Article:
Elshehaly, M, Randell, R, Brehmer, M et al. (4 more authors) (2021) QualDash: Adaptable Generation of Visualisation Dashboards for Healthcare Quality Improvement. IEEE Transactions on Visualization and Computer Graphics, 27 (2). pp. 689-699. ISSN 1077-2626

https://doi.org/10.1109/TVCG.2020.3030424

QualDash: Adaptable Generation of Visualisation Dashboards for Healthcare Quality Improvement

Mai Elshehaly, Rebecca Randell, Matthew Brehmer, Lynn McVey, Natasha Alvarado, Chris P. Gale, and Roy A. Ruddle

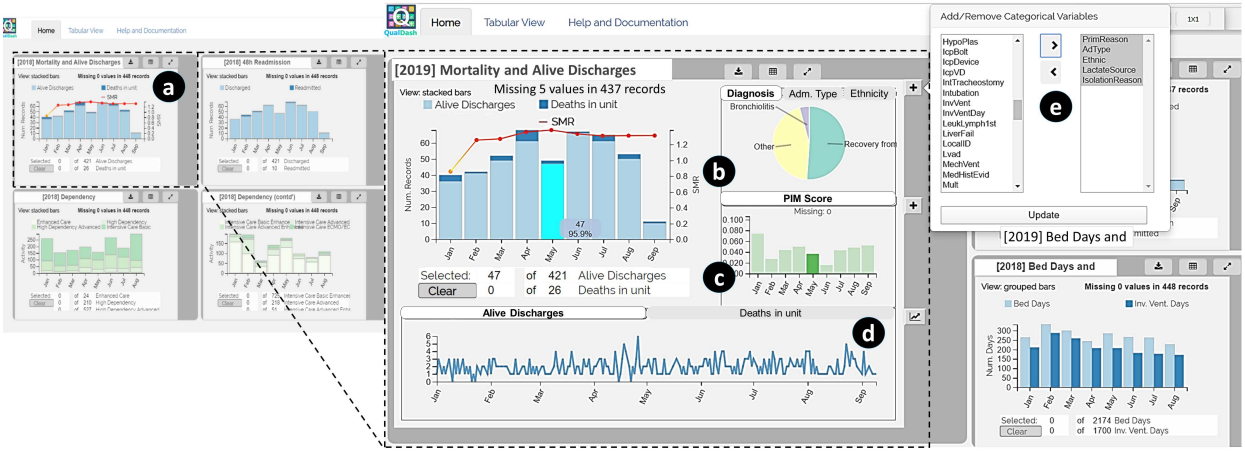


Fig. 1. A dashboard with four dynamically generated QualCards (left) including one for the *Mortality* metric (a); and an expansion of the *Mortality* QualCard with categorical (b), quantitative (c) and temporal (d) subsidiary views, which are customisable via a popover (e).

Abstract— Adapting dashboard design to different contexts of use is an open question in visualisation research. Dashboard designers often seek to strike a balance between dashboard adaptability and ease-of-use, and in hospitals challenges arise from the vast diversity of key metrics, data models and users involved at different organizational levels. In this design study, we present QualDash, a dashboard generation engine that allows for the dynamic configuration and deployment of visualisation dashboards for healthcare quality improvement (QI). We present a rigorous task analysis based on interviews with healthcare professionals, a co-design workshop and a series of one-on-one meetings with front line analysts. From these activities we define a metric card metaphor as a unit of visual analysis in healthcare QI, using this concept as a building block for generating highly adaptable dashboards, and leading to the design of a Metric Specification Structure (MSS). Each MSS is a JSON structure which enables dashboard authors to concisely configure unit-specific variants of a metric card, while offloading common patterns that are shared across cards to be preset by the engine. We reflect on deploying and iterating the design of QualDash in cardiology wards and pediatric intensive care units of five NHS hospitals. Finally, we report evaluation results that demonstrate the adaptability, ease-of-use and usefulness of QualDash in a real-world scenario.

Index Terms—Information visualisation, task analysis, co-design, dashboards, design study, healthcare.

1 Introduction

Visualisation dashboards are widely adopted by organizations and individuals to support data-driven situational awareness and decision making. Despite their ubiquity, dashboards present several challenges to visualisation design as they aim to fulfill the data understanding needs of a diverse user population with varying degrees of visualisation literacy. Co-designing dashboards for quality improvement (QI) in healthcare presents an additional set of challenges due to the vast diversity of: (a) performance metrics used for QI in specialised units, (b) data models underlying different auditing procedures, and (c) user classes involved. For example, while cardiologists in a large teaching hospital may monitor in-hospital delays to reperfusion treatment, some district general hospitals rarely offer this service, so they prioritise and monitor a different set of metrics. This within-specialty heterogeneity of tasks is further amplified when talking to healthcare professionals from different specialties, who use entirely different audit databases to record and monitor performance. Consequently, the problem of designing a dashboard that can adapt to this diversity requires a high level of human involvement and the appropriate level of automation for dashboard adaptation remains to be an open research question [44].

We present QualDash: a dashboard generation engine which aims to simplify the dashboard adaptation process through the use of customizable templates. Driven by the space of analytical tasks in healthcare QI, we define a template in the form of a Metric Specification Structure (MSS), a JavaScript Object Notation (JSON) structure that concisely describes visualisation views catering to task sequences pertaining to individual metrics. The QualDash engine accepts as input an array of MSSs and generates the corresponding number of visualisation containers on the dashboard. We use a card metaphor to display these containers in a rearrangeable and adaptable manner, and call each such container a "QualCard". Figure 1 (left) shows an example dashboard with four generated QualCards.

We followed Sedlmair et al.'s design study methodology [49] to de-

- *Mai Elshehaly and Rebecca Randell are with the University of Bradford, UK. E-mail: M.Elshehaly,R.Randell@bradford.ac.uk.*
- *Matthew Brehmer is with Tableau, Seattle, Washington, United States. E-mail: mbrehmer@tableau.com.*
- *Lynn McVey, Natasha Alvarado, Chris P. Gale and Roy A. Ruddle are with the University of Leeds, UK. E-mail: L.McVey,N.Alvrarado,C.P.Gale,R.A.Ruddle@leeds.ac.uk*

Manuscript received 30 Apr. 2020; accepted 14 Aug. 2020. Date of Publication xx Feb. 2021; date of current version xx Xxx. 20xx. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/xxxxxxx

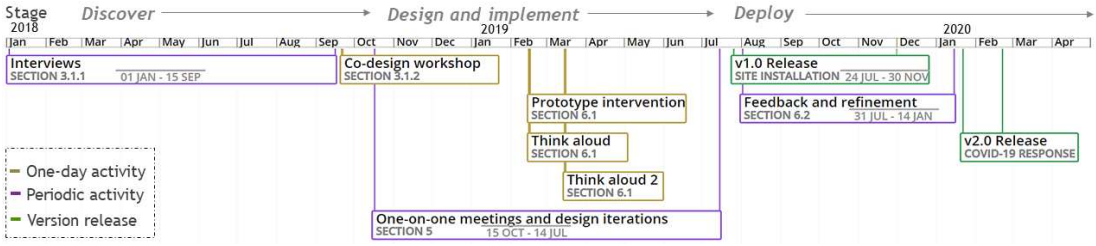


Fig. 2. Timeline of the QualDash project. Design and implementation activities which spanned a period of time are given a date range and a grey bar.

sign, implement and deploy QualDash in five NHS hospitals (Figure 2). In the *Discover* stage, we conducted 54 interviews with healthcare professionals and a co-design workshop, and the data we collected allowed us to characterise task sequences in healthcare QI. In the *Design* and *Implement* stages, we conducted nine one-on-one meetings with front line analysts, as we iterated to design an initial set of key-value pairs for MSS configuration. Next, we conducted a second workshop, where we elicited stakeholders' intervention with a paper prototype and a high-fidelity software prototype. The paper prototype was designed to focus stakeholders' attention on the match between tasks and QualCards. The software prototype was accompanied with a think-aloud protocol to evaluate the usability of the dashboards. These activities resulted in an additional set of metric features, which we added in another design iteration. In the *Deploy* stage, we deployed QualDash in five hospitals, conducted a series of ten meetings with clinical and IT staff to further adapt the MSSs to newly arising tasks, and we collected evidence of the tool's usefulness and adaptability. Suggestions for refinement were also gathered and addressed in a second version of QualDash.

Our contributions in this paper are: (a) a thorough task characterisation which led to the identification of a common structure for sequences of user tasks in healthcare QI (Section 3); (b) a mapping of the identified task structure to a metric card metaphor (a.k.a. the QualCard) and a Metric Specification Structure (MSS) that allows for concise configuration of dashboards; (c) a dashboard generation engine that accepts an array of MSSs and generates the corresponding QualCards with GUI elements that support further customisation (Section 5); (d) Our reflection on 62 hours of observing the deployment and adaptation of QualDash in the five NHS hospitals (Section 6).

2 Background and Related Work

Originally derived from the concept of balanced scorecards [20], quality dashboards inherit factors that are crucial to successful adoption, including scalability, flexibility of customisation, communication, data presentation, and a structural understanding of department-level and organization-level performance objectives and measures [4]. This paper aims to connect the dots between these factors through a continuous workflow that maps user **tasks** to **dashboard** specification. This section defines these relevant terms and outlines related work.

Tasks are defined as domain- and interface-agnostic operations performed by users [34]. A *task space* is a design space [19] that aims to consolidate taxonomies and typologies as a means to reason about all the possible ways that tasks may manifest [47]. This concept helps visualisation researchers to *"reason about similarities and differences between tasks"* [35]. Several task taxonomies, typologies, and spaces aim to map domain-specific tasks into a set of abstract ones that can guide visualisation design and evaluation [1–3, 7, 21, 30, 47]. These classifications have proven beneficial in several steps of the generative phases of visualisation design [1, 2, 11, 21, 48, 52]. Amar and Stasko [2] and Sedig and Parsons [48] promoted the benefits of typologies as a systematic basis for thinking about design. Heer and Shneiderman [17] used them as constructs when considering alternative view specifications; and Kerracher and Kennedy [21] promoted their usefulness as "checklists" of items to consider. By using a task space in the generative phase of design, Ahn et al. [1] identified previously unconsidered tasks in network evolution analysis. We leverage the opportunities that task classification offers to support our understanding of common structures for task sequences in the context of dashboard design in healthcare QI.

Dashboards are broadly defined as *"a visual display of data used to monitor conditions and/or facilitate understanding"* [53]. Sarikaya et al. highlighted the variety of definitions that exist for dashboards [44], while Few highlighted the disparity of information that dashboard users typically need to monitor [12]. We focus our attention on the definition and use of visualisation dashboards in a healthcare context, where Dowding et al. distinguished two main categories of dashboards that inform performance [10]: **clinical dashboards** provide clinicians with timely and relevant feedback on their patients' outcomes, while **quality dashboards** are meant to inform *"on standardized performance metrics at a unit or organizational level"* [10]. Unlike clinical dashboards which cater to a specialized user group within a specific clinical context (e.g. [26, 41, 58]), we further set the focus on quality dashboards, which exhibit a wider variety of users, contexts, and tasks.

The healthcare visualisation literature presents tools that are broadly classified [40] into ones that focus on individual patient records (e.g., LifeLines [31]), and ones which display aggregations (e.g. Lifelines2 [51], LifeFlow [55], EventFlow [32], DecisionFlow [16] and TimeSpan [26]). A common theme is that these tools dedicate fixed screen real-estate to facets of data. Consequently, they support specialised tasks that focus on specific types of events (e.g. ICU admissions [51]), or specific patient cohorts (e.g. stroke patients [26]).

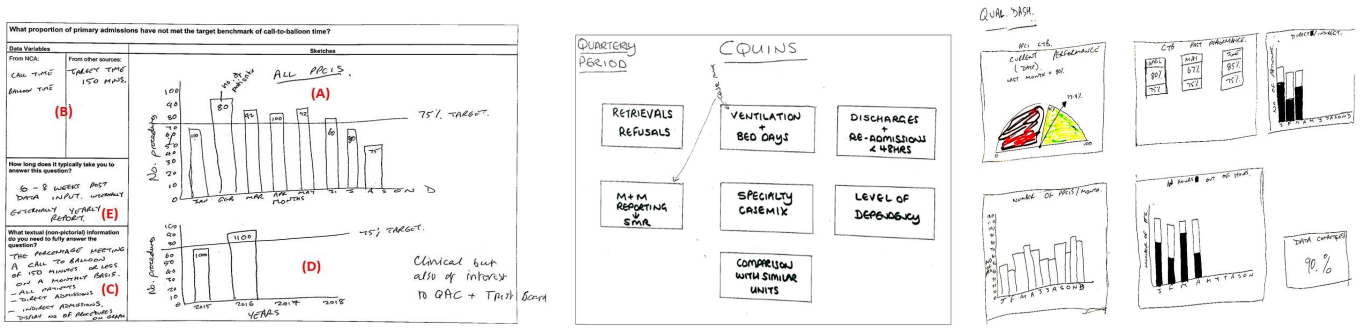
Commercial software such as Tableau offer a wealth of expressivity for dashboard generation, allowing interactive dashboards to be deployed to members of an organisation *"without limiting them to pre-defined questions"* [43]. Similar levels of expressivity are also attainable with grammars such as Vega [46] and Vega-lite [45]. These grammars empower users' to explore many visualisation alternatives, particularly when encapsulated in interactive tools [56, 57]. More recently, Draco [33] subsets Vega-lite's design space to achieve an even more concise specification. However, this specification and its precedents do not offer a mechanism to encode users' *task sequences*. In contrast, we contribute an engine that leverages taxonomic similarities of identified tasks to present "templates" for dashboard view specifications. While our MSS spans a subset of the design space of the aforementioned tools, we show that our concise templates enable the co-design and deployment of dashboards in healthcare QI. To our knowledge, this paper presents the first attempt to capture task sequences for audiences in healthcare QI and to match this characterisation to dynamic dashboard generation.

3 Task Analysis

Our analysis is guided by the dimensions of the dashboard design space identified by Sarikaya et al. [44] and the challenges specific to healthcare QI [39]. Namely, we sought to answer the questions: What user task sequences exist within and across audiences of healthcare QI? How are metrics and benchmarks defined? What visual features strike a balance between ease-of-use and adaptation? And how updatable should the dashboards be?

3.1 Data Collection

Our data collection started with an investigation of challenges and opportunities inherent in the use of National Clinical Audit (NCA) data for healthcare QI. NCAs provide national records of data about patient treatments and outcomes in different clinical areas. Participating



(a) (b) (c)

Fig. 3. Examples of participants' responses to Co-design Activity 1 (Story Generation) (a), and Co-design Activity 2 (Task Sequencing) (b and c).

hospitals regularly contribute to updates of over 50 audits, to systematically record and measure the quality delivered by clinical teams and healthcare organizations. We focus on two audits: the Paediatric Intensive Care Audit Network (PICANet) audit [37] and the Myocardial Ischaemia National Audit Project (MINAP) audit [54]. Both audits provided ample data to work with as well as access to a diversity of providers and stakeholders. We characterised the space of QI task sequences through interviews and a co-design workshop.

3.1.1 Interviews with Stakeholders

We interviewed 54 healthcare professionals of various backgrounds, including physicians, nurses, support staff, board members, quality & safety staff, and information staff. The interview questions elicited the participants' own experiences with NCA data, with an emphasis on the role of the data in informing quality improvement. Interviewees gave examples of when audit data were of particular use to them, where limitations and constraints arose, and what they aimed to learn from the data in the future. All interviews were audio-recorded and transcribed verbatim. We used these qualitative data to build a thematic understanding of audit-specific quality metrics. For each metric, we collated queries and quotes from the interview transcripts that led to the generation of an initial pool of 124 user tasks (Appendix A).

The next step was to establish context around these tasks, and to identify connecting sequences among them. Since including all 124 tasks for this contextualisation was not feasible, we selected a representative subset of tasks by considering structural characteristics, based on the three taxonomic dimensions highlighted for healthcare QI: granularity, type cardinality and target [11]. When selecting a reduced task subset, we included tasks that covered different granularity (e.g., unit or national level of detail) and type-cardinality levels (i.e. number of quantitative, temporal and categorical variables) from the task pool.

3.1.2 Co-design Workshop

We organised a half-day co-design workshop to build task sequences and contexts. Workshop participants were seven staff members from one hospital having clinical and non-clinical backgrounds (2 consultants, 2 nurses and 3 hospital information managers). We divided participants into two groups, depending on which audit they were most familiar with and assigned each group a task set that corresponds to one of the audits. Two project team members facilitated the discussion with each group over the course of two co-design activities.

Co-design Activity 1: Story Generation. Inspired by Brehmer and Munzner's approach to task analysis [7], we asked participants to answer questions about: *why* tasks are important, *how* tasks are currently performed, and *what* information needs to be available to perform a task (*input*) as well as *what* information might arise after completing the task (*output)*. We added a fourth question: *who* can benefit most from a task, which is inspired by agile user story-generation [9].

To facilitate the discussion around these four questions, co-design participants were presented with a set of "task cards" (Figure 3a). Each card focused on a single task and was subdivided into four main sections used to collect contextual information about it. The header of a card contained the task's body and an empty box for participants to assign a relevance score. Three parts of the card were dedicated to elicit information about how this task is performed in current practice: the data elements used, the time it takes, and any textual information that might be involved (sketch areas (B), (E) and (C), respectively, in Figure 3*a*). The majority of space on the card was dedicated to a large "sketches" section. This section provided a space for participants to sketch out the processes involved in performing the task, as well as any visualisations used in the same context.

Participants were presented with a set of task cards corresponding to the reduced task space described in Section 3.1.1. We also gave each participant a set of blank cards, containing no task in the task body section. Participants were given the freedom to select task cards that they deemed relevant to their current practice, or to write their own. For each task, participants were asked to solve the *what* and *how* questions individually, while the *why* and *who* questions were reserved for a later group discussion. During the discussion, we asked participants to justify the relevance scores they assigned to each task (*why*), elaborate on their answers, and then sort the cards depending on *who* they believed this task was most relevant to.

Co-design Activity 2: Task Sequencing. From the tasks that were prioritised in Activity 1, we asked participants to select *entry-point* tasks. These are questions that need to be answered at a glance without interacting with the dashboard interface. Once completed, we explained, these tasks may lead to a sequence of follow-up tasks. To identify these sequences, we returned the prioritised task cards from Activity 1 to participants. We asked participants to: *(i)* Select the most pressing questions to be answered at a glance in a dashboard; *(ii)* Sketch the layout of a static dashboard that could provide the minimally sufficient information for answering these tasks; and *(iii)* Select or add follow-up tasks that arise from these entry-point tasks.

3.1.3 Structure of Task Sequences

Our activities revealed that the use of NCA data is largely at the clinical team level, with more limited use at divisional and corporate levels. We identified entry-point tasks that required monitoring five to six key metrics for each audit (see Figure 3b). We have included a glossary of terms in supplementary material to explain each of these metrics.

Our analysis led to three key findings: **[F1]** individual metrics have independent task sequences; **[F2]** each metric has entry-point tasks that involve monitoring a small number of measures over time; and **[F3]** investigation of further detail involves one or more of three abstract *subsidiary tasks*:

- ST1: Break down the main measure(s) for patient **`sub-categories`**
- ST2: Link with other metric-related **`measures`**
- ST3: Expand in **`time`** to include different temporal granularities

[F1] was noted by participants during Activity 1 and maintained through Activity 2. Figure 3 shows example responses for different audits. In Figure 3*b*, a participant explained that a dashboard should provide a minimalist entry point into the metrics of interest to their

Table 1. Task sequence pertaining to the *call-to-Balloon* metric. Code prefixes MEP and MSUB indicate whether a task is an entry-point or subsidiary.

Code	Task	Quan.	Nom.	Ord.	Temp.
$MEP1-1$	**What proportion of primary admissions have / have not met the target benchmark of call-to-balloon time?**	2			1
$MEP1-2$	• On a given month, what was the total number of PCI patients?	1			1
$MEP1-3$	• On a given month, did the percentage of primary admissions that met the target of 150 minutes call-to-Balloon time exceed 70%?	1			1
MSUB1-1	• Of STEMI patients that did not meet the call-to-Balloon target, which ones were direct/indirect admissions?		+ 1		
MSUB1-2	• Where did the patients that did not meet the target come from?		+ 1		
MSUB1-3	• Are delays justified?		+ 1		
MSUB1-4	• How does a month with high number of PCI patients compare to the same month last year?				+ 1
MSUB1-4	• Did the patients who did not meet the target commute from a far away district?			+ 1	
MSUB1-5	• For a month with a high number of delays, what was the average door-to-balloon time?	+ 1			
MSUB1-6	• What is the life and death status of delayed patients 30 days after leaving the hospital?	Excluded			
MSUB1-7	• Compare the average of cases meeting the call-to-balloon target for own site versus district	Excluded			

Pediatric Intensive Care Unit (PICU). Another participant advocated this design by saying: *"I want something simple that tells me where something is worsening in a metric, then I can click and find out more"*.

In Figures 3*a* and *c*, participants faceted views for sequences pertaining to the call-to-balloon metric, for example. They explained that for this metric, patients diagnosed with ST Elevation Myocardial Infarction (STEMI) - a serious type of heart attack - must have a PPCI (i.e. a balloon stent) within the national target time of 150 minutes from the time of calling for help. An entry-point task for this metric regards monthly aggregates of admitted STEMI patients, and the ratio who met this target (Figure 3*a* sketch area (A), and Figure 3*c* bottom left). Participants then linked this to a breakdown of known causes of delay to decide whether they were justified (ST1). One source of delay, for instance, may be if the patient was self-admitted. This information was added by a participant as textual information in Figure 3*a* (sketch area (C)) and by another participant as a bar chart (top right corner of Figure 3*c*). Participants also noted that it is important to investigate the measures in a historic context (ST3) by including previous months (Figure 3*c* top middle) and years (Figure 3*a* sketch area (D)).

Table 1 lists the tasks of the call-to-balloon metric along with counts of different types of data in each task, as defined in [45]. *Quan*tiative, *Nom*inal, *Ord*inal and *Temp*oral measures required for the entry-point tasks are listed and additional measures considered for subsidiary tasks are marked with a + sign. Despite the variability of metrics across audits, the structure of entry point and subsidiary tasks remains the same. Appendix B lists the task sequences we identified for all metrics.

4 Design Requirements for QualDash

Equipped with a well-defined structure of task sequences, we looked into the use of visualisation grammars like Vega-lite [45] to generate views on a dashboard arranged to serve the identified tasks. Findings from the interviews and co-design workshop were further discussed in a sequence of nine one-on-one meetings with front-line analysts over the course of nine months. Front line analysts are audit coordinators and clinicians who are well-acquainted with audit data as they use it for reporting, presentation and clinical governance. Our meetings involved two consultant cardiologists, a consultant pediatrician, and two audit coordinators. Two of the consultants also held the title "audit lead".

During these meetings, we presented the concept of a QualCard as a self-enclosed area of the dashboard screen that captures information pertaining to a specific metric. We demonstrated design iterations of QualCard prototypes and discussed key properties that would be minimally sufficient to configure them. We leveraged the analysts' familiarity with the data by discussing queries as we sought to specify the `field`, `type` and `aggregate` properties of a Vega-lite data axis. This exercise helped us exclude tasks that required data that the audit did not provide (e.g., Task MSUB1-6 in Table 1). We also excluded tasks that required data not accessible within individual sites (e.g., Task MSUB1-7 in Table 1, because comparing against other sites was deemed infeasible as it required cross-site data sharing agreements).

Next, we explored different ways to compose layered and multi-view plots within each QualCard to address the identified task sequences. A number of design requirements emerged from these meetings:

- R1 **Support pre-configured reusable queries for dynamic QualCard generation.** Given the variability of metrics across sites and specialties, each unit requires a dynamically-generated dashboard that captures unit-specific metrics. Pre-configuration is necessary at this level of dashboard authoring, to define care pathways that lead a patient's record to be included in a metric.
- R2 **Each QualCard must have two states:**
  - R2.1 **An entry-point state** in which a QualCard only displays the metric's main measures aggregated over time.
  - R2.2 **An expanded state** in which a QualCard reveals additional views, catering to subsidiary tasks ST1, ST2 and ST3.
- R3 **Support GUI-based adaptability of subsidiary view measures** to cater to different lines of inquiry.
- R4 **Data timeliness**: support varying workflows and frequencies in data updates.
- R5 **Data quality**: present information on missing and invalid data pertaining to a metric's specific measures.
- R6 **Support exports** of visualisations and individual data records to be used in clinical governance meetings.
- R7 **Data privacy:** data cannot leave the hospital site.

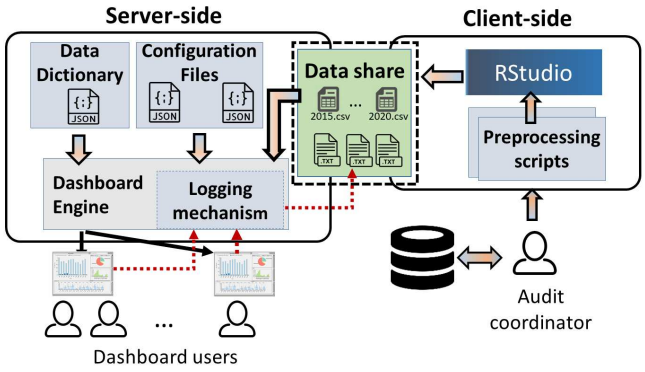
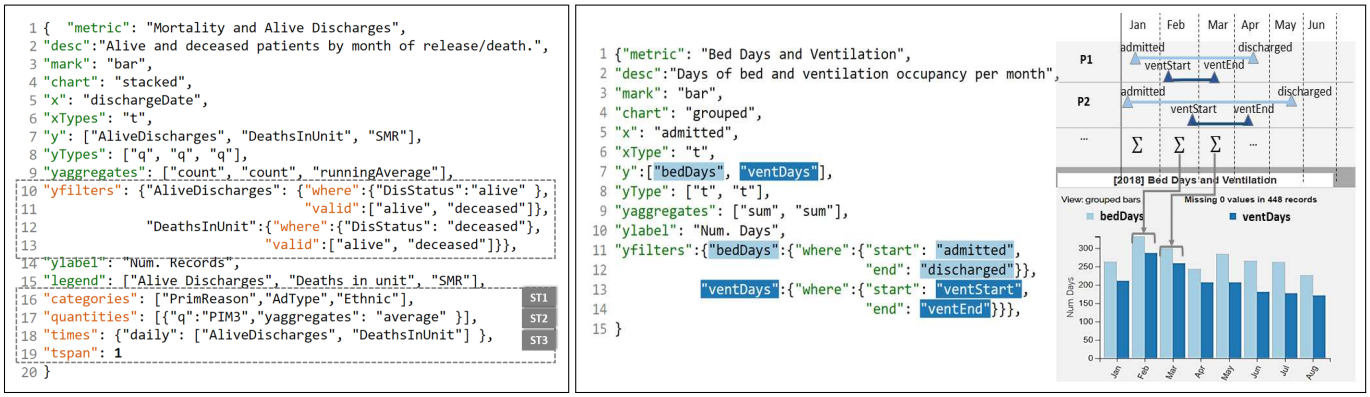


Fig. 4. The QualDash client-server architecture: the dashboard engine reads an array of MSSs from configuration files and fetches audit data supplied by an audit coordinator, to generate the QualCards.

5 QualDash Design

QualDash is a web-enabled dashboard generation engine that we designed, implemented and deployed to meet the above requirements.The QualDash architecture (Figure 4) consists of client-side and server-side components. Both the client and the server are setup locally within each hospital site so that data never leaves the site (R7). Audit data is supplied by an audit coordinator using a client machine and kept at an on-site data share that is accessible from the server.

To support timeliness (R4), QualDash includes an R script that performs pre-processing steps and uploads data to the shared location. Data pre-processing includes calculations to: (i) convert all date formats to the Portable Operating System Interface (POSIX) standard format [5] to support an operating-system-agnostic data model; (ii) calculate derived fields which are not readily available in audit data (e.g., we use the EMS R package [27] to pre-calculate the risk-adjusted Standardised Mortality Ratio (SMR) measure from PICANet data); and (iii) organise audit data into separate annual files for efficient loading in a web browser.



(a) (b)

Fig. 5. Examples of the MSS for: (a) The "Mortality and Alive Discharges" QualCard that is shown in Figure 1. The outlined parts manage inclusion criteria (Lines 10 – 13) and subsidiary tasks (Lines 16 – 19). (b) The "Bed days and Ventilation" QualCard includes temporal measures, derived from the `start` and `end` variables specified in `yfilters`. The QualDash engine breaks down intervals between `start` and `end` to calculate aggregates.

On the server-side, we developed a web tool that allows users to specify an audit and a time frame, and renders the corresponding dashboard in a web browser running a backend dashboard generation engine. The dashboard is dynamically generated and rendered from a JSON configuration file which resides on the server (R1). Configuration files supply an array of Metric Specification Structures (MSSs) to the QualDash engine, which in turn generates the corresponding QualCards. Each QualCard is a self-contained building block for dashboards that encompasses all entry-point and subsidiary information relating to a single metric (R2). A data dictionary is supplied by the audit supplier and contains natural language text descriptions of individual fields. Finally, a logging mechanism records users' interactions. Anonymous usage logs (R7) are fed back to the data share so that they can be accessed by the audit coordinator who then sends us the logs. The remainder of this section details the MSS design and describes how the dashboard engine interprets its elements to generate the QualCard interface.

5.1 The Metric Specification Structure (MSS)

The MSS is a self-contained JSON structure that includes information to dynamically generate concatenated views and support task sequences for an individual QI metric, including main measures and subsidiary components (R1- R2). This only requires a subset of the expressivity of general-purpose visualisation grammars like Vega-Lite [45]. Unlike Vega-Lite, which provides a large variety of possibilities for view composition via faceting, concatenation, layering and repeating views, the MSS provides a constrained design space that sets specific relationships across concatenated views, and allows us to concisely configure a QualCard; while leaving it up to the QualDash engine to set defaults that are common across metrics.

Figure 5*a* shows an example MSS which configures the "Mortality and Alive Discharge" QualCard. The MSS defines: *(i)* a metric name and a description of its main measures (Lines 1 – 2); *(ii)* a compact version of Vega-lite's `data`, `mark` and `encoding` fields (Lines 3 – 9); *(iii)* inclusion filters (Lines 10 – 13); *(iv)* an axis label and view legend (Lines 14 – 15); and *(v)* information for the subsidiary views (Lines 16 – 19). Appendix D provides the specifics of each of these MSS keys and maps them to the corresponding predicates in Vega-Lite [45]. We focus our discussion here on keys that capture the most relevant functionality for QualDash (outlined in Figure 5*a*).

The `yfilters` key allows the specification of inclusion criteria for patients considered in each measure. In the mortality metric example shown in Figure 5*a*, both the `AliveDischarges` and `DeathsInUnit` measures specify filtering criteria based on the discharge status of a patient as `"alive"` and `"deceased"`, respectively. In cases where multiple key-value pairs are used to filter the data, we define an `operator` that combines multiple criteria (not shown in Figure 5) using either a logical AND or OR operator. At the moment, the QualDash engine does not support composite criteria as this was deemed unnecessary for the majority of the healthcare QI tasks collected in our analysis. In the rare cases where composite criteria are required, we offload part of the composition to our R pre-processing scripts. Section 6.2.2 describes a use case that exemplifies this scenario.

The `yfilters` key extends Vega-Lite's `Filter` transform [45] in two ways: (1) To capture information on data quality (R5), we include a `valid` field that, rather than checking for missing data only, accepts a list of valid values for each measure. This is to accommodate audits where invalid entries are coded with special values in the data. (2) To support temporal aggregation, we define two special keys within the `where` clause of `yfilters`. The `start` and `end` keys specify boundary events in cases where a measure spans a period of time. For example, bed and ventilation days per month are the two main measures shown in Figure 5*b*. Since each patient can occupy a bed or a ventilator for an elongated period of time, we specify two derived fields to define these measures: `bedDays` and `ventDays` (Line 7 in Figure 5*b*). The QualDash engine looks for hints in the `where` clause to derive these measures. The dates on which a patient was `admitted` to and `discharged` from a hospital specify the `start` and `end` events of the `bedDays` measure, respectively. The QualDash engine calculates the days elapsed between these two events, and aggregates the corresponding monthly bins. Similarly, the `ventDays` measure is calculated using the time elapsed between the `ventStart` and `ventEnd` events.

For a QualCard's subsidiary views, a `categories` field defines what categorical variables are used to break down the main measures of the metric (`ST1`). For example, when clinicians regard patient mortality in a Pediatric Intensive Care Unit (see Line 16 in Figure 5*a*), they consider a breakdown of the primary reason for admission (`PrimReason`). They also check the admission type (`AdType`); and investigate whether a specific ethnicity had a higher mortality rate (`Ethnic`). A `quantities` field (Line 17 in Figure 5*a*) captures additional measures that users link to the main measures and would want to understand within the same context (`ST2`). The `quantities` are defined using the variable name(s) and an aggregation rule that is to be applied. For both the main view and the `quantities` view, the `yaggregates` key supports `count`, `sum`, `runningSum`, `average` and `runningAverage` aggregation rules. Finally, the `times` field (line 18 in Figure 5*a*) specifies a default temporal granularity for additional historical context (`ST3`). This field accepts a key-value pair where the key describes the time unit to be displayed by default, and the value lists measures that need to be displayed at this granularity. The number of years included in this temporal context is defined by the `tspan` field (line 19 in Figure 5*a*).

The MSS keys described in this section allow dashboard authors to ensure its safe use and interpretation (see Section 6.2.2) by capturing definitions that lead to a patient's inclusion in a measure. They also capture known task sequences that were identified in the analysis phase.

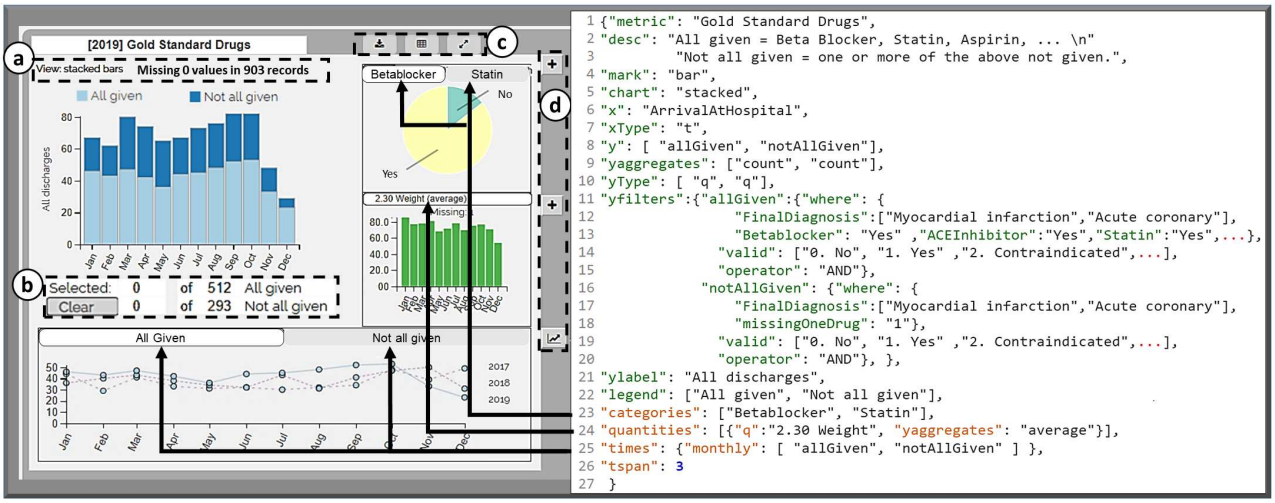


Fig. 6. QualCard for the Gold Standard Drugs metric (left) and the corresponding MSS (right). (a) information on data missingness (derived from lines 14 and 19 in the MSS); (b) selection information for each measure; (c) control panel; and (d) sub-view customisation buttons.

5.2 The QualCard Interface

The purpose of a QualCard is to act as a self-enclosed reusable and dynamically-generated visualisation container which captures task sequences pertaining to a single QI metric. To design this card metaphor, we first introduced our front-line analysts to the idea of a *Metric Bubble*, inspired by Code Bubbles [6] and VisBubbles [24], as enclosed views that allow flexible dragging and resizing. The idea was well-received. However, the free dragging behaviour was perceived as overwhelming. One clinician argued for a more restricted mechanism. In response, we considered two layout alternatives: a tabbed view and a grid view [22]. Empirical evidence shows that juxtaposed patches on a grid layout significantly reduce navigation mistakes [18]. Additionally, input from co-design showed an expectation by stakeholders to be able to obtain an overview of all metrics on the same screen at the entry point of their analysis. We therefore opted to display QualCards on a grid (Figure 1).

QualCard Structure. We define two states for each QualCard: entry-point and expanded (R2), as shown in Figure 1. In its entry-point form, a metric card only shows a main measures view which caters to the entry-point task(s). In its expanded form, a metric card adds three customisable sub-views to answer different subsidiary tasks of the types specified in `ST1`, `ST2` and `ST3`. Given that each metric's task sequence encompasses multiple tasks of each type, we designed navigation tabs in each subsidiary view to allow users to toggle between different `categories`, `quantities` and `time` granularities (R3). Figure 6 shows the mapping from subsidiary tasks in a MSS to a QualCard for the "Gold Standard Drugs" metric. There are two tabs in the categorical sub-view that correspond to the two entries in the `categories` key on line 23. Similarly, there is only one tab on the quantitative sub-view and two tabs in the temporal sub-view, which correspond to elements of the `quantities` and `times` keys, respectively. A similar mapping can be found between the mortality MSS in Figure 5*a* and the sub-views of the mortality QualCard in Figure 1 (labeled *b*, *c*, and *d*).

Visual Encoding. We iterated over design alternatives for visual encodings (i.e., mark types) to define preset defaults for each QualCard sub-view. Alternatives were drawn from theory [11, 29] and practice [13]. We elicited inputs from front-line analysts to: *(a)* set defaults for the visual encodings, and *(b)* design a minimalistic set of GUI elements that support interaction. For the main view of a metric card, bar charts were strongly advocated by our stakeholders. This was clear in the designs collected in our co-design activities (Figure 3). Other chart types including funnel plots [14] and run charts [36] are commonly used in practice to report on audit metrics and were also reflected in the co-design sketches. Analysts acknowledged that while funnel plots can help visualise institutional variations for a metric, they do not answer questions of what is happening in a specific unit over time. Run charts are a common way of process monitoring over time, where the horizontal axis is most often a time scale and the vertical axis represents the quality indicator being studied [36]. We use the same definition for the x- and y-axes of the main view and provide a choice between line and bar mark types. Generally speaking, analysts favored bars to line charts. One PICU consultant stated *"If you give me a bar chart, I don't need to put on my glasses to notice a month with unusually high mortality"*. This is in line with co-design results reported in [26].

The choice of mark is specified in the MSS using the `mark` and `chart` keys in addition to a logic that is inferred from the `yaggregates` key. To illustrate the latter, the example shown in Figure 5*a* tells the engine that there are different aggregation rules for the main measures in this metric. The first two measures represent counts, while the third is a running average. The QualDash engine handles this by generating a dual axis bar chart with differently colored bars representing the two count measures and a line representing the running average. The engine does not allow more than two different rules in `yaggregates`. The `chart` key tells the QualDash engine which type of bar chart to use for each metric. Supported chart types include stacked (Figure 6) and grouped (Figure 5*b*) bars. Small multiples of simple bar charts were also supported in early iterations of the QualDash prototype, but were deemed less preferable during co-design activities, due to their small size within QualCards. As an alternative, metrics can be rendered in multiple QualCards, effectively using the cards as "multiples".

To set defaults for subsidiary views, we showed different alternatives to analysts and asked them to express their preferences for categorical (ST1), quantitative (ST2), and temporal sub-views (ST3). To support `ST1`, a sub-view encodes each of the `categories` that can be used to break down record groups. We use an interactive pie chart for this sub-view (Figure 1*b*). This design choice is motivated by two factors: *(i)* a preference expressed by our analysts to distinguish the categorical domain of these measures from the temporal domain that forms the basis of bar charts in other sub-views (this is in line with best practices for keeping multiple views consistent [38]); and *(ii)* the prevalence of pie chart use in healthcare QI reports which leverages users' familiarity.

The metric card structure facilitated the discussion around modular interaction design decisions. Subsidiary views are linked to the entry-point view, in a manner that is specific to the view's corresponding tasks. Brushing a pie chart in a `categories` sub-view highlights the distribution of the corresponding category over time in the main measures view. The converse is also true, in that brushing one or more month(s) in the bar chart causes the pie chart to respond and display the distribution of categories within the brushed cohort. Consider, for example, the expanded mortality QualCard shown in Figure 1 (right). By highlighting a month with a relatively low number of discharged patients and a low number of deaths (May), one can see that 51% of

patients seen leaving the hospital in that month were recovering from surgery, only 4% had bronchiolitis, and the remaining patients had other reasons for admission. These percentages are displayed on the pie chart upon mouse hover. This design bears some similarity to Tableau's part-to-whole analysis using set actions [28].

The `quantities` sub-view contributes additional measures that complement the main measures in the entry point view (`ST2`). We use a bar chart for this sub-view and extend the same color palette used for the main view to the measures shown in the `quantities` sub-view. To avoid excessive use of color in a single QualCard, we limit the number of measures shown in the entry-point view to a maximum of five measures, and the number of tabs in the `quantities` sub-view to five tabs. For metrics that require larger numbers of measures to be displayed, we split them into two or more QualCards and include the appropriate combinations of measures in each. Extending the color palette means that the colors used in the entry-point view are not repeated in the `quantities` sub-view. This design decision was made to avoid any incorrect association of different measures within one QualCard. Highlighting a bar in the main view emphasizes the bar corresponding to the same time in this sub-view and adds translucency to other bars to maintain context (Figure 1*c*).

The `times` sub-view adds temporal contexts (`ST3`). In an early design, this view presented a two-level temporal aggregation to facilitate comparison of the same quarter / month / etc. across years. This early design used a small multiples view that drew small line charts in each multiple linking the same time unit across years. A button enabled users to toggle between this alternative and a traditional multi-series line chart. During an evaluation phase (described in Section 6), participants argued against the small multiples view and preferred to navigate between multi-series line charts of different temporal granularities. This co-design outcome is depicted in Figures 1 and 6 which display daily and monthly aggregates, respectively. The number of years shown in this view is determined by the `tspan` key.

In addition to visual encodings, some textual information is displayed in a QualCard. The `metric` MSS key specifies the title of the QualCard and the `desc` key specifies QualCard description that is displayed in a tooltip upon mouse-hover on the title. Figure 6*a* shows information on the visualisation technique used in the main measures view and indicates the quality of the metric's underlying data by listing the number of missing / invalid values out of the total number of records. The area delineated in Figure 6*b* displays the number of selected records out of the totals in each of the displayed measures. It also includes a "clear" button that clears any existing selection. Hovering the mouse on any tab in the sub-views pulls a description of the corresponding data field from the data dictionary and displays it in a tooltip.

GUI Interactions. Users can customise the measures shown in each sub-view (R3) via GUI buttons (Figure 6*d*), which allow flexible addition / removal of measures (i.e. tabs) in the categories and quantities sub-views and managing time granularities in the times sub-view. An example is shown in Figure 1*e* for modifying the categories displayed. Additionally, a grey area that outlines a QualCard acts as a handle that can be used to drag and reposition the card on the grid or expand it, when double-clicked. A "control panel" is included in each QualCard (Figure 6*c*), which contains buttons to download the visualisations shown on the card, and export selected records to a tabular view. These exports support the creation of reports and presentations for clinical governance meetings (R6). Finally, a button to expand the QualCard was added to complement the double-clicking mechanism of the QualCard's border. This button was added to address feedback we received in evaluation activities, described in Section 6.

Dashboard Layout. We allow users to toggle between `1x1`, `2x3`, and `2x2` grid layouts to specify the number of QualCards visible on the dashboard. Selecting the `1x1` layout expands all QualCards and allows users to scroll through them, as one would a PDF slide presentation. The other two layouts display QualCards in their entry-point form and use screen real estate to render them in either a 2x2 or a 2x3 grid. The engine does not set limits on the number of QualCards to be rendered.

6 Evaluation and Deployment

We conducted a series of off-site and on-site evaluation activities to validate the **coverage** and **adaptability** of QualDash. We define coverage as the ability of the QualCards to be pre-configured (R1) or expanded (R2) to cater to the variety of tasks identified through our task analysis. Adaptability is the ability of the QualCards to support different lines of inquiry as they arise (R3). Additional off-site activities aimed to predict the usability of the generated dashboards; and through on-site observations we sought to establish evidence of the usefulness of QualDash in a real-world healthcare QI setting. In this section, we describe the process and outcomes of 62 contact hours of evaluation.These activities were distributed across our project timeline, with each activity serving a summative evaluation goal, as defined by Khayata et al [23].

6.1 Off-site Evaluation

We used qualitative methods to establish the coverage and adaptability of the QualCard and the MSS. For this, we conducted three focus group sessions with hospital staff. We elicited the groups' agreement on the level of coverage that the QualDash prototype offered for the tasks, identified through interviews, and ways in which it could be adapted to new tasks generated through the groups' discussion.

The first focus group session featured an *active intervention with a paper prototype* [25]. The intention was to free participants from learning the software and rather set the focus on the match between the task sequences, the skeletal structure of the QualCard and the default choice of visualisation techniques in each of its views. Appendix C in our supplementary material details the activities of this session and provides a listing of the artefacts used. Participants were divided into three groups. Each group was handed prototype material (artefacts A, B and C in Appendix C) for a set of metric cards. For each card, the paper prototype included a task that was believed to be of high relevance, screen printouts of metric card(s) that address the task, and a list of audit measures that were used to generate the cards. A group facilitator led the discussion on each metric in a sequence of entry-to-subsidiary tasks, and encouraged participants to change or add tasks, comment on the data used, and sketch ideas to improve the visualisations.

After this first session, two sessions featured a *think-aloud protocol* [42] with participants in two sites. One of these two groups had engaged with the paper prototype, but both groups were exposed to the dashboard software for the first time during the think-aloud session. In each session, we divided participants into groups, where each group consisted of 1-2 participants and a facilitator using one computer to interact with the dashboard. The facilitators gave a short demo of the prototype. As an initial training exercise, the facilitator asked participants to reproduce a screen which was printed on paper. This exercise was intended to familiarize participants with the dashboard and the think-aloud protocol. Following this step, the facilitators presented a sequence of tasks and observed as participants took turns to interact with the dashboard to perform the tasks. Each session lasted for 75 minutes and we captured group interactions with video and audio recording.

Seventeen participants took part in the sessions, including information managers, Pediatric Intensive Care Unit (PICU) consultants, and nurses. We analyzed the sketches, observation notes, and recordings from all sessions and divided feedback into five categories:

- A *task-related* category captured the (mis)match between participants' intended task sequences and view compositions supported in the dashboard (R2).
- A *data-related* category captured comments relating to the data elements used to generate visualisations. This was to assess our findings (**F1** – **F3**) regarding data types included in the structure of the MSS (R1, R2).
- A *visualisation-related* category captured feedback on the choice of visual encoding in each view.
- A *GUI-related* category captured comments made on the usability of the interface.
- An *"Other"* category reported any further comments.

The three sessions and follow-up email exchanges with clinicians resulted in a total of 104 feedback comments. Task- and data-related feedback constituted 22% of the comments. Data-related feedback

at this stage of evaluation focused on issues like data validation and timeliness, rather than on the choice of data elements used to generate the QualCards. This focus shifted later when we introduced QualDash into the sites, at which point our evaluation participants took interest in accurately specifying the data elements used to generate each card. Nonetheless, our off-site activities captured a number of comments regarding aggregation rules. For example, it was noted that Standardised Mortality Ratio (SMR) should be displayed as a cumulative aggregate.

For task-related comments, we captured feedback in which participants noted a (mis)match between their tasks and the MSS. Participants requested adaptations that were in most cases supported by the MSS. One example is for the call-to-balloon metric, where clinicians wanted to include a monthly breakdown of patients who did not have a date and time of reperfusion treatment stored in the audit dataset. They explained that this would allow an investigation of whether STEMI patients admitted in a time period did not receive the intervention at all; or they did receive it but data was not entered in the audit. Accommodating such a request was done by adding a measure labeled as *"No PCI data available"* to the call-to-balloon MSS; for which we selected records having a diagnosis of `Myocardial infarction (ST elevation)` and a value `NA` as date of reperfusion treatment.

Additional tasks were collected throughout the activities. Requests were further categorised into customisation issues which could be addressed by simply modifying the corresponding MSS, and design issues, which were addressed in a subsequent design and development iteration. An example of the latter is a task that was requested by two PICU clinicians and required adding new MSS functionality. This task enquired about the last recorded adverse event for different metrics.

Visualisation and GUI-related feedback constituted 21% of the collected comments. These comments were largely positive and included a few suggestions to improve readability (e.g., legend position, size of labels, etc.). One participant commented: *"People approaching the visualisation with different questions can view consistent data subsets and that's good, because people will try different [sequences] but they're getting the same answer so this gives us a foolproof mechanism"*.

In addition to this qualitative feedback, we predicted the usability of the dashboards by administering a System Usability Scale (SUS) questionnaire [8] at the end of each think-aloud session. Participants completed the questionnaire after completing the tasks. Usability scores from the two participant groups were 74 in the first session and 89.5 in the second session, which indicates very good usability.

Enhanced MSS and QualCard. To provide textual information about adverse events, we added a key in the MSS to specify an `event` which specifies the type of adverse event that clinicians are interested to learn about for the metric. We further capture the event's `name`, `date`, a `desc` field which describes the event in plain text, and an `id` field which points to a primary key in the data that identifies the record involving the last reported incident. This information gets appended as text to the QualCard description tooltip which appears upon mouse hovering the QualCard's title (see Appendix D for more details).

6.2 Usefulness of Dashboards in Deployment

We conducted installation visits at the five hospitals to deploy the QualDash software. Prior to each visit, a hospital IT staff member helped us by setting up a server virtual machine, to which we were granted access via remote desktop. This made it possible for us to access both the client and server on the same physical computer within each site. During each visit, a staff member downloaded raw audit data from the audit supplier's web portal and passed it to the pre-processing R scripts, which in turn fed the data into QualDash. We ran validation tests to ensure that the data were displayed correctly. Through this process, we realised that field headers for the MINAP audit were not unified across sites, due to different versions of the audit being used. The QualDash architecture allowed this adaptation by modifying the field names in the config files to match the headers used in each site. This adaptation process took approximately 30-60 minutes in each site.

Following installation visits, we held a series of 10 meetings with clinical staff and data teams in the sites, with the aim of collecting evidence of QualDash's ability to support things like data timeliness (R4), quality (R5) and perceived usefulness of the dashboards' functionality (R1, R2, R3, and R6). The remainder of this section summarises the evidence we collected along these criteria.

6.2.1 Support for Data Timeliness

The client-server architecture of QualDash allows data uploads to take place from any computer in the hospital that has R installed. This process was perceived as intuitive by the majority of audit coordinators, who were in charge of data uploads. Of the five consultant cardiologists and four PICU consultants that we met, three explained that a monthly data upload was sufficient for their need; while others explained that they prefer uploads to be as frequent as possible. One PICU consultant explained that if QualDash was to be updated every week, this would allow them to keep a close monitoring of "rare events" such as deaths in unit and accidental extubation. From the audit coordinators' perspective, monthly uploads were decided to be most feasible to allow time for data validation before feeding it into QualDash. One audit coordinator agreed with PICU consultants at her site to perform weekly PICANet uploads. In another site, one IT member of staff explained that they would run a scheduled task on the server to support automated monthly MINAP uploads. In the general case, however, data upload schedules have been ad hoc and have been driven primarily by need.

All stakeholders appreciated that QualDash allows the right level of flexibility to support each site's timeliness requirements (R4). One PICU audit coordinator explained that she was keen on uploading data into QualDash to obtain information which she needed to upload into a database maintained by NHS England to inform service commissioning. This is typically done every three months. She explained that the process of extracting data aggregates to upload into this database used to take her up to two hours. However, with the use of QualDash she was able to perform this task in just 10 minutes.

Response to COVID-19 One cardiologist (and co-author of this paper) requested to ramp up MINAP data uploads to a daily rate at their site. This was in response to early reports of STEMI service delays in parts of the world during the COVID-19 pandemic [15, 50]. The cardiologist highlighted the need to monitor the call-to-balloon metric card during this time to detect any declines in the number of STEMI admissions (in cases where patients are reluctant to present to the service) and the number of patients meeting the target time. This request was forwarded to audit coordinators on the site, who in turn assigned the role of daily validation and upload to the site's data team.

6.2.2 Safe Interpretation: A Case Study

One of the main motivations behind the client-server architecture and the use of MSSs is to ensure that users looking at a specific performance metric share a common understanding of how their site is performing on the metric before their data gets pushed into the public sphere through a national annual report. Safe interpretation of information in this context relies on capturing patients' care pathways (R1) which determine a patient's eligibility for inclusion within that metric. The MSS and corresponding metric structure of QualDash allowed for focused discussions with stakeholders at different sites. We reflect here on one particular metric called "Gold Standard Drugs" (Figure 6), to demonstrate QualDash's support for safe interpretation.

The Gold Standard Drugs on discharge metric captures the number of patients who are prescribed correct medication upon being discharged from a cardiology unit in a hospital. Early co-design activities showed that the main task that users sought to answer for this metric was: *What is the percentage of patients discharged on correct medication per month?* Co-designers indicated that there are five gold standard drugs that define what is meant by "correct medication". These include betablocker, ACE inhibitor, Statin, Aspirin and P2Y12 inhibitor. Subsidiary tasks for this metric include investigating months with outlier proportions of patients receiving gold standard treatment, for those months, users asked questions such as: Which medication was mostly missing from the prescriptions (`ST1`)? Did the case mix have more patients that were not fit for the intervention (`ST1`)? What was the average weight of patients (`ST2`)? How does the number of prescriptions compare to the same month last year (`ST3`)?

Upon deploying QualDash in one of the sites, two cardiologists pointed out that this metric should not capture the entire patient population but should rather focus on patients eligible for such prescription. They explained that eligible patients can be determined from the patients' diagnosis but there was uncertainty about which diagnoses should be included. In response to this, we removed the corresponding QualCards from the MINAP dashboards while our team further investigated the patient inclusion criteria for this metric. The flexibility of metric card removal was especially beneficial in this case to avoid inaccurate interpretation. We removed the MSS from the configuration files. The remaining parts of the dashboard were not affected.

Upon further investigating this metric with audit suppliers, we learned that there are only two patient diagnoses that establish eligibility to receive gold standard drugs: "Myocardial infarction (ST elevation)", and "Acute coronary syndrome (troponin positive)/ nSTEMI". We updated the MSS accordingly and added this QualCard back. Here, a known limitation of the MSS design resurfaced when specifying the `yfilters` for patients who did not receive all five drugs. This group presented a composite expression that is not currently supported in QualDash which can be formulated as: `included_patient`:= $((betablocker = FALSE) \,||\, (apsirin = FALSE) \,||\, (statin = FALSE) \,||\, (ACEInhibitor = FALSE) \,||\, (P2Y12Inhibitor = FALSE))$ & $(finalDiagnosis \in$ [`'Myocardial infarction (ST elevation)','Acute coronary syndrome (troponin positive)/ nSTEMI'`]). To support this composition of & and || operators, we offloaded part of the calculation to a pre-processing step. Namely, we added a line in the R script to pre-calculate a field called *missingOneDrug* which captures the first term of the composition. This simplified the filter to: `included_patient`:= $(missingOneDrug)$ & $(finalDiagnosis \in$ ['Myocardial infarction (ST elevation)','Acute coronary syndrome (troponin positive)/ nSTEMI']). The latter fitted nicely in our MSS as shown in Figure 6. We conclude from the case of drugs on discharge that QualDash's process for MSS configuration enables the management of individual QualCards in different sites. This allows time for the dialogue around the correct data definitions and to verify them from the supplier before pushing the cards back into the sites, to ensure safe interpretation of visualisations.

6.2.3 Perceived Usefulness: A Case Study

To investigate the perceived usefulness of QualDash, we present here the case of the mortality metric in the PICUs. For this metric, early analysis from our interviews and co-design activities revealed a sequence of tasks that begin with two main questions: T1: What is the trend of risk-adjusted standardised mortality rate (SMR) over time? T2: What is the raw number of deaths and alive discharges per month? From these two entry point tasks, a sequence of subsidiary tasks investigates the case mix (`ST1`). One co-design participant explained the importance of comparing death counts with the same month last year (`ST3`) as it gives an indication of performance in light of seasonal variations of the case mix. Additionally, an interviewee explained the importance of considering relevant measures in the same context (`ST2`) such as the average PIM score (Pediatric Index of Mortality) explaining that *"you say, okay you've had 100 admissions through your unit, and based on PIM2 scoring, we should not expect worse than five deaths."*

To support these tasks, we designed a MSS that captures alive and deceased patients on a monthly basis (Figure 5). We added the SMR measure which is aggregated as a running monthly average. Sub-views include primary diagnosis (for case mix) and monthly averages of PIM score. Adaptation requests for this card included changes to the x-axis variable, such as deaths to be aggregated in the months in which they occurred or aggregated by date of patient admission.

When discussing this card with a PICU consultant, she noted *"If we have high SMR, that's a living nightmare for us, as we would need to investigate every single death. [With QualDash] I can export these deaths and look into the details of these records. That's very good. Someone can come in and say your SMR is too high and I can extract all the deaths that contributed to this SMR with 15 seconds effort."* As she then looked at the PIM score subview, she noted *"we can also use this to say we need to uplift the number of nurses in [months with high PIM score]. This will be very useful when we do reports and we interface with management for what we need"*. A clinician in another site observed mortality by ethnic origin (`ST1`) as he noted *"looking at families of Asian origin, survival rates are better than predicted."* This clinician also noted that the textual display of last recorded event is particularly useful for their team. He explained that this information enables him to keep his co-workers motivated for QI by saying something like *"alright, our last extubation was a couple of weeks ago, let us not have one this week."*

7 Conclusions & Future Work

We presented a dashboard generation engine that maps users' task sequences in healthcare QI to a unit of view composition, the QualCard. Our MSS offers a targeted and more concise specification, compared to expressive grammars like Vega-lite [45] and Draco [33], and this is a key reason why QualDash was straightforward to adapt during deployment in the five hospitals. That made it easy to correct small but important mismatches between clinicians' tasks and our original misunderstanding of them (e.g., the call-to-balloon metric), accommodate new tasks (e.g., for adverse events) and allow site-specific changes.

The lessons we have learned and factors in QualDash's positive reception lead us to the following recommendations for other design studies: **(a) Trust** in visualised information is enhanced by a level of moderation for dashboard authoring. In healthcare QI, quality metrics have specific patient inclusion criteria that reflect national and international standards. MSS configuration files acted as a communication medium between our visualisation team, clinicians and support staff. This allowed for moderated view definitions that ensured safe interpretation of the visualisation. **(b) Modular view composition**, as supported in the QualCards, enables focused communication between dashboard authors, users and system administrators. Comments on QualDash were fed back to us regarding specific QualCards, which enabled refinement and validation iterations to affect localised metric-specific views while leaving the remaining parts of the dashboard intact. **(c) Sequenced rendering** of views, which is materialised by QualCard expansion, provided a metaphor that captured dashboard users' task sequences and lines of enquiry pertaining to different metrics. Further evidence is required to establish the usability of the MSS as an authoring tool, and for that we have commenced a field evaluation of QualDash in the five hospitals. The results will be reported in a future paper.

We explored the generalisability of the Qualcard through discussions and demonstrations with clinicians and Critical Care research experts who are outside of the QualDash stakeholder community. The idea of custom-tailored visualisation cards that capture tasks sequences was very well received. This has led to budding collaborations as demand for this type of adaptable dashboard generation is gaining momentum with the diversity of tasks surrounding COVID-19 data analysis. We have received questions about whether we can generate QualCards to support decision makers' understanding of risk factors and vulnerable communities. We have also received some questions about the possibility of adding more visualisation techniques, like maps for geospatial data. In the current version of QualDash, we only support time and population referrer types [3]. However, based on these discussions, we foresee great opportunities to further develop our engine and support more referrers like geographic location.

Finally, we plan to identify new design requirements for possible transitions across QualCards. We expect that the modular nature of the self-contained QualCard will help focus these design decisions to localised areas of the dashboard screen and to specific task sequences. Such transitions were not found necessary for healthcare QI dashboards in our experience, but may be deemed necessary in other applications.

Acknowledgments

This research is funded by the National Institute for Health Research (NIHR) Health Services and Delivery Research (HS&DR) Programme (project number 16/04/06). The views and opinions expressed are those of the authors and do not necessarily reflect those of the HS&DR Programme, NIHR, NHS or the Department of Health.

REFERENCES

[1] J. Ahn, C. Plaisant, and B. Shneiderman. A task taxonomy for network evolution analysis. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):365–376, 2014.

[2] R. A. Amar and J. T. Stasko. Knowledge precepts for design and evaluation of information visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):432–442, 2005.

[3] N. Andrienko and G. Andrienko. *Exploratory analysis of spatial and temporal data: a systematic approach*. Springer Science & Business Media, 2006.

[4] A. Assiri, M. Zairi, and R. Eid. How to profit from the balanced scorecard: An implementation roadmap. *Industrial Management & Data Systems*, 106(7):937–952, 2006.

[5] V. Atlidakis, J. Andrus, R. Geambasu, D. Mitropoulos, and J. Nieh. Posix abstractions in modern operating systems: The old, the new, and the missing. In *Proceedings of the Eleventh European Conference on Computer Systems*, pp. 1–17, 2016.

[6] A. Bragdon, S. P. Reiss, R. Zeleznik, S. Karumuri, W. Cheung, J. Kaplan, C. Coleman, F. Adeputra, and J. J. LaViola Jr. Code bubbles: rethinking the user interface paradigm of integrated development environments. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1*, pp. 455–464. ACM, 2010.

[7] M. Brehmer and T. Munzner. A multi-level typology of abstract visualization tasks. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2376–2385, 2013.

[8] J. Brooke et al. SUS-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.

[9] M. Cohn. *User stories applied: For agile software development*. Addison-Wesley Professional, 2004.

[10] D. Dowding, R. Randell, P. Gardner, G. Fitzpatrick, P. Dykes, J. Favela, S. Hamer, Z. Whitewood-Moores, N. Hardiker, E. Borycki, et al. Dashboards for improving patient care: review of the literature. *International journal of medical informatics*, 84(2):87–100, 2015.

[11] M. Elshehaly, N. Alvarado, L. McVey, R. Randell, M. Mamas, and R. A. Ruddle. From taxonomy to requirements: A task space partitioning approach. In *2018 IEEE Evaluation and Beyond-Methodological Approaches for Visualization (BELIV)*, pp. 19–27. IEEE, 2018.

[12] S. Few. Common pitfalls in dashboard design. *Perceptual Edge*, 2006.

[13] Financial Times. Visual vocabulary, 2019. (accessed April 27, 2020) `https://ft.com/vocabulary`.

[14] C. P. Gale, A. P. Roberts, P. D. Batin, and A. S. Hall. Funnel plots, performance variation and the myocardial infarction national audit project 2003–2004. *BMC cardiovascular disorders*, 6(1):34, 2006.

[15] S. Garcia, M. S. Albaghdadi, P. M. Meraj, C. Schmidt, R. Garberich, F. A. Jaffer, S. Dixon, J. J. Rade, M. Tannenbaum, J. Chambers, P. P. Huang, and T. D. Henry. Reduction in st-segment elevation cardiac catheterization laboratory activations in the united states during covid-19 pandemic. *Journal of the American College of Cardiology*, 2020. doi: 10.1016/j.jacc.2020.04.011

[16] D. Gotz and H. Stavropoulos. Decisionflow: Visual analytics for high-dimensional temporal event sequence data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1783–1792, 2014.

[17] J. Heer and B. Shneiderman. Interactive dynamics for visual analysis. *Commun. ACM*, 55(4):45–54, Apr. 2012. doi: 10.1145/2133806.2133821

[18] A. Z. Henley, S. D. Fleming, and M. V. Luong. Toward principles for the design of navigation affordances in code editors: An empirical investigation. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 5690–5702. ACM, 2017.

[19] M. C. Jones, I. R. Floyd, and M. B. Twidale. Teaching design with personas. In *Proceedings of the International Conference of Human Computer Interaction Educators (HCIed)*, 2008.

[20] R. S. Kaplan and D. P. Norton. The balanced scorecard - measures that drive performance. *Harvard Business Review*, 70(1):71–78, 1992.

[21] N. Kerracher and J. Kennedy. Constructing and evaluating visualisation task classifications: Process and considerations. *Computer Graphics Forum (Proceedings of EuroVis*, 36(3):47–59, 2017.

[22] A. Key, B. Howe, D. Perry, and C. Aragon. Vizdeck: Self-organizing dashboards for visual analytics. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, p. 681–684. Association for Computing Machinery, New York, NY, USA, 2012. doi: 10.1145/2213836.2213931

[23] M. Khayat, M. Karimzadeh, D. S. Ebert, and A. Ghafoor. The validity, generalizability and feasibility of summative evaluation methods in visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):353–363, 2020.

[24] G. Li, A. C. Bragdon, Z. Pan, M. Zhang, S. M. Swartz, D. H. Laidlaw, C. Zhang, H. Liu, and J. Chen. Visbubbles: a workflow-driven framework for scientific data analysis of time-varying biological datasets. In *SIGGRAPH Asia 2011 Posters*, p. 27. ACM, 2011.

[25] D. Lloyd and J. Dykes. Human-centered approaches in geovisualization design: Investigating multiple methods through a long-term case study. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2498–2507, Dec 2011. doi: 10.1109/TVCG.2011.209

[26] M. H. Loorak, C. Perin, N. Kamal, M. Hill, and S. Carpendale. Timespan: Using visualization to explore temporal multi-dimensional data of stroke patients. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):409–418, 2016.

[27] C. C. C. F. Lunna Borges, Pedro Brasil. Epimed Solutions Collection for Data Editing, Analysis, and Benchmark of Health Units. `https://cran.r-project.org/web/packages/ems/ems.pdf`, 2019. [Online; accessed March 2020].

[28] B. Lyons. 8 ways to bring powerful new comparisons to viz audiences with Tableau Set Actions, 2018. Blog post (accessed April 27, 2020): `https://tinyurl.com/tableausetactions`.

[29] J. Mackinlay, P. Hanrahan, and C. Stolte. Show me: Automatic presentation for visual analysis. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1137–1144, 2007.

[30] M. Meyer, M. Sedlmair, P. S. Quinan, and T. Munzner. The nested blocks and guidelines model. *Information Visualization*, 14(3):234–249, 2015.

[31] B. Milash, C. Plaisant, and A. Rose. Lifelines: visualizing personal histories. In *Conference Companion on Human Factors in Computing Systems*, pp. 392–393, 1996.

[32] M. Monroe, R. Lan, H. Lee, C. Plaisant, and B. Shneiderman. Temporal event sequence simplification. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2227–2236, 2013.

[33] D. Moritz, C. Wang, G. L. Nelson, H. Lin, A. M. Smith, B. Howe, and J. Heer. Formalizing visualization design knowledge as constraints: Actionable and extensible models in draco. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):438–448, 2019.

[34] T. Munzner. A nested model for visualization design and validation. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):921–928, Nov. 2009. doi: 10.1109/TVCG.2009.111

[35] T. Munzner. *Visualization Analysis and Design*. CRC Press, 2014.

[36] R. J. Perla, L. P. Provost, and S. K. Murray. The run chart: a simple analytical tool for learning from variation in healthcare processes. *BMJ quality & safety*, 20(1):46–51, 2011.

[37] PICANet. Paediatric Intensive Care Audit Network. `https://www.picanet.org.uk/`, 2018. [Online; accessed 2019].

[38] Z. Qu and J. Hullman. Keeping multiple views consistent: Constraints, validations, and exceptions in visualization authoring. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):468–477, 2018.

[39] R. Randell, N. Alvarado, L. McVey, J. Greenhalgh, R. M. West, A. Farrin, C. Gale, R. Parslow, J. Keen, M. Elshehaly, R. A. Ruddle, J. Lake, M. Mamas, R. Feltbower, and D. Dowding. How, in what contexts, and why do quality dashboards lead to improvements in care quality in acute hospitals? protocol for a realist feasibility evaluation. *BMJ Open*, 10(2), 2020. doi: 10.1136/bmjopen-2019-033208

[40] A. Rind, T. D. Wang, W. Aigner, S. Miksch, K. Wongsuphasawat, C. Plaisant, B. Shneiderman, et al. Interactive information visualization to explore and query electronic health records. *Foundations and Trends in Human–Computer Interaction*, 5(3):207–298, 2013.

[41] J. Rogers, N. Spina, A. Neese, R. Hess, D. Brodke, and A. Lex. Composer: Visual cohort analysis of patient outcomes. *Applied Clinical Informatics*, 10(2):278–285, 2019. doi: 10.1055/s-0039-1687862

[42] Y. Rogers, H. Sharp, and J. Preece. *Interaction design: beyond human-computer interaction*. John Wiley & Sons, 2011.

[43] M. Rueter and E. Fields. Tableau for the enterprise: An IT overview, 2012. White paper (accessed April 27, 2020). `https://tableau.com/learn/whitepapers/tableau-enterprise`.

[44] A. Sarikaya, M. Correll, L. Bartram, M. Tory, and D. Fisher. What do we talk about when we talk about dashboards? *IEEE Transactions on Visualization and Computer Graphics*, 25(1):682–692, 2019.

[45] A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer. Vega-lite: A grammar of interactive graphics. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):341–350, 2017.

[46] A. Satyanarayan, R. Russell, J. Hoffswell, and J. Heer. Reactive vega: A streaming dataflow architecture for declarative interactive visualization. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):659–668, 2016.

[47] H.-J. Schulz, T. Nocke, M. Heitzler, and H. Schumann. A design space of visualization tasks. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2366–2375, 2013.

[48] K. Sedig and P. Parsons. Interaction design for complex cognitive activities with visual representations: A pattern-based approach. *AIS Transactions on Human-Computer Interaction*, 5(2):84–133, 2013.

[49] M. Sedlmair, M. Meyer, and T. Munzner. Design study methodology: Reflections from the trenches and the stacks. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2431–2440, 2012.

[50] C.-C. F. Tam, K.-S. Cheung, S. Lam, A. Wong, A. Yung, M. Sze, Y.-M. Lam, C. Chan, T.-C. Tsang, M. Tsui, et al. Impact of coronavirus disease 2019 (covid-19) outbreak on st-segment–elevation myocardial infarction care in hong kong, china. *Circulation: Cardiovascular Quality and Outcomes*, 2020.

[51] T. D. Wang. *Interactive visualization techniques for searching temporal categorical data*. PhD thesis, 2010.

[52] S. Wehrend and C. Lewis. A problem-oriented classification of visualization techniques. In *Proceedings of the 1st Conference on Visualization '90*, VIS '90, pp. 139–143. IEEE Computer Society Press, Los Alamitos, CA, USA, 1990.

[53] S. Wexler, J. Shaffer, and A. Cotgreave. *The Big Book of Dashboards: Visualizing Your Data Using Real-World Business Scenarios*. John Wiley & Sons, 2017.

[54] C. Wilkinson, C. Weston, A. Timmis, T. Quinn, A. Keys, and C. P. Gale. The myocardial ischaemia national audit project (minap). *European Heart Journal-Quality of Care and Clinical Outcomes*, 6(1):19–22, Jan 2020.

[55] K. Wongsuphasawat, J. A. Guerra Gómez, C. Plaisant, T. D. Wang, M. Taieb-Maimon, and B. Shneiderman. Lifeflow: visualizing an overview of event sequences. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1747–1756, 2011.

[56] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):649–658, 2016.

[57] K. Wongsuphasawat, Z. Qu, D. Moritz, R. Chang, F. Ouk, A. Anand, J. Mackinlay, B. Howe, and J. Heer. Voyager 2: Augmenting visual analysis with partial view specifications. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 2648–2659, 2017.

[58] Y. Zhang, K. Chanana, and C. Dunne. IDMVis: Temporal event sequence visualization for type 1 diabetes treatment decision support. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):512–522, Jan 2019. doi: 10.1109/TVCG.2018.2865076