The Journal of Systems and Software 211 (2024) 112008

Contents lists available at ScienceDirect

The Journal of Systems & Software

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

In Practice

A cloud-based data processing and visualization pipeline for the fibre roll-out in Germany[☆]

Michel Krämer[a,b,*], Pascal Bormann[a,b], Hendrik M. Würz[a,b], Kevin Kocon[a,b], Tobias Frechen[c], Jonas Schmid[d]

[a] *Fraunhofer Institute for Computer Graphics Research IGD, Fraunhoferstr. 5, 64283 Darmstadt, Germany*
[b] *Technical University of Darmstadt, Karolinenplatz 5, 64289 Darmstadt, Germany*
[c] *Deutsche Telekom IT GmbH, Landgrabenweg 151, 53227 Bonn, Germany*
[d] *Deutsche Telekom Technik GmbH, Alpenstr. 8, 87435 Kempten, Germany*

ARTICLE INFO

Keywords:
Data processing
Point clouds
Panorama imagery
Cloud computing
3D visualization

ABSTRACT

To support the roll-out of fibre broadband Internet in Germany, Deutsche Telekom has set itself the goal of connecting more than 2.5 million households per year to FTTH (Fibre to the Home). However, planning and approval processes have been very complex and time-consuming in the past due to high communication overhead between stakeholders, missing automation, and lack of information about planning areas. Telekom addresses this problem by collecting large amounts of geospatial data (3D point clouds and 360° panorama images), which can be used to automatically find suitable routes for fibre optic lines, to determine possible locations for distribution cabinets, as well as to build a 3D visualization helping to create detailed plans and to present them to decision makers. This speeds up planning tremendously, but processing this data and creating the visualization in a short time requires automation. In this systems paper, we present a data processing platform that we have built and operated together with Telekom over the course of the last six and a half years. The platform makes use of the cloud to manage Big Data in a scalable and elastic manner. It builds upon research results from us, specifically a scientific workflow management system to automate processing, as well as Fibre3D, a web-based tool that planners can use to display the processed data and to perform fine-planning. Besides the technological aspects, this paper also describes a practical use case that shows how the platform and Fibre3D help Telekom speed up the planning and approval process. We also summarize lessons learned and give recommendations for the design of systems similar to ours. With the presented technology, Telekom has been able to already connect more than 8 million households to FTTH and expects to even improve on this in the future. We consider our collaboration, therefore, an example of how well knowledge and technology transfer between research and industry can work, and, at the same time, what impact it can have on society.

1. Introduction

Digital transformation affects societies all over the world. More and more people use digital technology in their everyday lives, and businesses adopt analytics, broadband connectivity, and social media to improve outreach, productivity, and in turn market share. While digital transformation offers many opportunities for citizens, governments, and global economies, it also puts them under the pressure to always stay on top of the state of the art to maintain contact with friends, neighbours, and trade partners, as well as to keep up with competitors.

Access to broadband Internet is therefore a fundamental building block of digital transformation. According to the annual statistics published by the OECD (OECD, 2022e), as of December 2022, among the 38 member countries, an average of 34.9% of all inhabitants have a broadband Internet subscription (OECD, 2022f). While Germany ranks third on the list of the total (*absolute*) number of fixed broadband subscriptions among all OECD countries (behind the US and Japan) (OECD, 2022h), the *relative* share of high-speed fibre subscriptions (compared to DSL and cable) lies at only 9.2%, which puts Germany on the 35th place in this list (OECD, 2022g) far below the OECD average of 37.7%. In Korea, which is at the top, the share of fibre lies at 88%.

To improve this situation, the German federal government has put measures into effect to promote the roll-out of high-speed fibre networks (BMDV, 2021a). The German telecommunications provider Deutsche Telekom AG supports this undertaking and has set itself the

[☆] Editor: A. Bertolino.
[*] Corresponding author at: Fraunhofer Institute for Computer Graphics Research IGD, Fraunhoferstr. 5, 64283 Darmstadt, Germany.
E-mail address: michel.kraemer@igd.fraunhofer.de (M. Krämer).

https://doi.org/10.1016/j.jss.2024.112008
Received 6 September 2023; Received in revised form 16 February 2024; Accepted 18 February 2024
Available online 23 February 2024

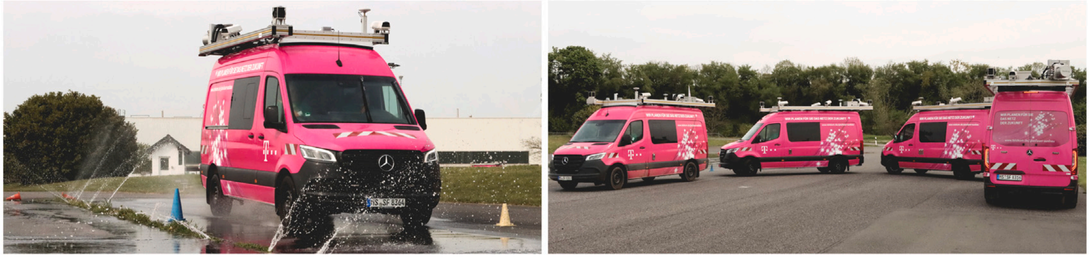


Fig. 1. T-Car: The fleet of mobile mapping cars of the Deutsche Telekom AG.

goal to deploy Fibre to the Home (FTTH) Internet connections to up to 2.5 million households per year (Kessing, 2022a, 2023). The planning processes required for this effort are time-consuming, so automation and ICT support are fundamental. Telekom is working together with partners from industry and research to build a cloud-based data processing and visualization platform, which allows them to speed up the planning processes and to improve the communication between planners, municipalities, and other stakeholders.

In this systems paper, we present an overview of this platform and give details on the parts that we at the Fraunhofer Institute for Computer Graphics Research IGD have been developing and operating together with Telekom over the course of the last six and a half years. We focus on the planning and visualization tool *Fibre3D*, the technical infrastructure behind this tool, and the data processing services needed to prepare the visualized geospatial data.

The main contributions of this systems paper are the architectural overview of the platform as well as the techniques used to automatically preprocess large amounts of 360° panorama images and 3D point clouds to make them suitable for web visualization. We also give details on the interaction techniques implemented in our Fibre3D tool and how they help users speed up the planning process. Finally, we summarize lessons we learned during the development of the platform and give some recommendations for the design of systems like ours. This work addresses researchers who are interested in big data processing and visualization and who face similar requirements and challenges (see Section 1.2). We hope it provides them with useful information and insights into a real-world production use case. We also consider our collaboration with Telekom a good example of how knowledge and technology transfer between research and industry can work in practice.

1.1. Baseline

When we started with the project in 2017, the process of planning broadband networks at Telekom involved many manual steps. Although data about existing networks was mostly available, planners had to collect information about new roll-out areas by hand, perform the planning, and then communicate the plans to the municipalities. This also involved joint on-site inspections, which had to be coordinated in advance. The process from the start of the planning for an individual area to the final approval by the responsible municipality took several months.

To speed up this process and to be able to achieve the goal of connecting 2.5 million households per year to FTTH, Telekom identified automation as one of the key factors. Similarly, the lack of information about new roll-out areas was considered a major obstacle. In consequence, Telekom started to acquire data with laser mobile mapping systems (LMMS) and panorama cameras mounted on top of vehicles. These devices are able to create 3D laser scans and 360° imagery from the surroundings while they are driving. Telekom initially asked external partners to perform this task, but quickly started to build their own vehicle called the *T-Car* (see Fig. 1).

Today, Telekom owns four such cars. Each of them can cover an average distance of 80 km per day in Summer and about 50 km per day in the other seasons. Together with up to three vehicles from external partners, between 2017 and the end of 2023, Telekom was able to collect data from 1,731 planning areas with a total size of 622 TB. The cars drove more than 191,000 km and took 184 million images. In 2023 alone, they drove 54,000 km and covered 401 new planning areas, which resulted in 58.3 million images and a total data size of 191 TB. This process is ongoing in 2024. Apart from that, Telekom makes use of additional information such as aerial imagery and cadastral maps.

All this data is processed by the cloud-based platform we describe in this paper. It performs anonymization and tiling of the collected panoramas and indexes the point clouds to make them suitable for web visualization. At the same time, AI-based data classification is applied to 3D point clouds and aerial images. The results are used to generate potential routes for the trenches needed to put the fibre cables under the ground as well as to determine preliminary positions of fibre optical distribution cabinets.

The results can be viewed in a web-based 3D visualization tool called *Fibre3D*. This tool allows planners to modify the preliminary positions of fibre distribution cabinets so they adhere to the policies that are in place for the given location. The planners also use this tool to make screenshots that they later put into the construction proposals provided to the municipalities (see Fig. 2).

The data acquisition with the T-Car and the automated processing pipeline combined with the 3D web visualization help speed up the planning and approval process tremendously. We demonstrate this in Section 6.

1.2. Challenges

When we started with developing the platform, we were faced with a number of challenges:

C_1 – *Automation.* As mentioned above, automation was identified as one of the key factors to speed up the planning and approval process. The platform therefore needed to offer a pipeline to automatically process the collected data to visualizable results without requiring human interaction. This allows data administrators to start a process or even several at once and let them run overnight or over the weekend, so that planners can commence their task on the next workday.

C_2 – *Big data.* The T-Cars collect up to several gigabytes per kilometre while they are driving through an area. Typical data sets of scanned planning areas have volumes ranging from hundreds of gigabytes up to a few terabytes. Several cars are deployed at the same time, collecting data all over Germany, which results in a large amount of information. The raw data is heterogeneous and unstructured, which means it must be processed and analysed before it can be used. However, the processing results must be available quickly so that planners can start with their work as soon as possible. At the same time, the data should be of high quality (up to date and accurate) and its lineage should be known and trustworthy. Finally, the data has a very high business value for Telekom. The platform therefore needed to address the well-known characteristics of Big Data typically referred to as the Five Big V's *Volume*, *Velocity*, *Variety*, *Veracity*, as well as *Value*.

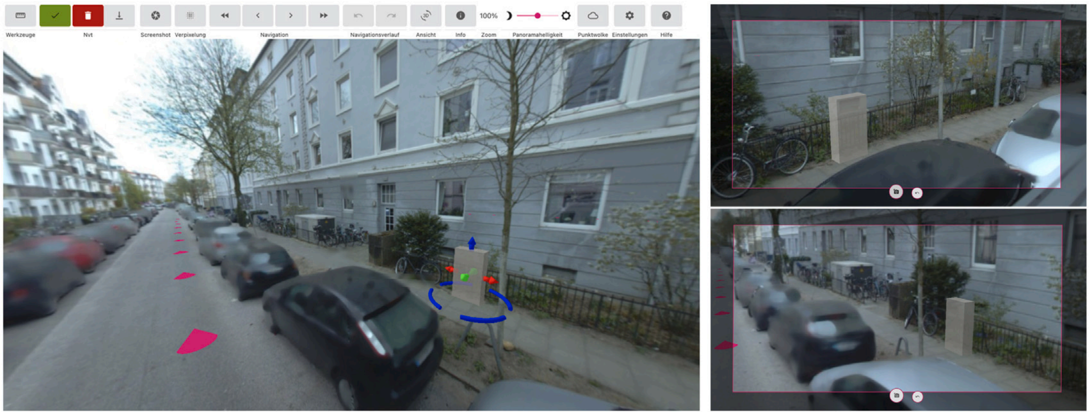


Fig. 2. A planned fibre optical distribution cabinet placed in Fibre3D (left image) and two screenshots to be put into the construction proposal (right images).

C_3 – *Scalability and elasticity.* In order to meet the two aforementioned challenges, the platform needed to be scalable so that it can handle data sets of arbitrary sizes. At the same time, it needed to be elastic so that new resources (virtual machines, memory, block devices, etc.) can be added on demand. We have therefore decided from the beginning to integrate the platform into a cloud infrastructure. Beyond scalability and elasticity, the cloud also allows us to save resources (and hence money) by allocating them as and when necessary and to release them again as soon as they are not needed any more.

C_4 – *Heterogeneous processing services.* Individual data sets are processed by services and algorithms developed by a number of external partners in this project. Various programming languages and technologies are used here (e.g. Java, Kotlin, Python, C++, Rust), and the services have different requirements towards the environment they run in. For example, some processes such as the image anonymization service (Section 4.1) require a Graphics Processing Unit (GPU), while others do not. Some services use large amounts of main memory and others parallelize work across many CPU cores. Our platform addresses this with a scientific workflow management system called Steep, which we developed in earlier work (Krämer and Senner, 2015; Krämer, 2018; Krämer et al., 2021). A scientific workflow is a data-driven model describing how and in which order one or more input data sets should be processed by independent programs (i.e. services) to generate a certain outcome. Steep is able to execute such workflows and schedule the services based on their required capabilities (Krämer, 2021).

C_5 – *Data visualization.* The results produced by our data processing pipeline needed to be visualized in an application that is platform-independent and works on workstations and laptops of a large number of planners. We therefore decided to develop Fibre3D, a web-based visualization and planning application that displays 360° panorama images and point clouds. Since we are dealing with large data volumes, the data needs to be preprocessed and indexed. In addition, the visualization tool should be interactive and intuitive.

C_6 – *Data sharing.* In the context of this project, many users with different roles (planners, data administrators, business and product owners, IT managers, developers, etc.) working at different locations (nationally as well as internationally) needed to be able to access the platform, to use the visualization, and to share information. Running the platform in the cloud helps us meet this requirement. Our system design also contains role-based access control and well-defined security policies.

1.3. Scope

Note that the whole infrastructure comprises a large number of components, tools, and services. Describing every aspect would be too much for one paper. In this work, we therefore only focus on the pipeline that processes panorama imagery and point clouds, as well as on Fibre3D with which the end users perform the planning. After an overview of related work (Section 2), the following sections discuss our system architecture (Section 3) and the panorama and point cloud processing workflow (Section 4). We then give details on our approaches to anonymize collected panorama images (Section 4.1), to tile them (Section 4.2), as well as to index collected 3D point clouds (Section 4.3). After that, we present Fibre3D (Section 5) and show how it is used in a practical use case (Section 6). We finish the paper with lessons learned and a general conclusion (Section 7).

2. Related work

In the following, we summarize existing related work. We focus on systems for cloud-based data processing (Section 2.1), related software architectures for urban planning (Section 2.2), as well as other approaches to visualization in urban planning (Section 2.3).

2.1. Cloud-based data processing

There are several solutions to orchestrate Big Data processing tasks in the cloud. They differ in usage, abstraction level and provided capabilities. For example, frameworks such as Apache Hadoop (Hadoop, 2022a) or Apache Spark (Zaharia et al., 2010) enable parallelization within an application based on MapReduce (Dean and Ghemawat, 2008). They are suitable if a problem is large but can be broken down into many smaller sub-problems that can be worked on independently in parallel. Specialized frameworks such as GeoMesa (Hughes et al., 2015) and GeoTrellis (Azavea, 2022d) build upon Hadoop and Spark and add functions for processing geospatial raster and vector data.

In contrast to these frameworks, distributed Workflow Management Systems (WMS) operate on a higher level and orchestrate the execution of multiple programs (sometimes called tasks or services) based on declarative and user-defined data-driven workflows. WMSs take care of managing computational resources and parallelization, which helps service developers focus on their actual tasks instead of having to think about the technical details of distributed computing.

Some workflow management systems specialize on specific use cases. For example, GEO-WASA (Medeiros et al., 1996), Geo-Opera (Alonso and Hagen, 1997), or LWGeoWfMS (Du and Cheng, 2017) offer predefined services for common tasks in the field of geodata processing. However, when other data needs to be processed, they quickly reach their limits. In this case, general purpose WMSs such as Steep (Fraunhofer, 2022i), Pegasus (Deelman et al., 2015), Argo Workflows (Argo, 2022b), Flyte (Flyte, 2022c) are more suitable.

In previous work, we used a general purpose workflow management system to implement a platform for the processing of large geospatial

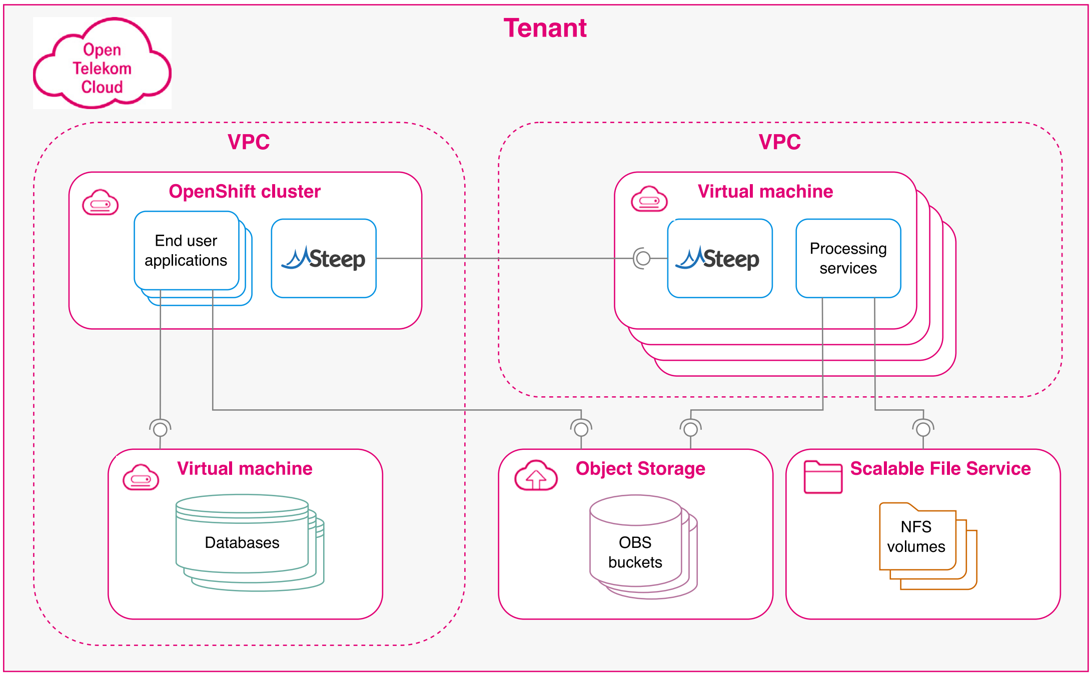


Fig. 3. Overview of the architecture of our cloud-based processing platform.

data in the cloud (Krämer and Senner, 2015). We were able to show that WMSs can be used to create a scalable, modular, and maintainable system. In particular, we could demonstrate that the approach to the orchestration of individual processing services allows independent and distributed teams to collaborate and build a joint software. Based on our good experience and the fact that a general purpose WMS addresses many of the challenges from Section 1.2, the platform we present in this paper employs the WMS Steep, which has been developed by us.

2.2. Software architectures for urban planning

For several years, cloud technologies have been used to improve urban planning processes, in particular in the context of the *Smart City* movement (Ludlow and Khan, 2012). Santana et al. report on key challenges for Smart City systems such as data management, service and device heterogeneity, as well as scalability (Santana et al., 2017). They conclude that dedicated software architectures for these systems are needed. Khan et al. present a software architecture for Big Data analytics supporting planning and decision-making in future Smart Cities (Khan et al., 2013). Their solution is based on well-known cloud technologies, tools, and open standards making their architecture easy to be integrated into existing Smart City environments.

While many Smart City architectures focus on the integration of Internet of Things (IoT) devices, the planning aspect in urban areas has also been explored. Photovoltaics are one technology where planning processes can be improved by utilizing cloud-based software architectures. They provide the necessary data and infrastructure to run large-scale simulations (Bottaccioli et al., 2018) and to visualize planned photovoltaic installations (Adeleke, 2018).

Microservice architectures were identified as a good candidate to implement Smart City systems, which led to the emergence of several reference architectures (Krylovskiy et al., 2015; Del Esposte et al., 2017; de M. Del Esposte et al., 2019). Microservices are independent programs, each of them serving one specific purpose, similar to the processing services controlled by a workflow management system.

In previous works, we successfully utilized the microservice architectural style to implement a Smart City system for secure data exchange (Krämer et al., 2019), as well as a cloud-based Big Data processing platform (Krämer and Senner, 2015; Krämer, 2018). The design decisions that led to the system we present in this paper are heavily based on the experience from these earlier works.

2.3. Visualization in urban planning

Visual data such as panorama imagery and point clouds play an important role in the context of urban planning. Freely available panoramas provided through services such as Google Street View (Google, 2022j) enable a variety of applications ranging from building instance classification (Kang et al., 2018), over assessing solar distribution in urban canyons or on streets in order to identify potential for photovoltaics (Liu et al., 2019; Li and Ratti, 2019), up to understanding the effect that street greenery has on physical activity (Lu, 2019). Availability of such data ultimately makes it possible to perform survey tasks remotely that traditionally required on-site visits (Berland and Lange, 2017). For situations where image data has insufficient information, point cloud data is often used, for example for urban reconstruction (Wang et al., 2018) or segmentation of urban scenes (Babahajiani et al., 2017). Web-based tools such as Potree (Schütz, 2016) or CesiumJS (Cesium, 2021b) make access to large-scale point cloud data possible, similarly to what Google Street View enabled for panorama images. Fast visualization of large-scale point cloud data typically requires substantial preprocessing, provided by tools such as PotreeConverter (Schütz et al., 2020) or Schwarzwald (Bormann and Krämer, 2020).

2.4. Summary

While the individual building blocks in terms of processing frameworks, urban planning software architectures, and visualization tools already exist, a system that combines these items and addresses all challenges we describe in Section 1.2 is not available off the shelf. Also, during the development of our system, we also faced many other sub-problems for which we could not find existing solutions. In the following sections, we describe the software architecture of our system and give details on these sub-problems and how we addressed them.

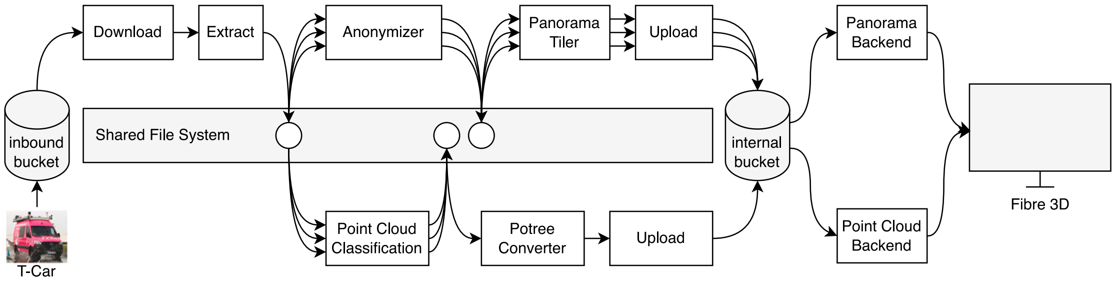


Fig. 4. Data preparation for a planning area. Data is recorded by the T-Car and uploaded to an inbound bucket. Afterwards, several services process it. If multiple arrows go into a service, several driving sections of the area are computed simultaneously on different VMs. The results are stored in an internal bucket and can be visualized in Fibre3D.

3. Architecture overview

An overview of the architecture of our system is given in Fig. 3. The system is embedded in a larger spatial data infrastructure (SDI) and runs in the Open Telekom Cloud (OTC).[1] The OTC provides an Infrastructure as a Service (IaaS) layer and allows customers to manage virtual machines, virtual networks, and object storage.

For reasons of isolation and security, the system is divided into two virtual private clouds (VPCs), one hosting an OpenShift cluster and various databases, and another one with dynamically created virtual machines performing all computations. In addition, there is an object storage service (OBS) that contains the data collected by the T-Cars as well as the processing results (see also challenge C_2). For fast data sharing between virtual machines, there is a shared file system (called *Scalable File Service* in the OTC) that can be mounted via NFS (C_6).

End-user applications such as our visualization software *Fibre3D* (C_5) run in the OpenShift cluster. They can use the databases to store application-specific information. All processing services are controlled by our workflow management system *Steep* (C_1). It starts dedicated virtual machines in the cloud, runs the services on them, and monitors their execution (C_3). Steep also manages available resources. Its capability-based scheduler creates VMs that match the requirements requested by the processing services with regard to hardware and software (Krämer, 2021) (C_4). For example, if a certain service requires a GPU to speed up processing, Steep creates a VM with such a device and deploys a VM image with the correct operating system and drivers to it. Likewise, if a service needs to perform a CPU-intensive task but does not need a GPU or a large amount of memory, Steep creates a smaller, less expensive VM. This allows us to make the best possible use of available resources and to reduce cloud costs.

Overall, T-Car data is processed as follows in our system:

1. The data is uploaded to a special bucket in the object storage called the *inbound bucket*.
2. One of the data administrators uses an end-user application called *FibreManager* (not part of this paper) to start one or more of a set of pre-defined workflows (see below).
3. Steep executes the workflows. For this, it creates virtual machines, runs the services on them and monitors their execution.
4. The services download data from the object storage, process it, and then upload the results back to the object storage into a bucket called the *internal bucket*.
5. The results can finally be viewed in end-user applications such as Fibre3D.

Basically, there are two central workflows that are executed with this platform:

Workflow 1: Panorama and point cloud processing. The panorama images collected by the T-Car may contain sensitive information such as faces of pedestrians or car licence plates. They are also too large for interactive web visualization. This workflow therefore applies image anonymization and tiling through services called *Anonymizer* and *Panorama Tiler*, respectively. At the same time, the point clouds are processed by a *Point Cloud Classification* service, which uses a convolutional neural network (CNN) developed by the Fraunhofer Institute for Physical Measurement Techniques IPM to identify street coverings, trees, rails, manholes, and other objects relevant to the planning process (Reiterer et al., 2020). In order to make the point clouds suitable for web visualization, a service called *PotreeConverter* (Schütz et al., 2020) creates a spatial acceleration structure.

Workflow 2: Trenches and fibre cabinets. The second workflow performs an AI-based classification of aerial imagery and combines the results with those from the *Point Cloud Classification* service from workflow 1 in order to generate potential routes for the trenches needed to put the fibre cables under the ground. Based on this, planners then use an external system not part of our platform, to create preliminary positions of fibre optical distribution cabinets. These positions are saved in a database in our platform.

As mentioned above, in this paper, we only focus on the visualization-centric preprocessing of panorama imagery and point clouds, namely workflow 1. Workflow 2 is beyond the scope of this paper. Nevertheless, the preliminary positions of fibre distribution cabinets play a role in the 3D web visualization later (see Section 6).

The following section describes workflow 1 in more detail on a technical level and shows how the different services are connected.

4. Panorama and point cloud processing workflow

Fig. 4 illustrates the panorama and point cloud processing workflow. White boxes represent the individual processing services.

The data collected by the T-Car for a certain planning area is uploaded as a ZIP archive to the *inbound bucket* in the object storage service. The first steps are to *Download* this archive and to *Extract* its contents to the shared file system.

While recording, the T-Car splits the data into so-called *driving sections* at a given interval. Each section is stored in a separate directory within the uploaded archive. This can be utilized by Steep to parallelize the data processing in the next step, where it starts multiple VMs for the *Anonymizer* and *Point Cloud Classification* service.

The *Anonymizer* employs a Convolutional Neural Network (CNN) to identify people and cars in panorama images and to blur them to make the panoramas compliant with data privacy policies. At the same time, the *Point Cloud Classification* service analyses the point clouds and uses another CNN to search for various object classes such as different types of surfaces (asphalt, grass, etc.), railway lines, manhole covers, curbstones, trees, cars, and building facades. The service exports information about these objects as a GeoJSON file, which is later used to plan potential trenches for fibre lines in workflow 2. It also

[1] https://open-telekom-cloud.com/

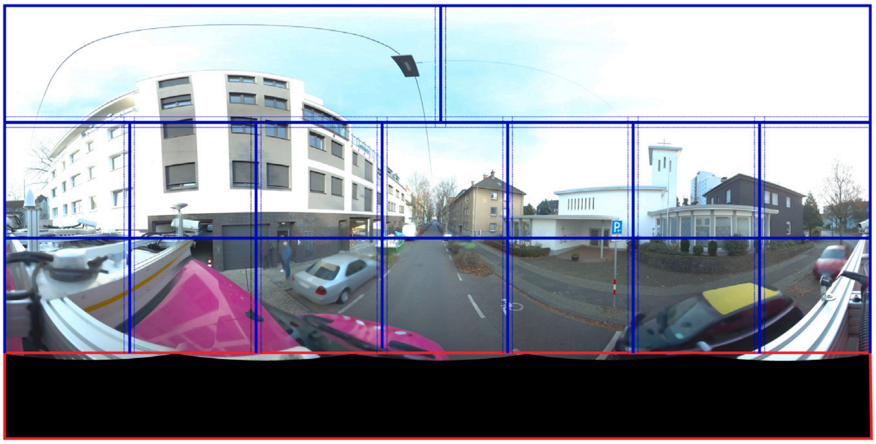


Fig. 5. Tiling scheme of the Anonymizer. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

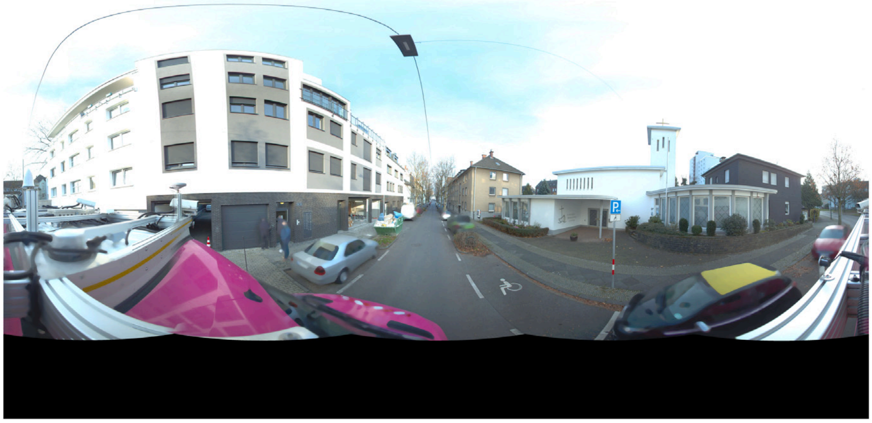


Fig. 6. Example of an anonymized Image.

reduces the resolution of the point clouds and combines them with the panorama images to create colourized points (Reiterer et al., 2020).

Both the *Anonymizer* and the *Point Cloud Classification* service require a GPU for execution. This distinguishes them from other services such as the *Panorama Tiler* or the *PotreeConverter*, which operate on the CPU. Since VMs with GPUs are more expensive than regular VMs, Steep instructs the *Anonymizer* and the *Point Cloud Classification* service to write their results to the shared file system and then shuts the VMs down.

After the *Anonymizer* has finished, Steep starts the *Panorama Tiler*, again distributed across different VMs, one for each driving section. The service splits a 360° panorama image into several small tiles. These tiles are needed by *Fibre3D* for visualization (see Sections 5 and 4.2). The results are uploaded to the *internal bucket* on the object storage, from which Fibre3D can access them through the *Panorama Backend* service.

In parallel to this, the *PotreeConverter* generates a Level of Detail (LoD) structure for the point clouds. In contrast to the previous services, the *PotreeConverter* is only started once per planning area and processes all driving sections at the same time. Because it should be possible to move freely through the entire planning area in Fibre3D, this requires an LoD structure that covers the entire data set and not individual driving sections. The generated LoD structure is also uploaded to the *internal bucket*.

Note that, due to the data-driven nature of scientific workflows, Steep can manage the branch of the pipeline containing the *Anonymizer* and the *Panorama Tiler* independently of the branch with the *Point Cloud Classification* service and the *PotreeConverter*. This means that as soon as the first driving sections have been processed by the *Anonymizer*, the first instances of the *Panorama Tiler* can be started, regardless of how much of the point cloud data has been processed and vice versa.

Likewise, Steep can manage multiple workflows at the same time. Several driving areas can therefore be processed in parallel. Steep allocates the required resources and distributes the work automatically (Krämer, 2021).

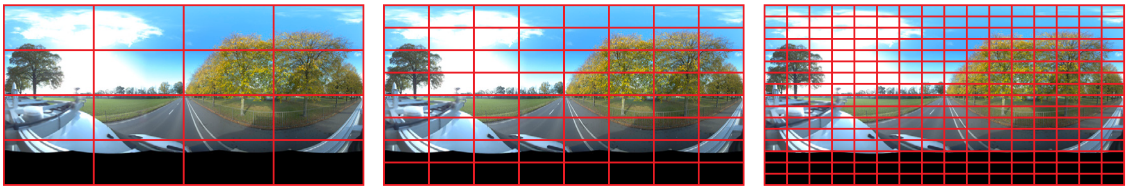
In the following sections, we focus on the internal details of the *Anonymizer* (Section 4.1), the *Panorama Tiler* (Section 4.2), and the *PotreeConverter* (Section 4.3). For details on the *Point Cloud Classification* service, we refer to the work of Reiterer et al. (2020).

4.1. Image anonymization

Personal information in panorama images needs to be erased or at least made unrecognizable to meet data privacy policies. In the processing pipeline, this task is performed by the *Anonymizer* service, which first uses a CNN to search for relevant objects in the images and then blurs the matching pixels. The Anonymizer consists of the four components *Cropping*, *Tiling*, *Object segmentation*, and *Blurring*:

1. Cropping. Due to technical reasons, each panorama image taken by the T-Car contains a black area at the bottom (see Fig. 5, red box). To avoid wasting time during object recognition, this area is cropped out first.

2. Tiling. A panorama image has a resolution of 8192 × 4096 pixels. In order to ensure a high recognition rate even for objects that are very small or that are in the background (far away from the camera),



Level 0 Level 1 Level 2

Fig. 7. Panorama tiling scheme: Images are split into 16, 64, and 256 tiles in level 0, 1, and 2 respectively.

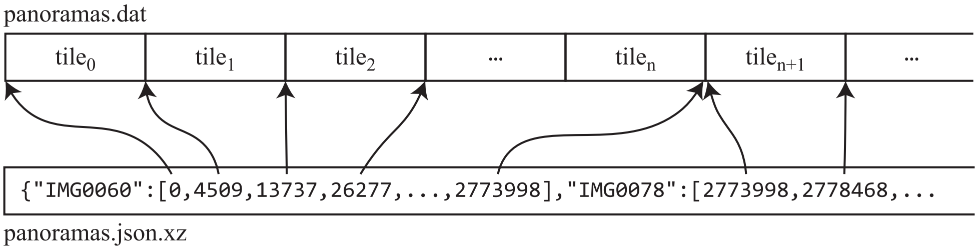


Fig. 8. File layout of our panorama tile format.

the panorama image is split into smaller tiles. We have implemented an irregular tiling scheme with smaller tiles in areas likely to contain many objects and larger tiles in other areas. Currently, each image is divided into three rows, of which the top row is divided into only two tiles and the two bottom rows into seven tiles each. This scheme is a good compromise between accuracy and performance. In addition, an overlap of 100 pixels is added between two tiles. This is important so that small objects on the edges can also be reliably detected. The individual tiles that are generated from a panorama image are shown as blue boxes in Fig. 5. The overlaps are marked with dashed thin lines.

3. Object segmentation. The core of the Anonymizer is the object recognition. We are using the CNN-based object recognizer YOLO (Wang et al., 2022) with pre-trained weights (Wong, 2022) to search for persons, cars, motorbikes, buses, and trucks in the images. Since YOLO was designed for real-time applications, it works very fast and makes data preparation cost-effective.

4. Blurring. The final step in the Anonymizer is to blur the detected objects. First, the entire panorama image is copied and completely blurred. The detected objects of the respective tiles are then transferred to the coordinates of the entire image. Finally, these pixels are cut out of the blurred image and pasted onto the original panorama image. The result is the anonymized panorama image, which can be seen as an example in Fig. 6.

It can be added that all steps except for the object recognition take place thread-parallelized in the CPU. The object recognition is parallelized in the GPU.

4.2. Image tiling

The 360° panorama images generated by the T-Car are stored as JPEG files. They have a resolution of 8192 × 4096 pixels with RGB 8-bit channels and are typically between 3 and 5 megabytes in size. Uncompressed, a single image occupies $8192 \times 4096 \times 3 = 96\,\text{MiB}$ of memory. Displaying such a large image in a web browser can significantly affect the user experience: first, the JPEG needs to be transferred, which can take several seconds (depending on the Internet connection), then it needs to be uncompressed and pushed onto the graphics card. Depending on the processing power of the workstation and, in particular, the graphics card, the user might experience lags in displaying the scene and, hence, interaction might become slow and choppy.

To avoid these issues, the *Panorama Tiler* service prepares the images by creating an image pyramid in different resolutions and by splitting the individual levels of detail into smaller image tiles. Fig. 7 shows the tiling scheme. For each image, three levels of detail are generated. In level 0, the image is split into $4 \times 4 = 16$ tiles. For levels 1 and 2, the Panorama Tiler creates 64 and 256 tiles, respectively. Each tile has a fixed resolution of 512 × 256 pixels. Individual tile image files are typically only a few kilobytes in size.

Images of this size can later be loaded into the browser in a streaming manner (see Section 5). Fibre3D only needs to load those files that are currently visible and can request more images on demand. Typically, only the low-resolution images from level 0 need to be displayed unless the user zooms in with the mouse wheel, in which case the higher-resolution images from level 1 and 2 are loaded. This improves responsiveness of the application and saves a lot of bandwidth at the same time.

Nevertheless, tiling has the drawback that the number of individual image files grows by orders of magnitude. For each original image, the Panorama Tiler creates 336 smaller images. A typical planning area with hundreds or thousands of panoramas can quickly result in millions of very small files. Storing such an amount of files in an S3-based object store has, in our experience, a huge impact on performance and costs. Every file has to be uploaded separately in a new HTTP request, and transmission latencies can pile up very quickly, so uploading can take several hours even if the millions of files only have a total size of a few gigabytes. Furthermore, whenever these files have to be touched again later (for example, to move them to a different location, or simply to delete them), the whole process again takes very long.

For these reasons, we implemented a simple archiving format, which we call *Panorama Bigfile*. Each Bigfile consists of an archive called `panoramas.dat`, which contains the individual image tiles, and an index file called `panoramas.json.xz` containing the byte ranges of the tiles in the `panoramas.dat` file. Fig. 8 shows the layout of these two files. For each panorama, the JSON file contains an array of start positions of image tiles. The end position of tile i is the start position of tile $i + 1$. The last entry in the array is the end position of the final tile of the panorama. The entries are sorted by zoom level and then by their index on the x and y axis.

Image tiles can later be accessed by obtaining the byte positions from the index file and issuing an HTTP range request to the object storage. We created a translator service (*Panorama backend* in Fig. 4) that reads the index file and maps a panorama name, a zoom level, and a respective x and y index to a byte range. This helps the web visualization to fetch the right image tiles and avoids transferring the index file to the client.

Using this approach has saved us time and money. Before we implemented the Bigfile format, as mentioned above, we had to upload millions of small image files for each planning area. This took many hours, and since each request was charged individually, also cost quite an amount of money (several thousands of Euros per year). The Bigfile format consists of only two files and uploading them costs just a few cents per area. Also, data upload can make use of the full network bandwidth, which reduces upload times to a few minutes on average.

4.2.1. Alternatives

Note that we store the index file using *XZ* compression (Tukaani, 2009). Compression is useful here because a JSON-based index file can become very large and downloading it from the object store can take our translator service several seconds. We tested various compression algorithms (GZIP, LZ4, Brotli, Snappy, and XZ) and found that XZ provides the best compromise between the time it takes to download the compressed file and the time it takes to extract it. This is supported by the findings of Shurtz who compared various compression algorithms with binary object formats to store JSON files on AWS S3 (Shurtz, 2019). With this, we were able to reduce the time it takes to access the index file to a few hundred milliseconds. Once an index is uncompressed, our translator service caches it in main memory.

Regarding the format of the archive file `panoramas.dat`, we also tested several alternatives. For example, at the end of each ZIP archive there is a dictionary that can be used to extract individual files without reading the whole ZIP stream. However, the format is rather complex and very hard to handle if the archive is stored on an object store (where you only have access via HTTP) instead of a local file system with random access. TAR archives, on the other hand, are easier to handle, but they are not suitable to store a large amount of very small files: TAR archives are structured in 512-byte blocks and individual entries are aligned to block boundaries. If you store, for example, ten files of a size of 1100 bytes, the tar archive will have a total size of 15,360 bytes, which is an overhead of about 40%. Since our panorama tiles are typically just a few kilobytes large, this overhead is unsuitable for our use case.

4.3. Point cloud indexing

To facilitate realistic fibre planning, users need interactive access to the point cloud to quickly judge spatial relationships. Both the visualization of and interaction with the point cloud require fast access to specific parts. We therefore preprocess the point cloud to create a spatial index. There are several tools available for generating an index suitable for point cloud visualization, such as *Schwarzwald*, *PotreeConverter* or *Entwine* (Bormann and Krämer, 2020; Schütz et al., 2020; Hobu, 2020). Initially, we used Schwarzwald to generate an indexed point cloud in the *3D Tiles* (Cesium, 2018) format, because the first version of Fibre3D was built on top of *CesiumJS*, which requires 3D Tiles. However, we realized that 3D Tiles in its current form is unsuitable for handling large point clouds, as the significant number of files that an indexing process for 3D Tiles generates becomes a bottleneck during copy operations in the cloud. This situation is similar to that of the panorama tiles. Additionally, we found CesiumJS to have insufficient support for point cloud visualization and interaction.

We therefore decided to switch to the well-known point cloud visualization framework *Potree* and use the latest version 2.0 of its associated indexing tool *PotreeConverter*, which stores all nodes of the point cloud index in two files (one for the data, one for the hierarchy) and thus drastically reduces the time required for copy operations. In terms of preprocessing time, both Schwarzwald and PotreeConverter are similar (see for example the findings of Kocon and Bormann (2021)), as they make extensive use of parallel processing. Both tools generate the same index structure, a multi-resolution octree where different levels represent different resolutions, which enables efficient visualization with various levels-of-detail. Since PotreeConverter stores point data in a single file, access to individual parts of the point cloud is realized through HTTP range requests (Schütz et al., 2020), similar to what we do for the panorama images.

5. 3D visualization

To enable interactive planning of fibre lines in a 3D environment, our architecture has to be able to display both panorama images and point clouds in a joint visualization. The panorama images give the visual context, while the point cloud is used to provide spatial context. This allows users to perform geometrical tasks such as measuring distances, angles, areas, or placing objects in the 3D environment, adhering to intersections with the scene geometry (see also Section 6). To meet these requirements, we developed an interactive web-application called *Fibre3D*. We display panorama images directly in the point cloud by projecting them onto a sphere located at the georeferenced coordinate of the panorama image within the scene.

Fibre3D supports two movement modes in 3D space: the *panorama mode* and the *free movement mode*. In panorama mode, the user can jump from one panorama image to another, thus moving between the fixed panorama positions akin to applications such as Google's *Street View* (Google, 2022j). In the free movement mode, only the point cloud is displayed, and the user can navigate freely in 3D space. While the panorama images provide a more detailed visual representation of the real world, utilizing the point cloud this way allows users to explore regions that are not well covered by panorama images. For example, they can move the camera close to a wall on a sidewalk or even observe the area from a bird's-eye view.

In the following section, we show how the results from the data processing pipeline and Fibre3D are used in practice to perform fibre planning.

6. Practical use case

After the data has been processed by workflow 1 of our pipeline, suitable routes for fibre optic lines have been calculated by workflow 2, and preliminary possible locations for fibre distribution cabinets have been determined (see Section 3), the users can finish the fibre planning process in Fibre3D. Here, they basically check if the locations for the distribution cabinets are correct and adhere to local regulations, create screenshots from various view angles, and export those into a PDF file that can be sent to the municipality and other stakeholders to support the decision-making process.

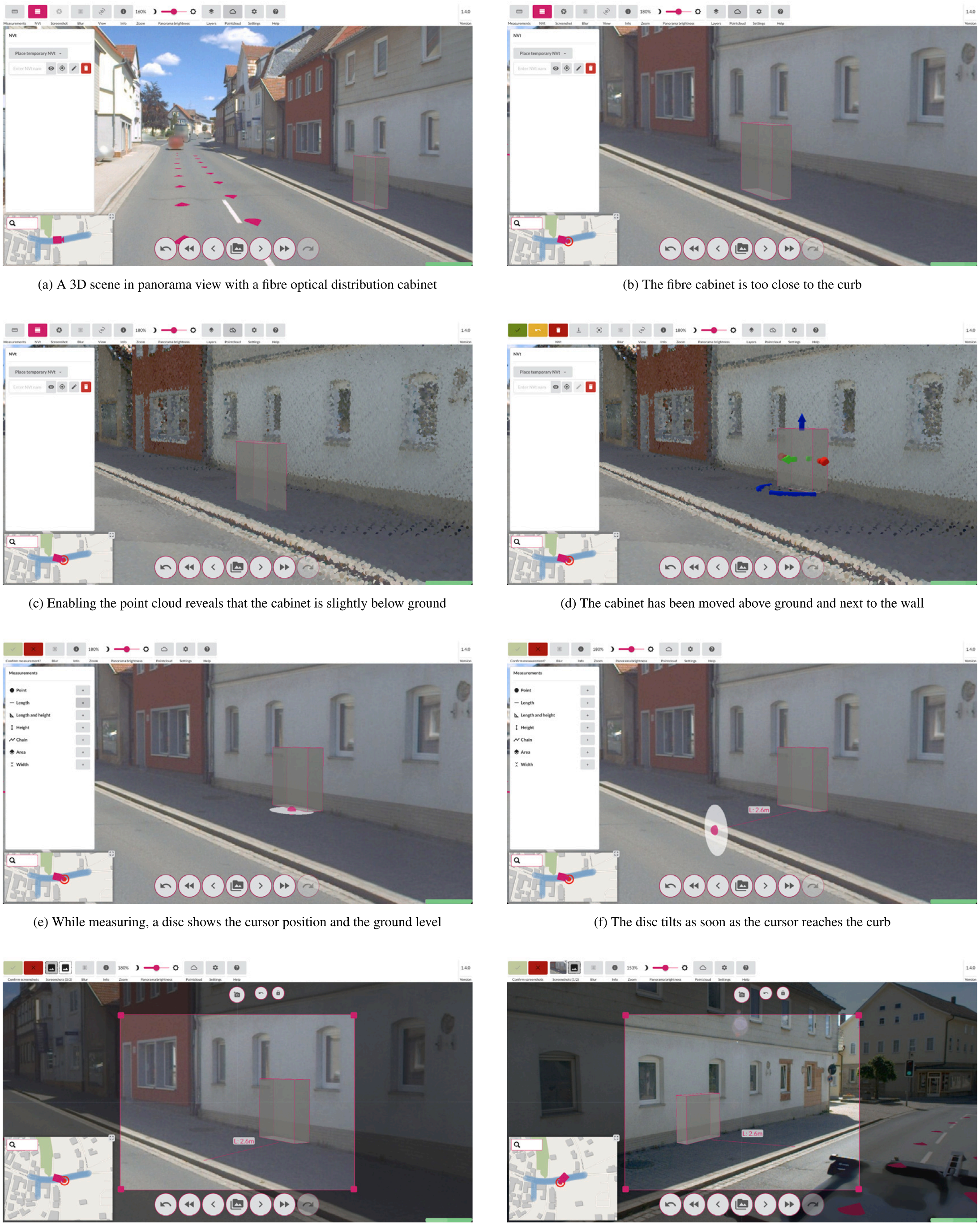
A user opens Fibre3D and visits each distribution cabinet one after the other. Fig. 9 shows the steps of a typical workflow. First, the user checks the location in the panorama view (Figs. 9(a) and 9(b)), but since it is rather hard to judge if the cabinet is actually placed correctly in a 2D image, the user enables the 3D point cloud (Fig. 9(c)). In this example, the cabinet is too close to the curb and also too low (the bottom of the cabinet is slightly below the ground), which becomes clearly visible in the point cloud.

To fix this, the user clicks on the cabinet to make the controls appear, to move the cabinet closer to the wall, and to raise it slightly. The user checks the new location again (Fig. 9(d)).

Next, the user employs the measure tool to draw a line into the 3D scene from the cabinet to the curb and to check if the space between cabinet and curb is wide enough (Fig. 9(e)). While drawing, Fibre3D shows a disc that tells the user if the surface under the mouse cursor is flat or not. When the user reaches the curb, the disc will be tilted (Fig. 9(f)). This feature uses the 3D point cloud in the background. It works even when the point cloud is not visible and helps the user operate in the 3D scene and make accurate measurements.

When satisfied with the placement of the cabinet, the user opens the screenshot tool to take two images from different view angles (Figs. 9(g) and 9(h)). These screenshots will then be transferred in the background to another component in our platform, which generates the PDF that can be sent to the municipality and decision makers.

Fibre3D contains many more features such as additional tools to measure areas, heights, and widths, which the users from time to time use to compare other (possibly older) digital plans with the data the



(a) A 3D scene in panorama view with a fibre optical distribution cabinet

(b) The fibre cabinet is too close to the curb

(c) Enabling the point cloud reveals that the cabinet is slightly below ground

(d) The cabinet has been moved above ground and next to the wall

(e) While measuring, a disc shows the cursor position and the ground level

(f) The disc tilts as soon as the cursor reaches the curb

(g) Two screenshots of the planned location of the cabinet are taken

(h) The screenshots also show that there is enough room between cabinet and curb

Fig. 9. A typical planning use case with Fibre3D: The user fixes the automatically calculated location of a fibre optical distribution cabinet. They measure the distance between cabinet and curb and take two screenshots from different view angles. These screenshots will later be inserted into the construction proposal.

T-Car has collected (i.e. the ground truth, closest to reality). Fibre3D also offers a navigation mode that allows the users to move freely through the 3D point cloud. The tool can also be controlled with the keyboard. For example, the keys W, A, S, and D can be used to navigate just like it is known from computer games. Fibre3D is also customizable: Users can assign different shortcuts or change the visualization. For the point cloud, among other parameters, the tool allows the users to change the point density and size, which helps them adjust the performance to their hardware.

7. Summary

In this paper, we presented a cloud-based platform that supports the roll-out of high-speed fibre networks in Germany. We specifically focused on the pipeline that processes visualization-centric data including panorama images and 3D point clouds. This data is collected with special mobile mapping vehicles operated by Deutsche Telekom (T-Cars) and external partners with the aim to create a virtual representation of areas where fibre network should be deployed. Our web-based 3D visualization allows expert users to display the data and to perform planning in virtual reality.

In this section, we highlight some lessons we learned during the development of the platform and finish the paper with a general conclusion.

7.1. Lessons learned

The following list summarizes insights we have gathered and which we believe may be useful to other researchers facing similar requirements as we did. We specifically focus on the challenges from Section 1.2 (labelled C_1 to C_6) and give some recommendations for the design of systems like ours.

Automation and independent microservices. Based on the good experience from our previous work, we integrated a scientific workflow management system into our architecture to control and automate how the collected data should be processed by the various services in our pipeline (see Section 4) (C_1). We believe that this is the right approach, particularly since our platform includes independent microservices developed by different project partners (C_4). These services use various technologies and have independent lifecycles. Also, there is no shared code base and no common software framework where the services should or can be integrated. What we would like to underline here is that controlling the service execution and the data flow between them through a workflow management system is a powerful solution. We recommend considering the microservice architectural style and the scientific workflow model during design of any Big Data processing system.

Big data, elasticity, and scalability. In order to be able to store and process large amounts of data (C_2), the cloud plays a key role in our architecture. Even though using the cloud was not yet common practice in the geospatial community when we started the project in 2017, our decision was the right one. The possibility to elastically (i.e. on demand) acquire new storage and compute resources and to release them when they are not needed any more enables our system to adapt automatically to the workload and has helped us save time and money (C_3). It has also enabled us to share data between thousands of users across the company (C_6). Regarding scalability, we would like to highlight our approach to managing the millions of files that are created during data processing. In particular, our Bigfile format for image tiles (Section 4.2) has helped save time and money during data upload and transfer between environments. The same applies to the PotreeConverter, which we use for point cloud indexing and which also only creates two files (see Section 4.3). The fact that such large numbers of files can be a problem was something we only learned during the project. You typically do not work with the same amount of data in your development environment and problems only become evident when your software goes to production. We recommend to always keep issues like this in mind and to consider them as early as possible, preferably already during the design of the software architecture.

Parallelization and data models. Along the same lines, we believe that our approach to divide the collected data into driving areas and these areas in turn into driving sections (see Section 4) was the right choice. It has helped us to parallelize data processing and to scale horizontally across multiple virtual machines (C_3), which has in turn saved a lot of time. The basic data acquisition, storage, and processing strategy should be one of the first things to decide on during the design of a system and applying the divide-and-conquer strategy should be considered.

Beware of the vendor lock-in. Although, we believe that the cloud is key to our architecture, we would like to point out that there are some drawbacks involved. Even though the cloud has become commodity in recent years, there are no real standards that would allow you to migrate between different cloud environments. Every cloud vendor has their own services and interfaces, and tailoring a software for one environment leads to a vendor lock-in. Migrating to another environment can be very costly, and in the worst case, it might force you to rebuild your entire architecture. Common standards are an important topic for research, but due to the lack thereof, we recommend keeping this in mind when deciding for a certain cloud environment and/or whether the cloud should be used at all (C_2, C_3, C_6).

No standardized data visualization formats. As described in Section 4.3, we initially used CesiumJS to build our web-based visualization app (C_5) but later migrated to Potree. While CesiumJS is suitable to visualize generic 3D geospatial data, Potree is much more flexible and powerful when it comes to point cloud rendering. For applications dealing with point clouds we would therefore always recommend Potree. Also, similar to what we described above, the PotreeConverter only creates two files instead of thousands of small ones, but converting our data from the 3D tiles format used in CesiumJS to the Potree format was a migration that we did not expect. Our general recommendation is therefore to always be aware that migrations and data conversions might become necessary at some point in the project and to account for them early on.

8. Conclusion

As described in Section 1, the current goal of Telekom is to provide up to 2.5 million households with Fibre to the Home (FTTH) Internet per year. For this, between 2017 and the end of 2023, they have collected data for 1,731 planning areas with a total size of 622 TB. This process will continue in the future, and without a cloud-based platform like the one we described in this paper, processing the sheer amount of data in a reasonable time, let alone just storing it, would be barely manageable.

Our platform addresses all the challenges defined in Section 1.2. It offers *automation* by employing our scientific workflow management system Steep, which acquires cloud resources in a *scalable and elastic* manner (i.e. automatic, on demand, and well-proportioned), deploys *heterogeneous processing services* to the created virtual machines, and controls their execution to process *Big data* sets. Our web application Fibre3D allows users to *visualize* the data in 3D and to perform interactive planning in virtual reality. Being cloud-based, our platform also enables *data sharing*. Various applications and services can access the data at a central location and can always rely on it being up-to-date.

One major takeaway from our collaboration with Telekom is that it is indeed possible to have a knowledge transfer between industry and research, to develop practical solutions for real-world challenges, and, at the same time, to gain insights and conduct research. Besides the results presented in this paper, over the last couple of years, we were able to utilize the knowledge acquired from the collaboration and to publish works specifically in the areas of workflow management (Krämer, 2021; Krämer et al., 2021), artificial intelligence (Kocon et al., 2022; Würz et al., 2023), and point cloud processing (Kocon and Bormann, 2021; Bormann and Krämer, 2020).

The project also has an impact on the practical work of domain experts. The 3D visualization enables planners and decision makers from municipalities to avoid time-consuming on-site inspections as well as to communicate faster and more precisely, possibly including several feedback cycles and discussing alternatives. This is leveraged not only through the screenshots from Fibre3D that are included in the construction proposals but also by the possibility to share URLs to the 3D visualization with external stakeholders or to quickly present the plans with a laptop and a presenter in front of decision makers. Nevertheless, the new process requires planners to familiarize themselves with working in 3D, which is something many of them have never done before. We have closely collaborated with them to make the user interface as intuitive as possible but placing a 3D object on a 2D display can be cumbersome. However, the possibility to view the location of a cabinet from different angles and to turn on the 3D point cloud at any time helps a lot in this respect.

Finally, our research and the collaboration with Telekom has a high impact on society. Compared to six and a half years ago, before we started this project, the developed platform has tremendously sped up the overall planning process. Planning processes that previously took several months can now be completed within a few weeks. Planners from Telekom have reported that, on average, the processes are about 75% faster than before. Up to now, they have planned FTTH connections for about 10 million households with this platform. They already implemented more than 8 million of them, 2.5 million thereof in 2023 alone (Sanchez, 2023; Kessing, 2022b, 2023).

Requirements in terms of how many households can be connected to FTTH per year and, thus, how much data needs to be collected and processed in a short time are ever-increasing. In the future, we will therefore work on further improving the performance of data processing by optimizing algorithms and increasing scalability. Current numbers show that Telekom is on a good way to have more than 10 million households connected by the end of 2024.

CRediT authorship contribution statement

Michel Krämer: Conceptualization, Software, Writing – original draft, Project administration. **Pascal Bormann:** Software, Visualization, Writing – original draft, Conceptualization. **Hendrik M. Würz:** Conceptualization, Software, Writing – original draft. **Kevin Kocon:** Software, Writing – original draft. **Tobias Frechen:** Conceptualization, Project administration. **Jonas Schmid:** Data curation, Project administration.

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

The data that has been used is confidential.

References

Adeleke, A.K., 2018. Web-based GIS modelling of building-integrated solar photovoltaic system for the City of Cape Town (Ph.D. thesis). University of Cape Town.

Alonso, G., Hagen, C., 1997. Geo-opera: Workflow concepts for spatial processes. In: International Symposium on Spatial Databases. Springer, pp. 238–258. http://dx.doi.org/10.1007/3-540-63238-7_33.

Argo, 2022b. Argo workflows. https://argoproj.github.io/workflows/, Last visited: 2022-05-04.

Azavea, 2022d. GeoTrellis. https://geotrellis.io/, Last visited: 2022-04-13.

Babahajiani, P., Fan, L., Kämäräinen, J.-K., Gabbouj, M., 2017. Urban 3D segmentation and modelling from street view images and LiDAR point clouds. Mach. Vis. Appl. 28 (7), 679–694.

Berland, A., Lange, D.A., 2017. Google street view shows promise for virtual street tree surveys. Urban Forestry Urban Greening 21, 11–15.

BMDV, 2021a. Bundesministeriums für Digitales und Verkehr. Richtlinie “Förderung zur Unterstützung des Gigabitausbaus der Telekommunikationsnetze in der Bundesrepublik Deutschland”. https://bmdv.bund.de/goto?id=481494, Last visited: 2022-03-15.

Bormann, P., Krämer, M., 2020. A system for fast and scalable point cloud indexing using task parallelism. In: Biasotti, S., Pintus, R., Berretti, S. (Eds.), Smart Tools and Apps for Graphics - Eurographics Italian Chapter Conference. The Eurographics Association, pp. 153–162. http://dx.doi.org/10.2312/stag.20201250.

Bottaccioli, L., Patti, E., Macii, E., Acquaviva, A., 2018. GIS-based software infrastructure to model PV generation in fine-grained spatio-temporal domain. IEEE Syst. J. 12 (3), 2832–2841. http://dx.doi.org/10.1109/JSYST.2017.2726350.

Cesium, 2018. CesiumGS/3d-tiles: Specification for streaming massive heterogeneous 3D geospatial datasets. https://github.com/CesiumGS/3d-tiles, Last visited: 2024-02-26.

Cesium, 2021b. Cesium GS, Inc. CesiumJS. https://cesium.com/platform/cesiumjs, Last visited: 2021-07-19.

de M. Del Esposte, A., Santana, E.F., Kanashiro, L., Costa, F.M., Braghetto, K.R., Lago, N., Kon, F., 2019. Design and evaluation of a scalable smart city software platform with large-scale simulations. Future Gener. Comput. Syst. 93, 427–441. http://dx.doi.org/10.1016/j.future.2018.10.026, URL https://www.sciencedirect.com/science/article/pii/S0167739X18307301.

Dean, J., Ghemawat, S., 2008. MapReduce: simplified data processing on large clusters. Commun. ACM 51 (1), 107–113.

Deelman, E., Vahi, K., Juve, G., Rynge, M., Callaghan, S., Maechling, P.J., Mayani, R., Chen, W., Ferreira da Silva, R., Livny, M., Wenger, K., 2015. Pegasus: a workflow management system for science automation. Future Gener. Comput. Syst. 46, 17–35. http://dx.doi.org/10.1016/j.future.2014.10.008.

Del Esposte, A.M., Kon, F., M. Costa, F., Lago, N., 2017. InterSCity: A scalable microservice-based open source platform for smart cities. In: Proceedings of the 6th International Conference on Smart Cities and Green ICT Systems. In: SMARTGREENS 2017, SCITEPRESS - Science and Technology Publications, Lda, Setubal, PRT, pp. 35–46. http://dx.doi.org/10.5220/0006306200350046.

Du, Y., Cheng, P., 2017. LWGeoWfMS: A lightweight geo-workflow management system. In: IOP Conference Series: Earth and Environmental Science. Vol. 59, (1), IOP Publishing, 012070. http://dx.doi.org/10.1088/1755-1315/59/1/012070.

Flyte, 2022c. Flyte. https://flyte.org/, Last visited: 2022-05-04.

Fraunhofer, 2022i. Steep workflow management system. https://steep-wms.github.io/, Last visited: 2022-05-04.

Google, 2022j. Street View. https://www.google.com/streetview/, Last visited: 2022-04-27.

Hadoop, 2022a. Apache Hadoop. https://hadoop.apache.org, Last visited: 2022-04-13.

Hobu, 2020. Entwine. https://entwine.io/, Last visited: 2022-03-24.

Hughes, J.N., Annex, A., Eichelberger, C.N., Fox, A., Hulbert, A., Ronquest, M., 2015. GeoMesa: a distributed architecture for spatio-temporal fusion. In: Pellechia, M.F., Palaniappan, K., Doucette, P.J., Dockstader, S.L., Seetharaman, G., Deignan, P.B. (Eds.), Geospatial Informatics, Fusion, and Motion Video Analytics V. Vol. 9473, International Society for Optics and Photonics, SPIE, pp. 128–140. http://dx.doi.org/10.1117/12.2177233.

Kang, J., Körner, M., Wang, Y., Taubenböck, H., Zhu, X.X., 2018. Building instance classification using street view images. ISPRS J. Photogram. Remote Sens. 145, 44–59.

Kessing, M., 2022a. Deutsche Telekom sets even faster pace for fiber rollout. https://www.telekom.com/en/media/media-information/archive/dt-sets-even-faster-pace-for-fiber-rollout-1022644, Last visited: 2023-06-19.

Kessing, M., 2022b. Starke Jahresbilanz: 5G- und Glasfasernetze wachsen weiter. https://www.telekom.com/de/medien/medieninformationen/detail/starke-jahresbilanz-5g-und-glasfasernetze-wachsen-weiter-1023350, Last visited: 2023-06-23.

Kessing, M., 2023. Netze-Bilanz 2023: Glasfaser und 5G legen weiter zu. https://www.telekom.com/de/medien/medieninformationen/detail/netze-bilanz-2023-glasfaser-und-5g-legen-weiter-zu-1055912, Last visited: 2024-02-12.

Khan, Z., Anjum, A., Kiani, S.L., 2013. Cloud based big data analytics for smart future cities. In: Proceedings of the 2013 IEEE/ACM 6th International Conference on Utility and Cloud Computing. UCC '13, IEEE Computer Society, Washington, DC, USA, pp. 381–386. http://dx.doi.org/10.1109/UCC.2013.77.

Kocon, K., Bormann, P., 2021. Point cloud indexing using big data technologies. In: 2021 IEEE International Conference on Big Data (Big Data). IEEE, pp. 109–119.

Kocon, K., Krämer, M., Würz, H.M., 2022. Comparison of CNN-based segmentation models for forest type classification. AGILE: GIScience Series 3, 42. http://dx.doi.org/10.5194/agile-giss-3-42-2022.

Krämer, M., 2018. A Microservice Architecture for the Processing of Large Geospatial Data in the Cloud (Ph.D. thesis). Technische Universität Darmstadt, http://dx.doi.org/10.13140/RG.2.2.30034.66248.

Krämer, M., 2021. Efficient scheduling of scientific workflow actions in the cloud based on required capabilities. In: Hammoudi, S., Quix, C., Bernardino, J. (Eds.), Data Management Technologies and Applications. In: Communications in Computer and Information Science, Vol. 1446, Springer, pp. 32–55. http://dx.doi.org/10.1007/978-3-030-83014-4_2.

Krämer, M., Frese, S., Kuijper, A., 2019. Implementing secure applications in smart city clouds using microservices. Future Gener. Comput. Syst. 99, 308–320. http://dx.doi.org/10.1016/j.future.2019.04.042, URL http://www.sciencedirect.com/science/article/pii/S0167739X18304539.

Krämer, M., Senner, I., 2015. A modular software architecture for processing of big geospatial data in the cloud. Comput. Graph. 49, 69–81. http://dx.doi.org/10.1016/j.cag.2015.02.005, URL http://www.sciencedirect.com/science/article/pii/S0097849315000138.

Krämer, M., Würz, H.M., Altenhofen, C., 2021. Executing cyclic scientific workflows in the cloud. J. Cloud Comput. 10 (25), 1–26. http://dx.doi.org/10.1186/s13677-021-00229-7.

Krylovskiy, A., Jahn, M., Patti, E., 2015. Designing a smart city internet of things platform with microservice architecture. In: 2015 3rd International Conference on Future Internet of Things and Cloud. pp. 25–30. http://dx.doi.org/10.1109/FiCloud.2015.55.

Li, X., Ratti, C., 2019. Mapping the spatio-temporal distribution of solar radiation within street canyons of boston using Google Street View panoramas and building height model. Landscape Urban Plan. 191, 103387.

Liu, Z., Yang, A., Gao, M., Jiang, H., Kang, Y., Zhang, F., Fei, T., 2019. Towards feasibility of photovoltaic road for urban traffic-solar energy estimation using street view image. J. Clean. Prod. 228, 303–318. http://dx.doi.org/10.1016/j.jclepro.2019.04.262, URL https://www.sciencedirect.com/science/article/pii/S0959652619313538.

Lu, Y., 2019. Using google street view to investigate the association between street greenery and physical activity. Landsc. Urban Plan. 191, 103435.

Ludlow, D., Khan, Z., 2012. Participatory democracy and the governance of smart cities. In: 26th Annual AESOP Congress. Ankara, Turkey, 11th - 15th July.

Medeiros, C.B., Vossen, G., Weske, M., 1996. GEO-WASA-combining GIS technology with workflow management. In: Proceedings of the Seventh Israeli Conference on Computer Systems and Software Engineering. IEEE, pp. 129–139. http://dx.doi.org/10.1109/ICCSSE.1996.554918.

OECD, 2022e. Organisation for Economic Co-operation and Development (OECD). broadband portal. https://www.oecd.org/digital/broadband/broadband-statistics/, Last visited: 2024-02-02.

OECD, 2022f. Organisation for Economic Co-operation and Development (OECD). Fixed and mobile broadband subscriptions per 100 inhabitants - June 2022. https://web-archive.oecd.org/2023-07-13/132827-1.2.OECD-FixedMobileBB-2022-12.xls, Last visited: 2024-02-02.

OECD, 2022g. Organisation for Economic Co-operation and Development (OECD). Percentage of fibre connections in total broadband - June 2022. https://web-archive.oecd.org/2023-07-13/132819-1.10-PctFibreToTotalBroadband-2022-12.xls, Last visited: 2024-02-02.

OECD, 2022h. Organisation for Economic Co-operation and Development (OECD). Total fixed and mobile broadband subscriptions by country - June 2022. https://web-archive.oecd.org/2023-07-13/132822-1.1-TotalBBSubs-bars-2022-12.xls, Last visited: 2024-02-02.

Reiterer, A., Wäschle, K., Störk, D., Leydecker, A., Gitzen, N., 2020. Fully automated segmentation of 2D and 3D mobile mapping data for reliable modeling of surface structures using deep learning. Remote Sens. 12 (16), http://dx.doi.org/10.3390/rs12162530, URL https://www.mdpi.com/2072-4292/12/16/2530.

Sanchez, A., 2023. Breitbandausbau: Telekom bietet jetzt 6 Millionen reine Glasfaser-Anschlüsse. https://www.telekom.com/de/medien/medieninformationen/detail/telekom-bietet-jetzt-6-millionen-reine-glasfaser-anschluesse-1041702, Last visited: 2023-07-17.

Santana, E.F.Z., Chaves, A.P., Gerosa, M.A., Kon, F., Milojicic, D.S., 2017. Software platforms for smart cities: Concepts, requirements, challenges, and a unified reference architecture. ACM Comput. Surv. 50 (6), http://dx.doi.org/10.1145/3124391.

Schütz, M., 2016. Potree: Rendering Large Point Clouds in Web Browsers. Technische Universität Wien.

Schütz, M., Ohrhallinger, S., Wimmer, M., 2020. Fast out-of-core octree generation for massive point clouds. Comput. Graph. Forum 39 (7), 1–2, URL https://www.cg.tuwien.ac.at/research/publications/2020/SCHUETZ-2020-MPC/.

Shurtz, R., 2019. JSON compression: Alternative binary formats and compression methods. https://www.lucidchart.com/techblog/2019/12/06/json-compression-alternative-binary-formats-and-compression-methods/, Last visited: 2022-05-11.

Tukaani, 2009. Tukaani. The .xz file format. https://tukaani.org/xz/xz-file-format-1.0.4.txt, Last visited: 2022-10-18.

Wang, C.-Y., Bochkovskiy, A., Liao, H.-Y.M., 2022. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv preprint arXiv:2207.02696.

Wang, R., Peethambaran, J., Chen, D., 2018. LiDAR point clouds to 3-D urban models: A review. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 11 (2), 606–627. http://dx.doi.org/10.1109/JSTARS.2017.2781132.

Wong, K.-Y., 2022. Weights YoloV7. https://github.com/WongKinYiu/yolov7/releases/download/v0.1/yolov7-mask.pt, Last visited: 2023-06-26.

Würz, H.M., Kocon, K., Pedretscher, B., Klien, E., Eggeling, E., 2023. A scalable AI training platform for remote sensing data. AGILE: GIScience Series 4, 53. http://dx.doi.org/10.5194/agile-giss-4-53-2023.

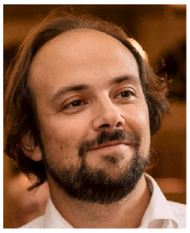
Zaharia, M., Chowdhury, M., Franklin, M.J., Shenker, S., Stoica, I., et al., 2010. Spark: Cluster computing with working sets. HotCloud 10 (10–10), 95.

Michel Krämer is deputy department head of the Spatial Information Management competence center at the Fraunhofer IGD and a lecturer at TU Darmstadt. He has a Ph.D. in Computer Science. His research interests are in processing of big geospatial data, progressive indexing, microservices, and Cloud Computing. Michel Krämer has been lead software engineer and scientific coordinator in a number of national and international research projects. He has authored open-source software such as the Steep Workflow Management System and GeoRocket. Michel Krämer contributes to many popular open-source frameworks including Vert.x and the Citation Style Language (CSL).

Pascal Bormann is a research assistant at the department of Spatial Information Management at the Fraunhofer IGD, a Ph.D. student at TU Darmstadt, and a lecturer at Darmstadt University of Applied Sciences. His Ph.D. topic is the processing of large-scale point cloud data using efficient and adaptive indexing techniques. His research interests include high-performance computing and systems programming.

Hendrik M. Würz is a researcher at the Fraunhofer Institute for Computer Graphics Research IGD. Before, he studied computer science and visual computing at the Technical University of Darmstadt. He is interested in the distributed processing of big data and the management of cloud resources. Hendrik M. Würz has experience in the preparation of geodata for visualization and the efficient execution of tasks in the cloud. He is a contributor to the open-source workflow management system Steep.

Kevin Kocon is a research associate at Fraunhofer IGD in Darmstadt. He earned his Master's degree in Computational Engineering at TU Darmstadt and is currently pursuing a PhD. His research focuses on the application of artificial intelligence and the handling of large geospatial data sets, in particular for computer vision tasks.



Tobias Frechen is a System Architect in the Geo-Hub at Deutsche Telekom IT GmbH. He received his degree in Computer Science from the University of Bonn. During his studies, he worked as a student assistant at the Institute for Hygiene and Public Health in the area of Medical Geography & Public Health. There he collaborated on various research projects and discovered his interest in geographic information systems, which he still deals with both in his spare time and professionally.

Jonas Schmid is Product Owner at Geo-Hub within Deutsche Telekom Technik GmbH. Earlier, he contributed to Germany's fibre optic expansion, strategically planning routes, and ensuring site selection. He also excelled as Project Manager in Mobile Mapping, overseeing operations, deployment areas, and data processing. Jonas is recognized for his tech-savviness, strategic thinking, and project management expertise.