SoftwareX 26 (2024) 101729

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original Software Publication

Snap4City dashboard manager: A tool for creating and distributing complex and interactive dashboards with no or low coding

Pierfrancesco Bellini, Marco Fanfani, Paolo Nesi *, Gianni Pantaleo

DISIT Lab, Department of Information Engineering, University of Florence, Via S. Marta 3, Florence 50139, Italy

A R T I C L E I N F O

Keywords:
Dashboards
Business intelligence
Visual analytics
Widget
Iot
Digital twin
Smart city
Business logic

A B S T R A C T

Dashboards are fundamental to inspect data in smart city and industrial contexts. Due to the complexity of such interfaces, dashboard design and construction can be a difficult and time-consuming activity, usually requiring coding competences, not always available among users. In this paper the Snap4City Dashboard Manager is proposed. Unlike other available solutions, the Dashboard Manager allows users to create dashboards in a graphical way by composing widgets able to display multiple kinds of data. Additionally, each widget can be extended to both retrieve data from any source and enable event-driven communications among different widgets and users.

1. Metadata

Nr	Code metadata description	*Please fill in this column*
C1	Current code version	*v7.7.2*
C2	Permanent link to code/repository used for this code version	https://github.com/disit/dashboard-builder https://hub.docker.com/r/disitlab/dashboard-builder/
C3	Permanent link to reproducible capsule	https://www.snap4city.org
C4	Legal code license	GNU AFFERO GENERAL PUBLIC LICENSE, Version 3.
C5	Code versioning system used	git
C6	Software code languages, tools and services used	JAVA, Javascript, PHP, HTML, CSS, SQL
C7	Compilation requirements, operating environments and dependencies	JAVA7 or above, MySQL 5.5 or above, PHP 5.7, Apache HTTP Server, Linux or Windows environment
C8	If available, link to developer documentation/manual	https://www.snap4city.org/drupal/sites/default/files/files/Dashboard%20Manager%20-%20Technical%20%20documentation.pdf https://www.snap4city.org/drupal/sites/default/files/files/Dashboard%20Manager%20Db%20Documentation.pdf training: https://www.snap4city.org/944 https://www.snap4city.org/download/video/course/p2/Snap4City-p2-Dashboards-v12–6.pdf https://www.snap4city.org/download/video/course/p2/
C9	Support email for questions	paolo.nesi@unifi.it

2. Motivation and significance

The growing attention to smart cities and communities and the increasing production of data of multiple kinds, models, and shapes has led to a great effort in developing systems and platforms able to efficiently handle and present such information. Data ingestion and processing (transformation, transcoding, etc.) are relevant and enable the production of predictions, detection of anomalies, generation of hints, etc. The visualization and interactive navigation of the above-mentioned hints, detected anomalies, predictions and derived data/results are fundamental for both public administration and industrial decision-makers who have to extrapolate some considerations and come to a decision. To this end, much attention has been moved from simple data presentation to actual visual analytic and business intelligence tools for control and planning, including what-if analysis. Visual analytics and insights allow to better understand data and extract knowledge out of

* Corresponding author.
E-mail address: paolo.nesi@unifi.it (P. Nesi).

https://doi.org/10.1016/j.softx.2024.101729
Received 27 January 2024; Received in revised form 5 March 2024; Accepted 29 March 2024
Available online 12 April 2024

them, with the goal of supporting decision-making processes. The increasing popularity of these tools has extended their use in several heterogeneous contexts, involving very different user profiles. In this context, it is essential also to consider final users, with the requirement of managing different complexity levels in the development of dashboards and visual tools.

In most cases, dashboards for decision makers can be created by exploiting business intelligence tools with classic faceted indexing and search to access data, as for example SOLR [1] or Kibana/Grafana [2]. Such solutions focus on single views of data, filtering, and drilling down, rather than representing key performance indicators (KPIs) and status and are limited on the data type they can manage, and on the widget types they can offer and build in the user interface. Solutions such as SpagoBI [3] and OpenDataSoft [4] have limited rendering capabilities, insufficient to build smart city dashboards. Indeed, the necessity of tailoring dashboards on specific user requirements is a relevant topic, as highlighted by recent literature surveys [5]. For these reasons, custom solutions have been developed for specific cases, as for example for London [6], Amsterdam and Duisburg [7], Dublin [8], Milan [9]. Such solutions being developed for specific scenarios do not provide general dashboard building tools and therefore require long development times and high costs. Similarly, research projects have investigated on smart city dashboards [10] without providing specific tools for building them. More interesting are approaches like Cities-Boards [11], where authors have proposed a dashboard building framework; however, the process is based on a graphical programming language that is successively transformed into a web interface, therefore requiring programming competences to start with. Similarly, in [12], dashboards are generated by serializing information expressed in knowledge graphs. In [13], a tool based on the definition of meta-models to build dashboards using a code-generator has been proposed. Such solution does not require deep programming competences and users have to learn a modeling tool in order to select data, define widgets (with their properties), and manually connect each variable to the right widget input. The capability of automatically associating any data kind to the right visualization modality is a relevant topic in dashboard building as stated in [14]. For example, in [15], data kind-widget association has been provided by exploiting a semantic reasoner that selects the best visualization modality, given a data input. In [16], authors have proposed the PTAH meta-model to design dashboards and address the selection of a correct widget using feature models to describe all the available widgets and relate them to the PTAH concepts. More recently, some smart-city framework solutions have been proposed for the development of dashboards, for example as paid solutions like Wisetown [17], others as open source are still not accessible to be downloaded and used [18]. Briefly, there is not any open source easy-to-use dashboard building and management system, which is general enough to be deployed in different smart-city or industry scenarios, so as to be used without any programming competence for creating simple solutions; there is no dashboard building and management system which may need only small coding, when it comes to creating complex visual analytic and business intelligence tools. Moreover, in the current state-of-the-art most solutions do not offer the possibility of showing3D data.

Therefore, there is the need to create meaningful graphic and custom interactive representations (beyond the classic GUIs and dashboards), with visual intelligence and business intelligence tools, providing capabilities for displaying 2D maps, KPIs, time series, and other kinds of plots and diagrams, with high level of interactivity and customization of the business logic, to implement what-if analysis tools, and to enable 3D visualization of advanced interfaces, with local and global digital twins interactive representation [18,19]. The Snap4City Dashboard Manager, as presented in this paper, covers multiple domains addressing use cases as for example the observation of current status and the execution of prediction or reconstruction exploiting what-if analysis, in industrial and smart-city contexts. The produced Dashboards provide visual tools representing data as: multi data maps, time series, vectors, tables, heatmaps, etc., and complex data for specific domains, such as: mobility and transport (traffic flow density network and animations, origin destination maps and animations, trajectories); energy (simulation of energy production with PV panels); environmental (the widespread presence of pollutant in a 3D shape of the city, and its related animation); 3D representation with what-if analysis to observe the impact of possible changes, etc. Moreover, a Dashboard Builder has been realized in order to facilitate and speed up any visual tool construction. It consists of a set of web tools to build connected and intelligent dashboards/visual tools through the composition of several kinds of widget in a completely graphical way, thus requiring programming only when custom business logic is needed, thus becoming more accessible to a wide range of users. The Dashboard Builder allows to create simple or advanced dashboards according to the target audience and the specific application scenario and it offers a wizard which can shorten any dashboard production by reducing the time to associate data with graphic representations, thanks to the data semantic modeling exploitation, as reported in [20].

The Dashboard Manager and Builder are modules of the Snap4City open-source platform (www.snap4city.org) [21]. Dashboard Manager is able to manage multiple organizations as tenants and billions of data with its key focus on interoperability. At present, it is in operational use in several Smart Cities, industries, and areas within Italy (Firenze, Pisa, Livorno, Prato, Lonato del Garda, Modena, Merano, Cuneo, etc.) and Europe, too, (Antwerp, Santiago De Compostela, Valencia, PontDuGard-Occitanie, Dubrovnik, Mostar, and West Greece, etc.). The largest installation of the platform is a multi-tenant with 19 organizations and more than 8000 operators / developers, more than 1700 Dashboards, among which about 280 are public. The solution has been installed in almost all continents and the Dashboard tool has been downloaded from GitHub and included in Snap4City tools thousands of times.

The paper is organized as follows: in Section 2 the architecture and the main functionalities of the Dashboard Manager and Builder are presented. In Section 3, some examples on the usage of the Dashboard Builder are provided, while in Section 4 impacts of such proposed software are discussed. Finally, in Section 5 conclusions are drawn.

3. Software description

In this section, both architecture and functionalities of the Dashboard Builder are presented. The Dashboard Builder is the core part of the Snap4City Dashboard Manager, which collects and distributes dashboards on demand. Dashboards are organized in Organizations which are tenants and can be shared, cloned, and delegated in access among users. The Dashboard Builder provides a set of tools for creating dashboards which can provide information and interaction tools to users and receive and send data from/to the platform and from/to third parties solutions.

4. Software architecture

The architecture of the Dashboard Builder is represented in Fig. 1. The Dashboard Builder is composed by three main blocks: the **Widget Collection**, the **Dashboard Wizard**, and the **Dashboard Editor**. The **Dashboard Editor** is used to create/modify dashboards (including their logic, visual analytics, what-if tools, etc.), by collecting and configuring Widgets and their relationships, sizing and placing them into dashboard canvas [20,22]. Each widget has a number of capabilities in presenting data, collecting data and interacting with users and protocols.

The **Widget Collection** includes several ready-to-use widgets and custom widgets (that can be created for implementing new interactive graphic representations and Synoptics by using any SVG graphic editor). Each **Widget** is realized as an independent module which can: (i) present information to the user, (ii) get actions/interactions from the user, and (iii) interact back and forward with different channels. Channels are

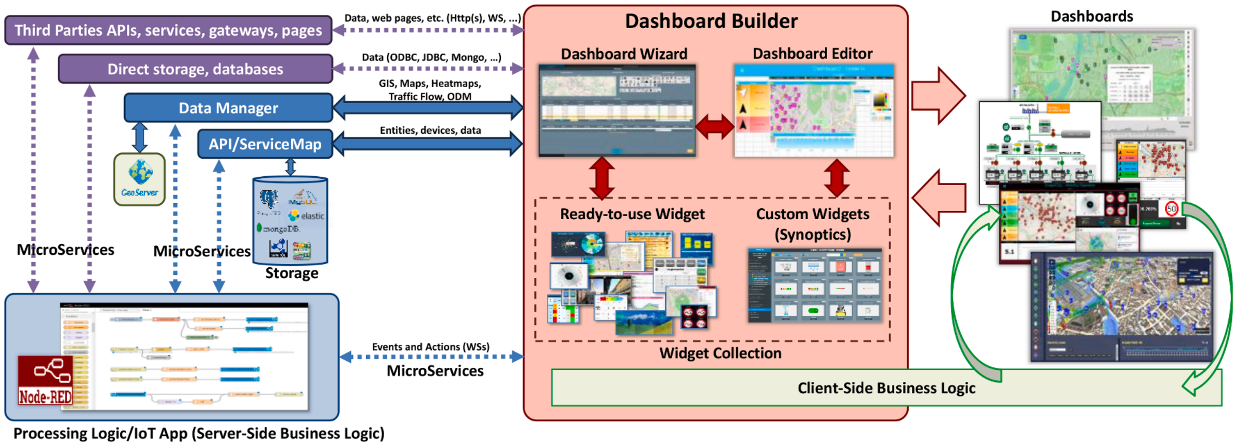

Fig. 1. Block diagram illustrating the Dashboard Builder architecture and its integration with the Snap4City platform (blue blocks and arrows) and with external resources (purple blocks and arrows).

implemented as protocols and formats and allow to exploit storage systems (e.g., knowledge bases, relational DB, ODBC, JDBC, NoSQL api), any heterogeneous data sources, connection protocols such as HTTP/HTTPs, API REST, WebSocket, IoT Brokers API, etc. Therefore, widgets can work/react in an event driven way by Web sockets, and also access the historical data (time series) of sensors, maps, heatmaps, traffic flows, origin-destination matrices (ODMs), as well as query GIS servers (e.g., a GeoServer via WMS, WFS protocols).

Such dashboard editing/creation is simplified by the **Dashboard Wizard**, by means of which users can create/connect dashboards in a few steps, exploiting pre-build templates. Moreover, the related wizard guides users in the selection of the most appropriate widgets for displaying the data of interest, or stating from the preferred widget to identify the data which can be used for populating it, or stating from the map to identify the data which are present in the area and the widgets for their rendering, etc. The Wizard assists users by reducing complexity, providing suggestions on finding combinations between data types (time series, vectors, array, maps, trajectories, heatmaps, origin destination, point of interest, typical trends, histograms, etc.), and graphic representations (trends, multi-trends, pie, donut, maps, chords, hierarchies, solar, dendrograms, single content, Italian flag, traffic flow, 3D building, etc.). Once the editing operation has been completed, users can save the related dashboard (with the possibility to delegate it or grant access to different users) and it is made available in the dashboard collection.

Moreover, with the aim of enabling developers in using the Dashboard Builder to create custom visual analytics, business intelligence, and what-if analysis tools, a flexible approach for modeling any business logic is provided with two different manners: Server-Side Business Logic (SSBL) and Client-Side Business Logic (CSBL) [23,24]. According to the SSBL approach, some graphic Widgets of dashboards have a counter part in the Node-RED nodes [25] and thus are regarded as MicroServices which the Node-RED can send data and controls to, and which the Node-RED can receive events/actions from, as provided by users [26]. This approach allows the dashboard designer to create SSBL by using the visual programming in Node-RED. This approach also implies that once a new widget node is deployed on a Node-RED flow, the related widget is automatically created into the selected dashboard and a WebSocket secure connection is established. The integration of Dashboards with Node-RED is also used to activate Data Analytics (data processing with machine learning and artificial intelligence algorithms) based on user actions on dashboards and/or scheduling in Node-RED [27,28].

The CSBL approach is realized by coding segments of JavaScript directly into the graphic interface configuration of widgets (*green block in* Fig. 1). The CSBL code can call: (i) any external APIs (*purple blocks and arrows in* Fig. 1), (ii) any API and data base services of the Snap4City platform (*blue blocks and arrows* in Fig. 1), and (iii) specific functions to send/receive commands and data to other widgets (*green block in* Fig. 1). This approach allows users who can interact with some widget graphic element (a line, a legend, a bar, a pin on map, etc.) to activate a rendering, a computing, or a visualization on one or more widgets in the dashboard, and even open another dashboard with some parameters. With a minimal JavaScript programming capability to code the logic in these dashboards, a user can add intelligence functionalities to any widget to retrieve data directly from internal and external sources and generate and catch messages from other widgets in an event-driven way.

5. Software functionalities

The Dashboard Builder allows users to create interactive dashboards by composing widgets starting from a white canvas by means of the Dashboard Editor (see Fig. 2). As above described, widgets are specialized in handling different data kinds, without requiring any programming competence. In addition, widgets can exploit business logic as CSBL and SSBL which can be defined on server side with Node-RED and client side in small JavaScript coding. The widget selection and generation within dashboard can be accelerated by using the Dashboard Wizard (see Fig. 3), that matches data to widgets taking into account data types, locations, organization, and any multiclass faceted search. In any case, widgets put on dashboard can be edited for setting their configuration and relationships among them by using graphical panels and contextual menus on graphics elements. Such setting includes map position and zoom, colors, time series (single/multi, stacked, shaded, comparative, double axes, etc.), kinds of bar/pie plots (staked, horiz./vert., etc.), all the selector parameters and data to be recalled on map with icons, color, heatmaps, tracks, ODM, shapes, scenarios, etc. Moreover, any widget can be sized, titled and placed in any position within the dashboard, in a way the user prefers. Examples of the basic available widgets in the Widget Collection are presented in Fig. 4, while a very large set is accessible from the training course in PDF, Interactive and videos [29].

The Dashboard Builder includes the following functionalities:

- create, import, export, delete dashboards and/or widgets;
- dashboards and/or data can be made private or public; when private the access can be controlled;
- manage widgets for data: single contents (text or numbers), time trends (even accepting multiple time series, staked and not stacked, single and double axes, etc.) with drill-down and comparison capabilities, speedometer, gauge, single bars; Kiviat charts, spider net, donuts and pies, bar series and histograms, chords, etc.; tables with dynamic row indexing according to the selected column, search, column selection, server side paging, etc.;

Fig. 2. Dashboard editor of the proposed dashboard builder.

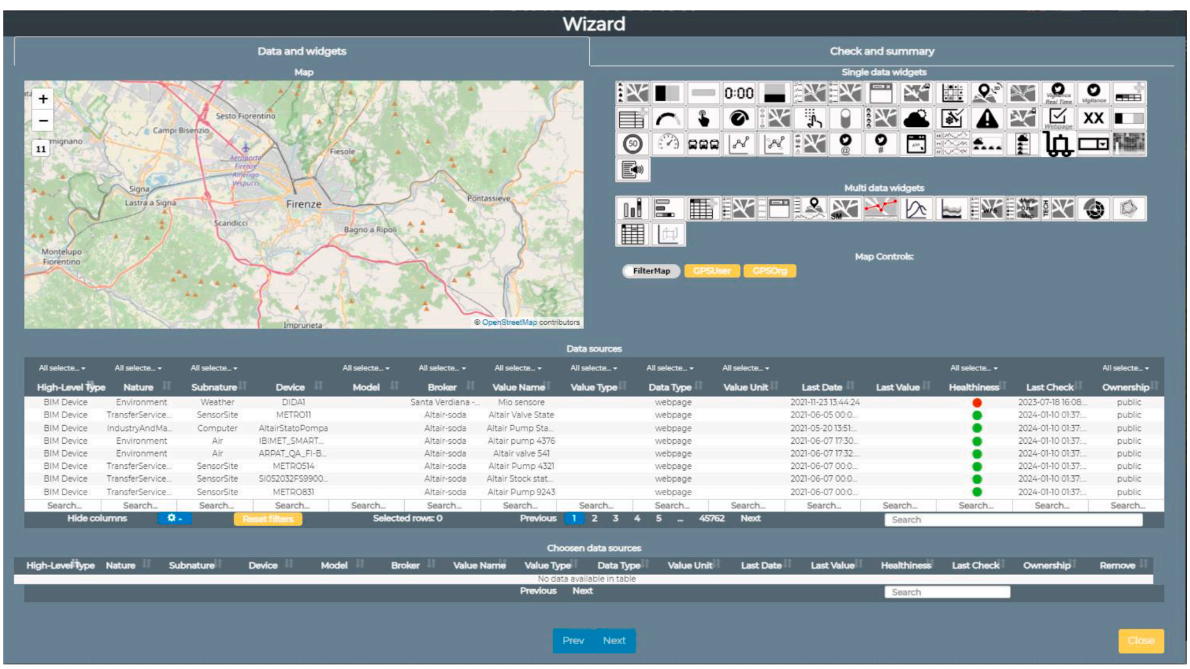

Fig. 3. Dashboard Wizard for data kinds and widget association.

- manage widgets External Services, to embed external web pages and services, BIM viewer, TV cameras (using WebRTC), etc.;
- manage special widgets, such as Weather Forecast, Social Media monitoring, Healthcare status, synoptics as SVG augmented with anchors, to display static and dynamic data [30];
- manage 2D multi-data maps for visualizing geo-referenced data, such as Points of Interests (POI), Smart City sensors, personal devices and personal data tracking, as well as trajectories, paths, heatmaps, etc., and allowing user interactions and any triggering of CSBL/SSBL;
- manage widgets which are nodes in Nore-RED and allow bidirectional web-sockets secure communication for SSBL;
- allow to code CSBL into widgets to enable the creation of visual analytics, business intelligence and what-if analysis tools;
- manage widgets for providing tools of selection, menus, connected to 2D and 3D map widgets, to show and hide information dynamically on user request;
- manage interaction and interactive widgets, such as buttons, knobs, dimers, keypads, etc. which are used to interact with IoT devices and applications, and also any interaction of graphic elements of widget, so as to provoke actions via standard relationships as drill down, zoom/pan on maps, etc., as well as riggers on CSBL and messages to SSBL;
- manage 3D multi-data maps to build smart city digital twin interfaces where 3D reconstructions of city entities are augmented with static and real-time information coming from sensors, analytic services, open data, etc.;

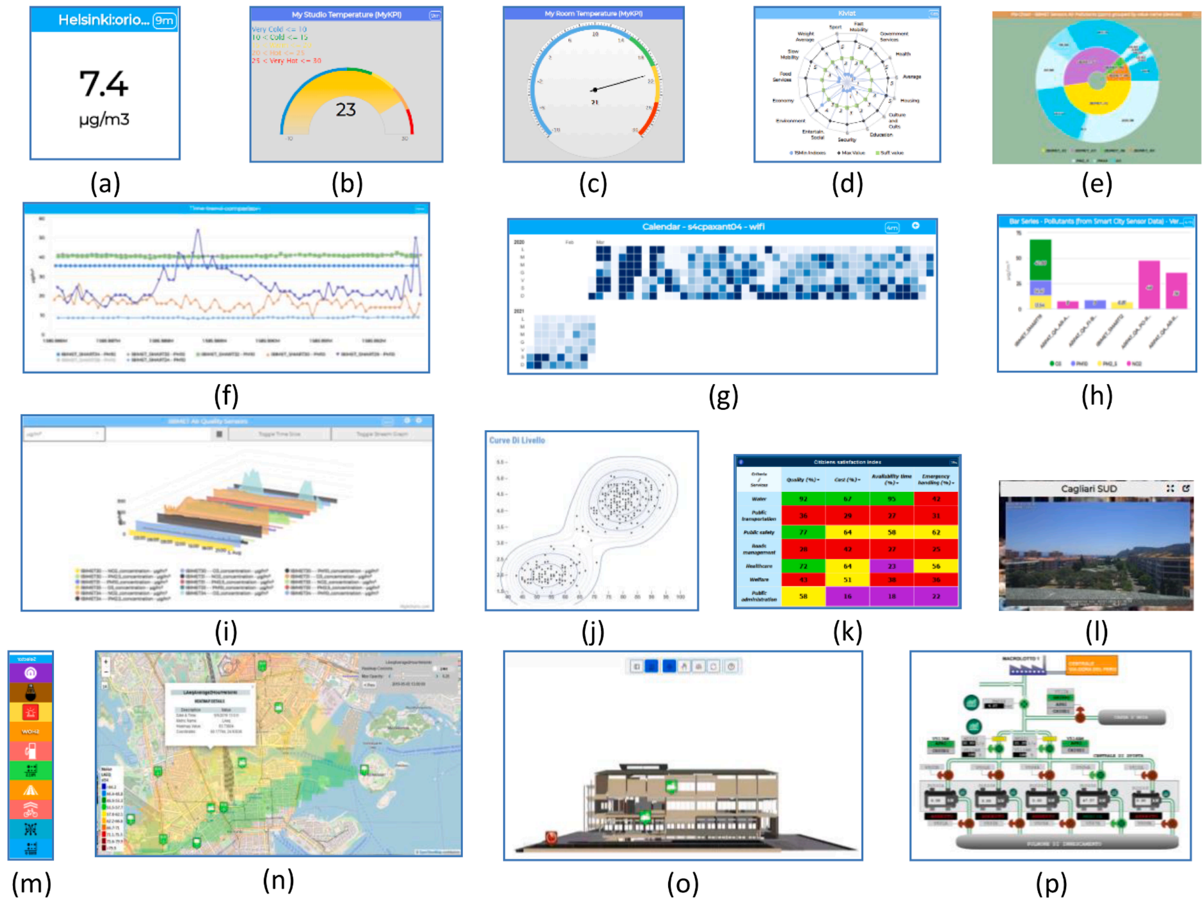

Fig. 4. Some of the main widgets included in the Snap4City Widget Collection: (a) single content text, number, HTML; (b) gauge; (c) speedometer; (d) Kiviat/spider net; (e) donut; (f) multi time series; (g) calendar heatmap; (h) bar chart; (i) data cube; (j) contour/level lines; (k) table; (l) TV-Cam; (m) selector menu (icon version); (n) 2D multi data map; (o) BIM viewer/inspector; (p) custom widget (synoptic).

- extensively support CSBL and SSBL solutions to enable the creation of visual analytics and business intelligence tools and related applications [24];
- provide secure end-to-end connections from users to data, also supporting authentication and authorization mechanisms with standards such as LDAP, OpenID connect, etc., and in the respect of data privacy according to GDPR [31,32].

Each and every widget includes CSBL [24] to enable a dynamic exchange of information among one another, but also with internal and external services. With a minimal coding effort, users can implement abstract functions to let the widget requests data from services and storages and throw messages from/to other widgets in order to update/change visualization. For example, after a drill-down operation on a time-series to select a specific time range, a widget sends a message to one or more widgets to control their rendering, or what they have loaded

Fig. 5. 3D multi-data map widget used to represent the smart city digital twin of Florence, Italy.

from the storage.

6. Illustrative examples

To illustrate the Dashboard Builder, in Fig. 2 an example of the graphical interface used to compose a dashboard is reported. Using buttons in the top bar, users can set the dashboard metadata (e.g., title, subtitle), change theme, add different widgets, import/export, save and request a preview of the dashboard. Widgets appear in the main panel and can be sized and placed using drag-and-drop. Due to its particularity and complexity, in Fig. 5 an example of the 3D multi-data map is presented. The 3D multi-data map can handle different 3D representations for the terrain, the buildings, and other urban elements. It can visualize sensors and services as pins, different maps and static and animated heatmaps by texturing the terrain, interactive elements, so as to show the road graph, cycling paths and public transport routes and stops. Moreover, specific 3D representations are used to show real-time traffic conditions (as animated 3D arrows and crest) and measurements from city sensors (presented as 3D pillars with a height proportional to the observed value). A freely accessible demo showing these capabilities of a 3D multi-data map can be found at https://digitaltwin.snap4city.org/. A video on this tool can be accessed from: https://www.youtube.com/channel/UC3tAO09EbNba8f2-u4vandg

7. Impact

Dashboards are extremely diffused GUIs, from industrial to smart-city contexts, and their development could be difficult and time consuming due to specific user requirements, in particular if interactive functionalities must be granted and direct access to data and business logic processes is requested. The proposed Dashboard Builder offers an easy-to-use solution to those problems, allowing even non expert users to build effective dashboard with minimal efforts and a high level of tailoring possibilities, while providing automatic association among widget and data types. Moreover, the proposed solution is extendable, thanks to its modular architecture, and able to accommodate more complex visualizations and interactions, granting additional customization possibilities to more expert users. For example, in External Content D3 graphic libraries can be used. Therefore, the Dashboard Builder offers multiple level of complexity, meeting the needs of different users.

The use of the Dashboard Builder eases any realization of interfaces when used to display results of research and planning activities. For example, city councils can exploit map widgets to observe traffic density reconstruction [33] and heatmaps of pollutant dispersion [28], as well as to view predictions of landslide events [27], and to visualize 3D digital twins of the whole city [34]. On the other hand, charts, time trends, tables, etc., can be used to visualize results of pedestrian detection in video surveillance [35] for security and commercial scenarios, as well as to show data and measurements of industrial plants [30,36,37]. Several scenarios have been developed with the Dashboard Builder in a large number of domains (mobility and transport, environment, energy, security, tourism, etc.), as reported in https://www.snap4city.org/4.

An evaluation of the impact of the Dashboard Builder in improving and simplifying the process of creating dashboards and visual data analytics has been reported in [20]: during a training course of the Snap4City platform, users with different levels of technical expertise were provided with exercises and questionnaires on the Dashboard Builder, in order to assess if the tool is effective in matching the user's intention and needs with respect to the results obtained by creating the dashboard. Some of the most significant results emerging from this survey are the following: 93 % of participants stated that the Dashboard Builder would be useful for their work; 72.63 % were more than somehow satisfied with the easiness concerning the dashboard production with respect to the other products in their usage; 96.51 % were more than somehow satisfied with the completeness of the dashboard.

The Dashboard Builder, integrated into the Snap4City platform, has been deployed in several industrial and smart-city contexts by DISIT lab and several private companies and public administrations for their installations as listed in https://www.snap4city.org/661. Many other private installations are not listed. The largest installation includes 19 organizations and more than 8000 operators and developers. More than 1700 dashboards have been created (among which, more than 500 are connected with SSBL based on Node-RED), including more than 11,700 active widgets managing an average of about 2.2 million of complex data messages per day, from more than 260 thousand distinct data sources. Moreover, the Dashboard Builder was used to realize the dashboards in use to monitor the ISPRA Joint Research centre (JRC) of the European Commission. A training course is accessible at https://www.snap4city.org/944.

The Dashboard Builder is also one of the main tools used and installed by Snap4 [38], a spin-off of the University of Florence that develops data ingestion and monitoring solutions exploiting the Dashboard Builder to customize dashboards for several industrial realities.

A list of publications regarding these activities is reported in https://www.snap4city.org/426.

8. Conclusions

In this paper the Dashboard Builder (freely available from our GitHub https://github.com/disit/dashboard-builder and as Docker container https://hub.docker.com/r/disitlab/dashboard-builder/) was presented. The Dashboard Builder is a modular software used to create interactive dashboards with minimal efforts carried out even by non-expert users, thanks to its graphical interface. By exploiting different widgets available in the Dashboard Builder collection, and with the help of the Dashboard Wizard, a dashboard can be composed to visualize data in charts, time trends, tables, maps, and even full urban digital twin. Additionally, widgets can interact with multiple storage systems and with analytic processes and allow the possibility to specify client-side and server-side business logic functionalities to exchange information and update their visualizations in an event driven manner, thus developing visual analytics, business intelligence and what-if analysis tools. The Dashboard Builder, integrated into the Snap4City platform, is widely used both in academic and industrial fields, since it provides an easy and quick solution for presenting research results and visualize historic and real-time data.

CRediT authorship contribution statement

Pierfrancesco Bellini: Validation, Software, Formal analysis, Conceptualization, Investigation, Methodology, Writing – original draft, Writing – review & editing. **Marco Fanfani:** Writing – original draft, Visualization, Conceptualization, Investigation, Software. **Paolo Nesi:** Writing – review & editing, Writing – original draft, Validation, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Gianni Pantaleo:** Writing – review & editing, Validation, Methodology, Investigation, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

A special thanks to the many users and developers working on the Snap4City platforms. Snap4City (https://www.snap4city.org), and Km4City are open technologies of DISIT Lab https://www.disit.org . The Dashboard Builder has been partially funded by REPLICATE H2020 RnD project of the European Commission, and by Sii-Mobility MIUR, and by Snap4City project and action, and by the Herit-Data Interreg project.

References

[1] Apache Software Foundation. APACHE SOLR. 2024. (Version 9.5.0) [Computer software], SOLR website: https://solr.apache.org/ (Accessed on January 23, 2024).

[2] Elasticsearch B.V. Kibana. 2024. (Version 8.13) [Computer software] ElasticSearch Kibana website: https://www.elastic.co/kibana (Accessed on January 23, 2024).

[3] FIWARE. SpagoBI. 2024. (Version 5.2.1) [Computer software] SpagoBI website: https://spagobi.readthedocs.io/en/latest/ (Accessed on January 23, 2024).

[4] Opendatasoft. OpenDataSoft. 2024. [Computer software] OpenDataSoft website: https://www.opendatasoft.com/en/ (Accessed on January 23, 2024).

[5] Vázquez-Ingelmo A, Garcia-Peñalvo FJ, Therón R. Information dashboards and tailoring capabilities - A systematic literature review. IEEE Access 2019;7: 109673–88. https://doi.org/10.1109/ACCESS.2019.2933472.

[6] Greater London Authority. London Data Store. 2024 [Computer software] London data store website: https://data.london.gov.uk/ (Accessed on January 23, 2024).

[7] Craglia M, Scholten HJ, Micheli M, Hradec J, Calzada I, Luitjens S, Ponti M, Boter J. Digitranscope: the governance of digitally-transformed society, eur 30590 en. Luxembourg: Publications Office of the European Union; 2021, JRC123362. https://doi.org/10.2760/503546. ISBN 978-92-76-30229-2.

[8] McArdle G, Kitchin R. The Dublin Dashboard: design and development of a real-time analytical urban dashboard, ISPRS Annals of Photogrammetry. Remote Sens Spat Inf Sci 2016;IV-4/W1:19–25.

[9] F. Causone, E. Morello, A. Capozzoli, T. CerquitelliT. Bianchi, L. Cappelli, P. Colusso, A. Guzzetti, V. Volpe, Smart City DiVA Project. 2018. Smart City DiVA project website: https://www.asp-poli.it/smart-city-diva-smart-city-data-visualization-and-analytics/ (Accessed on January 23, (2024).

[10] Farmanbar M.; Rong C. Triangulum City Dashboard: an Interactive Data Analytic Platform for Visualizing Smart City Performance. Processes 2020, 8, 250. 10.3390/pr8020250.

[11] Rojas E, Bastidas V, Cabrera C. Cities-board: a framework to automate the development of smart cities dashboards. IEEE Internet Things J 2020;7(10): 10128–36. https://doi.org/10.1109/JIOT.2020.3002581. Oct.

[12] Santos H, Dantas V, Furtado V, Pinheiro P, McGuinness DL, Blomqvist E, Maynard D, Gangemi A, Hoekstra R, Hitzler P, Hartig O. From data to city indicators: a knowledge graph for supporting automatic generation of dashboards. The semantic web. ESWC 2017. Lecture notes in computer science, 10250. Cham: Springer; 2017. https://doi.org/10.1007/978-3-319-58451-5_7.

[13] Vázquez-Ingelmo A, García-Peñalvo FJ, Therón R. MetaViz – A graphical meta-model instantiator for generating information dashboards and visualizations". J King Saud Univ Comput Inf Sci 2022;34(10):9977–90. https://doi.org/10.1016/j.jksuci.2022.09.015. Part BPages.

[14] Radovanović S, Majstorović B, Kukolj S, Bjelica MZ. Device cloud platform with customizable remote user interfaces. In: Proceedings of the 2014 IEEE fourth international conference on consumer electronics berlin (ICCE-Berlin); 2014. p. 202–4. https://doi.org/10.1109/ICCE-Berlin.2014.7034231.

[15] S. Van Hoecke, C. Huys, O. Janssens, R. Verborgh, R. Van de Walle, "Dynamic monitoring dashboards through composition of web and visualization services", in: Mandler B., et al. Internet of things. iot infrastructures. iot360 2015. lecture notes of the institute for computer sciences, social informatics and telecommunications engineering, vol 170. Springer, Cham. (2016). 10.1007/978-3-319-47075-7_50.

[16] Logre I, Mosser S, Collet P, Riveill M. Sensor data visualisation: a composition-based approach to support domain variability. In: Proceedings of the european conference on modelling foundations and applications (ECMFA; 2014.).

[17] Wisetown. Urban Dashboard. 2024. [Computer software] Wisetown Urban Dashboard website: https://wise.town/en/cities/city-dashboard/ (Accessed on January 23, 2024).

[18] EDAG PRODUCTION SOLUTIONS. Smart City Dashboard. [Computer software] Edag Smart City Dashboard website: https://smartcity.edag.com/en/referenzen/smart-city-dashboard/ (Accessed on January 23, 2024).

[19] Adreani L, Bellini P, Fanfani M, Nesi P, Pantaleo G. Design and develop of a smart city digital twin with 3d representation and user interface for what-if analysis. In: Proceedings of the international conference on computational science and its applications; 2023. p. 531–48. Cham: Springer Nature Switzerland.

[20] Han Q, Nesi P, Pantaleo G, Paoli I. Smart city dashboards: design, development and evaluation. In: Proceedings of the IEEE ICHMS 2020, international conference on human machine systems, September; 2020. http://ichms.dimes.unical.it/.

[21] DISIT Lab. Snap4City. 2024. [Computer software] Snap4City: https://www.snap4city.org (Accessed on January 23, 2024).

[22] Bellini P, Cenni D, Marazzini M, Mitolo N, Nesi P, Paolucci M. Smart city control room dashboards: big data infrastructure, from data to decision support. J. Vis. Lang. Comput 2018;4:75–82.

[23] Bellini P, Bologna D, Fanfani M, Ipsaro Palesi LA, Nesi P, Pantaleo G. Rapid prototyping & development life cycle for smart applications of internet of entities. In: Proceedings of the 2023 27th international conference on engineering of complex computer systems (ICECCS); 2023. p. 142–51. https://doi.org/10.1109/ICECCS59891.2023.00026.

[24] P. Nesi, "Client-Side Business Logic Widget Manual", Personal communication. CSBL: client-side business logic widget manual: https://www.snap4city.org/download/video/ClientSideBusinessLogic-WidgetManual.pdf, (Accessed on January 23, 2024).

[25] OpenJS Foundation. Node-RED. 2024. (Version 3.1.8) [Computer software] Node-RED website: https://nodered.org/ (Accessed on January 23, 2024).

[26] Badii C, Bellini P, Difino A, Nesi P, Pantaleo G, Paolucci M. MicroServices suite for smart city applications. Sensors 2019. https://doi.org/10.3390/s19214798. MDPI.

[27] Collini E, Palesi LAI, Nesi P, Pantaleo G, Nocentini N, Rosi A. Predicting and understanding landslide events with explainable AI. IEEE Access 2022;10: 31175–89. https://doi.org/10.1109/ACCESS.2022.3158328.

[28] Bilotta S, Nesi P. Estimating CO2 emissions from IoT traffic flow sensors and reconstruction. Sensors 2022;22(9):3382. https://doi.org/10.3390/s22093382.

[29] P. Nesi, "Training Course of Snap4City. Part 2", Personal communication. Training course of snap4city: https://www.snap4city.org/944, see for dashboard Part 2. (Accessed on March 05, 2024).

[30] Badii C, Bellini P, Cenni D, Mitolo N, Nesi P, Pantaleo G, Soderi M. Industry 4.0 synoptics controlled by IoT applications in node-RED. In: Proceedings of the IEEE Internet of Things; 2020. , November 02-06.

[31] European Parliament and Council of the European Union. "European General Data Protection Regulation (GDPR)." 2016. European general data protection regulation (GDPR) website: https://gdpr.eu/ (Accessed on January 23, 2024).

[32] Badii C, Bellini P, Difino A, Nesi P. Smart city IoT platform respecting GDPR privacy and security aspects. IEEE Access 2020. https://doi.org/10.1109/ACCESS.2020.2968741.

[33] Bilotta S, Nesi P. Traffic flow reconstruction by solving indeterminacy on traffic distribution at junctions. Future Gener Comput Syst 2021;114:649–60.

[34] Adreani L, Bellini P, Colombo C, Fanfani M, Nesi P, Pantaleo G, Pisanu R. Implementing integrated digital twin modelling and representation into the Snap4City platform for smart city solutions. Multimed Tools Appl 2023:1–26.

[35] Collini E, Ipsaro Palesi LA, Nesi P, Pantaleo G, Zhao W. Flexible thermal camera solution for Smart city people detection and counting. Multimed Tools Appl 2023: 1–29.

[36] Bellini P, Cenni D, Mitolo N, Nesi P, Pantaleo G, Soderi M. High level control of chemical plant by industry 4.0 solutions. J Ind Inf Integr 2022;26:100276.

[37] Bellini P, Ipsaro Palesi LA, Giovannoni A, Nesi P. Managing complexity of data models and performance in broker-based internet/web of things architectures. Elsevier; 2023. Journal, Internet of Things.

[38] SNAP4 s.r.l. Website. 2024. SNAP4 s.r.l. website: https://www.snap4.eu/ (Accessed on January 23, 2024).