes an overview of the data, then the possibility of zooming, and showing the details on demand. Since media planning problem is a multiple variable problem, details asked on demand are giving additional challenge in terms of what, where and how to represent the detail information.

The relevancy of different variables also varies according to the user profile (e.g. media planner, marketing expert or high level manager). Therefore, according to the profile, different visualization methods need to be utilized.

On top of that, the solution should also have a mobile compatibility: so, real-time mechanisms for logical and technical adaptation of the displayed content to resource-constraint environments should also be provided.

The present paper presents a research study on the possibility of designing a data visualization solution providing:

- heterogeneous visualization modes, from basic image/audio/video/text to Gantts charts, pie charts or sparklines;
- interactive content visualization, according to an unpredictable sequence of user requests;
- compatibility to mobile devices.

3. State of the art

The main points to be achieved by data visualization are better processing of information, as well as good user engagement and interactivity. When data is presented just in plain text, extracting all the necessary information is not obvious, as there is a need for comparing multiple seemingly disconnected data points. The task is more complicated if user has to analyse and obtain information out of large volumes of data, like in media planning case. The objective of data visualization is making key trends and patterns in data directly observable [6].

In conjunction with actively communicating the meaning of the data, the user interactivity should be natural and intuitive. This interaction can help uncover insights which leads to better, data-driven decisions. In case of media planning, where each data entry contains multiple components, a solution needs to be found for visualizing basic components first, then going to the higher level components based on user interaction [7].

Components in a media plan are grouped by level of information they are conveying, and suitable visualization solutions are found for each group. Groups with time series are suitable for Gantt charts, while financial components can be represented via multiple other types of graphs – bars, pie charts, *etc.* To show massive data on the same page, sparklines are also considered as a visualization option. As data is both massive and complex, possibility of combining multiple types of visualization is explored.

When it comes to all those types of visual representations, different software solutions belonging to JavaScript family can be considered.

As both visualization modes and their underlying technologies are of various typologies, we started the state-of-the-art study by considering the simplest case, as presented in Table 1.

Table 1 synoptically provides information about: (1) two most commonly used visualization modes related to media planning (represented on the rows), that is Gantt chart and sparklines and (2) five JavaScript based visualization libraries (represented on the columns), selected according to their popularity at the time of the study, namely: D3 js [8], AnyChart [9], Highcharts [10], Chart js [11], Chartist [12].

D3 js is a library with high interactivity level, however complexity rises when it comes both to utilizing all possible interactivity options or combining them with another technology. It does not have native Gantt chart support. Instead, each chart is created element by element, which gives code complexity, especially in case of media planning, with its extensive data. As it uses document object model manipulation, having large volume of data results in significantly slower chart rendering compared to some other libraries.

AnyChart and Highcharts offer a general-purpose interactivity level: while they are outperformed by D3 js in this respect, they have the advantage of being easier to be combining to other technologies. Gantt charts rendering does not get significantly slower with larger amount of data of media planning. Charting options regarding Gantt charts are extensive as well. Moreover, additional interactive options are possible to be added, with no greater complexity than compared to D3 js library.

Chart js and Chartist did not support Gantt chart creation, so they were tested for sparkline add-ons instead, with Chartist giving better native support and diversity.

Aside for possibility of combining different technologies, selected ones need to ergonomically represent requested data, be it Gantt, bar, pie, financial chart *etc.* As Gantt chart for a time series in media planning is needed, selected library should have possibility of both plotting that type of charts, as well as possibility of having different technology built on top of it. Highcharts are successfully tested regarding both of those options. As most libraries do offer graphs like bar chart and pie chart, more focus is on the sparklines for the next visualization level. Diversity of sparklines is good utilizing all mentioned libraries, however upon testing, jQuery gives very diverse high interactivity sparkline options, with diversity being significantly higher compared to most other tested libraries, and utilization complexity being lower.

Table 1: Off-the-shelf libraries for common data visualization techniques (Gantt charts and sparklines)

		D3 js	AnyChart	Highcharts	Chart js	Chartist
Gantt charts	Native support	☹️	🙂	🙂	☹️	☹️
	Interactivity	🙂	😐	😐		
	Code complexity	☹️	🙂	🙂		
Sparklines add-ons	Native support	🙂	🙂	🙂	😐	🙂
	Diversity	🙂	😐	🙂	😐	🙂
	Interactivity	🙂	🙂	🙂	☹️	☹️
	Code complexity	☹️	😐	😐	😐	🙂

The results of this preliminary analysis are synoptically presented in Table 1 in graphical symbols: happy green face, neutral yellow face, and sad red face. Happy green face corresponds to a functionality that is natively supported. Neutral yellow face is used in cases where there is a possibility of creating such a solution, however questions about the code complexity arise. A sad red face is used to depict the cases in which inner technological limitations avoid the targeted functionality.

It can be thus seen that popular current-day technologies cannot cope with all the expectancies of our application, in the sense that inner limitations in jointly representing Gantt charts and sparklines are met for any of the five investigated libraries. In practice, such limitations are expected to be even stronger, as more visualization modes are to be dealt with: as explained before (Figure 1), image/video are expected to be presented alongside Gantt charts, statistical measures (average, correlations, *etc.*) are expected to complement sparklines, *etc.*

Hence, despite the large variety of off-the-shelf software libraries, we cannot expect any individual solution to solve the media plan problem in its general aspect. In order to avoid extensive software development required to upgrade individual solutions to the level we need, as well as the subsequent technological lock-in for such an approach, our study considers the possibility of bringing together various technologies with their own benefits. This way, we advance *Multi-Layer Visualization* (as described in Section 4) that allows any type of visualization element to be created in its most appropriate libraries while ensuring their spatio-temporal synchronization, as well as the proper evolution with the user interaction.

4. Advanced solution: media plan Multi-Layer Visualization

To allow any type of visualization element to be created in its most appropriate library while ensuring the overall spatio-temporal synchronization, we designed the architectural framework illustrated in Figure 2.

Figure 2 illustrates the media plan information workflow, from raw data stored in the media plan to the visual content interactively displayed on the user device.

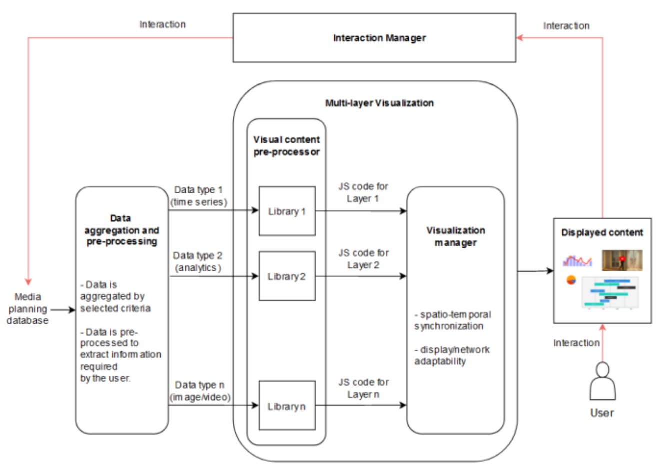

Figure 2*: Multi-layer visualization workflow*

According to this workflow, media plan data is processed by two blocks, referred to in Figure 2 as *Data selection and aggregation* and *Multi-Layer Visualization*. Figure 2 also includes an *Interaction Manager* module, that captures and processes the events generated by the user when interacting with the displayed content.

The workflow presented in Figure 2 is executed in loops. First, an initialization is made based on a preestablished rule. Then, upon user interaction with displayed content, request for the new information is sent back through the Interaction Manager. Updated data is then aggregated and processed again for the new visualization. This type of workflow is based on basic principles of data visualization: *overview first, zoom, and details on demand.*

4.1 *Data selection and aggregation*

The processing workflow starts by considering as entry data the media plan database. We represented the media plan as a tree-structure (as previously shown in Figure 1), with various information on the campaign such as budgeting plans, chosen media vehicles, audio-visual data (video, image), etc.

According to the user request (or according to the initialization rule), selected information about media plans is fed to *Data Aggregation and Pre-processing*. Default aggregation is done according to the vehicle type: duration of planned campaigns of each brand for each type of vehicle, where timeline for each specific vehicle is aggregated.

Aggregation can also be done by brand to give insight into planning for various vehicles over some time period, or by other selected criteria.

These information is then presented (in raw or formatted format) to the next block.

4.2 *Multi-Layer Visualization*

In its turn, the Multi-Layer Visualization module is composed of two sub-modules: *Visual Content Pre-processor* and *Visualization Manager*.

4.2.1 *Visual content pre-processor*

The block integrates a set of existing data visualization libraries that are required for media plan displaying. Each such a library can have a specific role, from time-series visualization to Gant charts creation and video decoding.

For instance, Highcharts is used for time-series data and Gantt charts, while jQuery is used for data related to analytics and sparklines.

Inputting data into these libraries is also part of the pre-processing of the visual content, as aggregated data should be processed to fit the input requirements of visualization library.

The output consists of JavaScript codes for each layer.

4.2.2 *Visualization manager*

Visualization manager takes these outputs and processes them while ensuring the spatio-temporal synchronization and providing display and network adaptability to produce displayed content.

To ensure spatial synchronization, we design an algorithm having 3 steps:

- spatial coordinates specific to each library are first extracted; since in most libraries those coordinates are not part of the API, they need to be extracted from the document object model (DOM) tree;
- positions for the next layer are computed according to the extracted coordinates; each computation is done while taking care of relative distances of the content to other parts of the page;

- according to the computed positions, objects of the next layer are then added in separate divisions to the visual representation.

The *Visualization manager* output consists in the content to be directly displayed on the user device.

4.3 *Interaction manager*

Interaction manager supports two types of interaction. The first one is the general operating system interaction for launching the application, closing/resizing/repositioning the application window, *etc.* These interactions are as provided by the operating system and our work relates to the management of the mouse position clicking. Second type is enhanced interaction for *zoom first, details on demand usage*. This interaction is specifically designed in our study to trigger advanced visualization modes, like adding analytics over an already exiting graphical content. Three modes of interaction of this type are currently programmed:

- interaction buttons applying the second layer of visualization to all elements in the initial visualization;
- clicks on predefined parts of the initial visualization layer;
- resizing/repositioning in the context of Multi-layer visualization (that is, windows containing content managed by different libraries).

5. Experimental illustrations

5.1 *Experimental testbed*

The experimental testbed is composed of:

- a server hosting the databases. Two different databases, with 1564 and 7103 entries are processed. These two databases share most of the main variables of media plan (brand, vehicle, start date, finish date, budget, *etc.*), so fist layer Gantt chart is created in similar way for both of them. However, as some additional variables (target audience, location, creative reference, buying method, *etc.*) may vary, different types of deeper layer analytic data are examined in respect to each of the databases. Although these databases contain real data, our illustrations correspond to anonymized data (to avoid any potential conflict of interest). The database server is inquired by JSON requests;
- a server hosting the implementation of the Multi-layer visualization solution; specifically, HTML5, JavaScript, Highcharts, jQuery. HTML 5 with JavaScript is chosen, as it can provide interactivity, which is needed in representing such a content. Libraries such as Highcharts and jQuery are chosen because they natively support high-level interactivity and possibility of adding new interactivity levels. Content is SVG based, so rendered graphics are natively scalable, keeping high quality at any resolution, and it is possible to add layers of interactivity over objects contained in the graphic.

5.2 *Experimental illustrations*

The usage of the Multi-layer visualization will be illustrated by seven-steps visualization process on a media plan, as illustrated in Figure 3.

This illustrations process follows the overview first, zoom and details on demand. Time-series visualization (step ①) represents "overview first", and upon interaction with it, "zoom" and "details on demand" are presented (steps ② - ⑦). Steps ② and ③ show how details can be (iteratively) added on a basic visualization. Step ④ shows how a basic visualization can be expanded. Finally, steps ⑤ – ⑦ show how details and expansions can work together.

The "*Overview first*", i.e. Step ①, is illustrated in form of a Gantt chart with a table, as presented in Figure 4. Each row of the table contains information about advertisement plan, namely brand that is advertised, throughout multiple vehicles, which can be magazine, TV, radio, web page *etc.*, as well as start date and end date of the marketing campaign. Considering complexity of the data and the fact that each media plan "entry" consists of both type of vehicle and the vehicle itself, the Gantt is presented by aggregation of vehicles by each of the requested brands. As this layer is aggregated over time by type of vehicle, there are no specific vehicles shown at this point. Instead, information about categories are provided: online vehicles, national radio stations, all national magazines, all national TV stations.

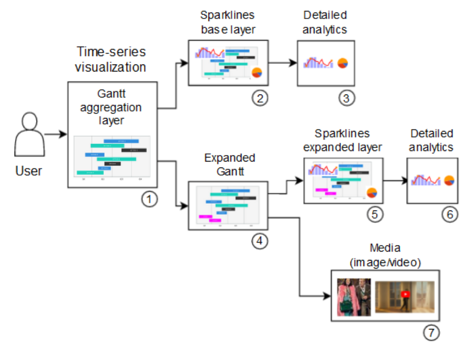

Figure 3: *Media plan visualization in 7 steps*

The "*zoom and demand on request*" principle (i.e. Steps ② - ⑦) is a kind of "smart zooming" providing a more detailed information. For instance, by selecting one bar on the Gantt chart, the complex information is spread into next level with new details. Figure 3 presents workflow for such interaction, while Figures 5 – 11 presents various type of details thus obtained.

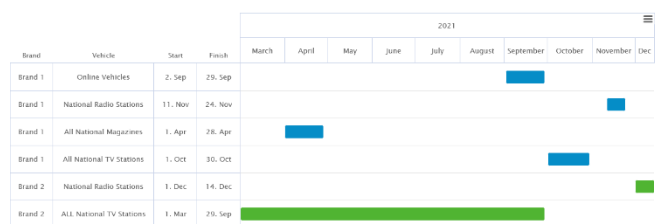

Figure 4: *First (initialization) content: a Gantt chart and its table. The bars in the Gantt chart are interactive (by a sequence of clicks detailed information can be provided)*

To represent additional requested information, sparkline representations are added over the first layer at this point, *cf.* Step ② in Figure 3. Sparkline layer appears by selecting analytics option (button) next to the Gantt chart. It is added over bars of a Gantt chart. As shown in Figure 5, sparklines can be any types of chart to represent additional information: bar chart, pie, horizontal bar, *etc.* Since it is a case of media planning, those charts can represent any additional type of information contained in a media plan, with criteria by budget, media, calendar year, season, vehicle category, region *etc*. Figure 5 firstly shows two bars – blue and dark grey, which are representing parts of a Gantt chart created with Highcharts library. For each bar, spatial anchors, need to be computed according to the coordinates of the bar on a Gantt chart. Spatial anchors are calculated while taking care of the relative distances of bar compared to the starting coordinates of the Gantt chart, and position of the Gantt chart in respect to the page itself. As these distances are not part of the API, they are extracted from the document object model tree of a chart.

Figure 6 shows the base Gantt aggregation layer on the top of which sparklines are added.

The "*details on demand*" concept can be iteratively applied. For instance, Figure 7 corresponds to Step ③ and shows how insights about a sparkline can be provided. In the example, a conventional zoom, with interactive elements, is visualized. Chosen sparkline at this level is a pie chart, which shows vehicles utilization by percentage.

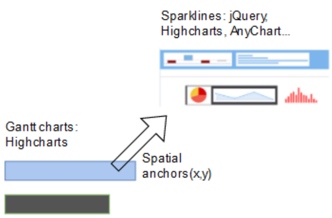

***Figure 5**: Adding the sparkline layer*

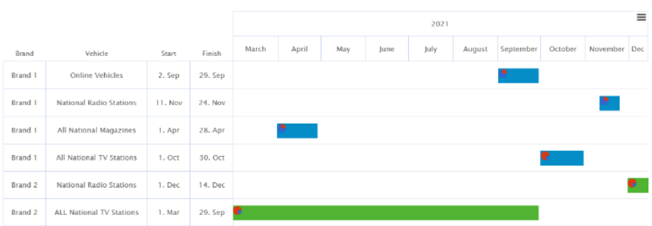

***Figure 6**: Sparklines over base layer*

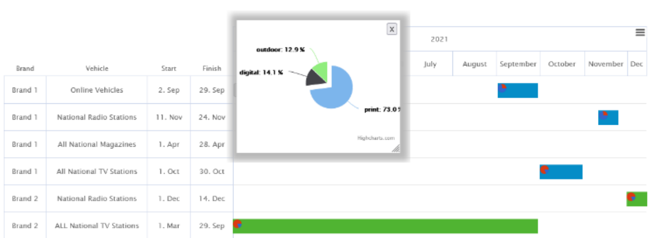

***Figure 7**: Simple zoom over a sparkline layer*

The Step ④ shows the possibility for expanding the aggregated media plan. For instance, the last row in Figure 4 (in green bar) corresponds to aggregated campaigns for brand 2 for all national TV stations, lasting from March 1st until September 29th. Clicking on the mentioned green bar results in Figure 8. Figure 8 shows separately two previously aggregated campaigns and gives the exact TV channels and advertising campaign dates for each one. They are represented in orange bars at the last two rows of the Gantt chart. Campaign for TV channel 1 lasts from March 1st until June 30th, and for TV channel 2 from June 24th until September 29th. In this case, two campaigns were aggregated, however, this can be done for multiple aggregated campaigns.

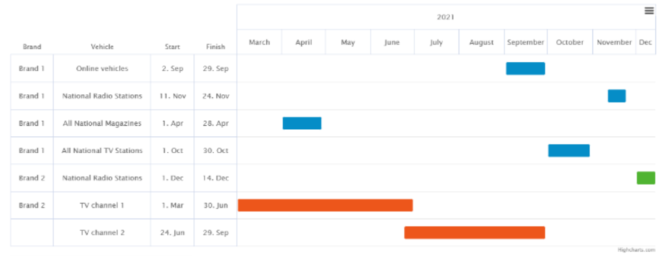

***Figure 8**: Expanded layer*

Figure 9 corresponds to Step ⑤ and demonstrates that expanding and details mechanisms can be combined. To this end, sparklines are added over the expansion. As it may be of particular notice, different types of sparklines correspond to different levels of expansion. For still aggregated parts – whole Brand 1 (represented with blue bars) and Brand 2 for National radio stations (green bar), sparklines are the same pie charts as at the previous level. New expansion levels – Brand 2 for TV channel 1 and TV channel 2 give possibility for new, more detailed analytical data, represented with bar charts.

Figure 10 corresponds to Step ⑥ and is showing analytics level over expanded Gantt. It is a bar chart with detailed budget planning analytics for Brand 2.

Figure 11 corresponds to the Step ⑦ and shows media layer with image and video alongside expanded Gantt. Media layer is added at this point, because information about Brand 2 is expanded enough to get to the un-aggregated core version of media plan, where this information is located. At this level of expansion, only possible media plans for displaying media layers are related to Brand 2 with vehicles being TV channel 1 or TV channel 2.

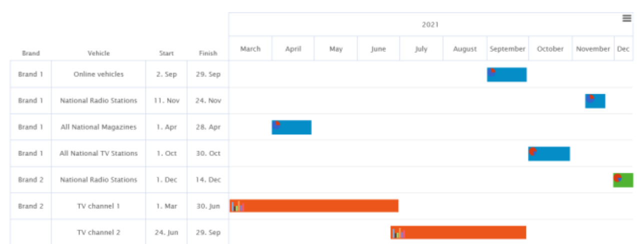

***Figure 9**: Sparklines over expanded layer*

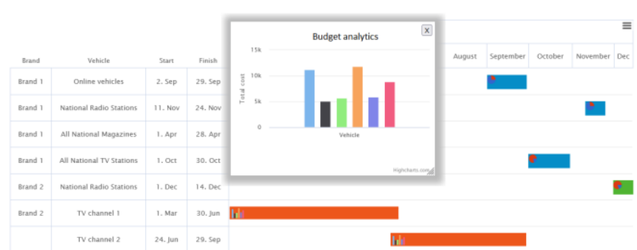

***Figure 10**: Budget analytics over expanded layer*

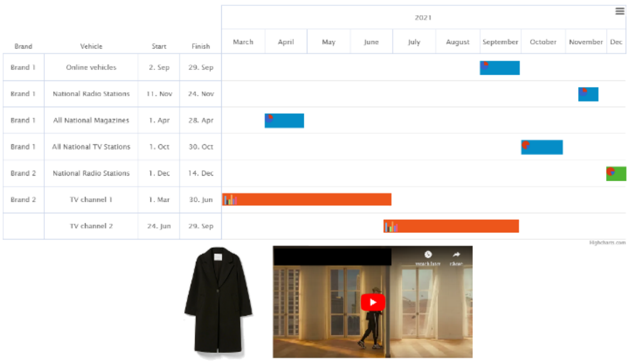

***Figure 11**: Media layer*

6. Conclusion

Media plan is a complex structure composed of heterogeneous information such as text, graphics, analytics, image, video, *etc.* Visualizing multiple plans over time would require interactive content visualization, according to an unpredictable sequence of user requests. To solve this, the intuitive solution would be to choose a mobile development technology and to develop the code. Such a solution cannot be accepted as (1) there is no technology able to cater for all of the needs of media plan visualization, hence the required code would be too complex, and (2) it would result into a technology lock-in, with no possibility for future extensions. Our study follows a different approach: it considers the JavaScript as a basic technology and creates the architectural framework (Multi-Layer Visualization) that allows different libraries, matched to different types of visual representations to be integrated together. Multiple libraries are examined, and proof of concepts for our study is created by using Highcharts and jQuery libraries. The PoC is done on two real databases, with our illustrations presenting anonymized version of data. Such solution has following advantages. First, the code complexity is kept low. Secondly, the displayed content is SVG-based, so natively rendered graphics are scalable, keeping high quality at any resolution and making it possible for complex interactivity mechanisms to be added over objects contained in the graphic. Finally, the modularity is ensured in a sense that new libraries can be added or obsolete one can be replaced without changing the other modules. Future work would include examining more deeply the analytics layer according to the relevant information in the media plan, as well as improving the interactive manager by adding new complex interactions and more options. Carrying subjective evaluation campaigns for the user quality of experience with the multi-layer visualization is also part of our future work.

References

[1] https://www.thebridgecorp.com/media-planning/ (last visit: 09/18/22)

[2] A. Mitchell, "The use of alternative knowledge-acquisition procedures in the development of a knowledge-based media planning system", International Journal of Man-Machine Studies, vol. 26, no. 4, pp. 399-411, 1987.

[3] A. Defrance, "The enunciative contract and media planning", International Journal of Research in Marketing, vol. 4, no. 4, pp. 311-330, 1988.

[4] P.C. Jha, R. Aggarwal, A. Gupta, "Optimal media planning for multi-products in segmented market", Applied Mathematics and Computation, vol. 217, pp. 6802-6818, 2011.

[5] U. Kulbida, O. Kaneva, A. Zykina, "Media planning optimization treatment", Dynamics of Systems, Mechanisms and Machines, 2014

[6] H. Shen, T. Bednarz, H. Nguyen, F. Feng, T. Wyeld, P.J. Hoek, E. Lo, "Information visualization methods and techniques: State-of-the-art and future directions", Journal of Industrial Information Integration, vol, 16, 2019

[7] U. Rashid, M. Viviani, G. Pasi, "A graph-based approach for visualizing and exploring multimedia search result space", Information Sciences, pp. 303-322, 2016

[8] https://d3js.org/ (last visit: 09/18/22)

[9] https://www.anychart.com/ (last visit: 09/18/22)

[10] https://www.highcharts.com/ (last visit: 09/18/22)

[11] https://www.chartjs.org/ (last visit: 09/18/22)

[12] https://gionkunz.github.io/chartist-js/ (last visit: 09/18/22)

Author Biography

Marina Ljubojevic received her BS in electrical engineering from the University of Belgrade (2018) and MsC in multimedia networking at Telecom Paris (2020). She is currently a PhD student with Telecom SudParis. Her PhD program is founded by info3 (https://www.infocubed.com/); yet, the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest

Mihai Mitrea holds an HDR degree Pierre and Marie Curie University in Paris (2010) and a PhD from Politehnica University in Bucharest (2003). He is currently Associate Professor at Telecom SudParis. He is vice-president of the Cap Digital's Technical Commission on Digital Content and serves as advisor for the French delegation at ISO/IEC JTC1 SC29 (a.k.a. MPEG)