Article

Establishing a Generic Geographic Information Collection Platform for Heterogeneous Data

Songcao Liu, Bozhao Li *, Yuqiao Chen and Zhongliang Cai

School of Resource and Environmental Sciences, Wuhan University, Wuhan 430079, China;
songcao.liu@whu.edu.cn (S.L.); yuqiaochen@whu.edu.cn (Y.C.); zlcai@whu.edu.cn (Z.C.)
* Correspondence: libozhao@whu.edu.cn; Tel.: +86-027-68778893

Abstract: Geographic information collection platforms are widely used for acquiring geographic information. However, existing geographic information collection platforms have limited adaptability and configurability, negatively affecting their usability. They do not support complete field collection workflows or capture data with complex nested structures. To address these limitations, this paper proposes a generic geographic information collection platform based on a comprehensive XML schema definition and a corresponding XML toolkit. This platform includes professional and non-professional versions of collection software, as well as a management system. Users can configure controls and define nested tables within this platform to collect heterogeneous and complex nested data. Moreover, the platform supports functions such as task assignment, local deployment servers, multitasking parallelism, and summary statistics of heterogeneous data, ensuring complete workflow support for field data collection. The platform has been applied in agriculture, forestry, and related fields. This paper uses the agricultural industry structure survey as a case study. Practical applications and our case study show that this platform can reduce software development costs, lower user knowledge prerequisites, and fulfill 95% of geographic information collection scenarios.

Keywords: generic geographic information collection platform; XML schema definition; workflow design; agricultural industry

1. Introduction

Geospatial data collection is a research priority in the field of spatial information. Geospatial data collection includes field data collection, remote sensing data collection, and geographic information science data conversion [1]. Among these methods, field data collection is an essential means of gathering ground-truth spatial information [2]. Field data collection plays an important role in many geospatial applications, including crop yield estimation [3], hydrologic studies [4], mobile health [5], township land resources surveying [6], and so on. The methods of collecting geographic information in the field have undergone many developments. A few decades ago, traditional field geographic information collection relied heavily on field personnel to record data on paper maps or paper forms manually [7]. The process is time-consuming, inefficient, error-prone, and complex [8–10]. As time went by, with advancements in mobile devices, computer sciences, and satellite navigation technologies, researchers developed mobile device-based geographic information collection platforms [11–13]. These platforms integrate electronic maps, GPS positioning, information collection, and information uploading, liberating field personnel from traditional surveying constraints [14,15]. They have been applied in many fields, such as disaster management [16], landscape ecology [17], soil research [18], and forest survey [19].
When mobile data collection platforms first emerged, researchers focused on developing customized mobile data collection platforms to serve specific scenarios [20]. For instance, Thoonil was specifically developed for fishermen to acquire and upload fishing information [21], STATUS was developed for the construction industry [22], and the CMR App was primarily used to record the tracks of wild animals [23]. These customized platforms can enhance the efficiency of geographic information collection and are tightly coupled with their intended uses [7,24]. However, they cannot be applied to new collection scenarios, resulting in users having to develop new software [25]. Customized platforms often have limitations such as low flexibility, weak architecture reusability, and increased enterprise costs [26]. Generic software is designed to support users in various application scenarios without modification in the source code, thereby reducing the problem of repeated development for different application scenarios [27]. Based on the idea of generic software, researchers began to design generic mobile collection platforms.

The generic mobile collection platform developed for field data collection needs to support users who are collecting data in various field collection scenarios. Various application scenarios mean a generic mobile collection platform must support those working under diverse environmental conditions and collect structurally varied spatial data [28]. Based on the existing research, environmental conditions include different network environments, common operating platforms, various basemap formats, and so on [29,30]. Popular mobile collection platforms like ArcGIS Field Map [31], QField [32], Open Data Kit (ODK) [33], and FAIMS [34] support common spatial databases and attribute data under various environmental conditions. For example, ArcGIS Field Map and QField allow users to configure structured, text, and multimedia data, offering visualization controls such as drop-down boxes and multiple-choice buttons [35]. However, there is a nested attribute data structure in common collection scenarios, such as in-house surveys, where a house contains multiple households, each with multiple phone numbers, and the number of users and phone numbers is variable [36]. Current field collection software lacks solutions for this nested data issue. In addition, studies indicate that the standard geographic information collection workflow includes task configuration, task assignment, data collection, data review, data storage, data upload, data visualization, and data analysis [37,38]. However, most mobile collection platforms focus solely on data collection and miss key features like background management systems, making them incapable of supporting workflows involving task allocation and heterogeneous data visualization [28,34]. Researchers must integrate ArcGIS Field Map or QField with commercial online storage services and desktop software to manage data collection, upload, storage, and visualization [28,39–42]. This process requires specialized knowledge, is not user-friendly, and supports only online data storage, not local servers [28]. Although ODK offers local server deployment, it requires a domain name and only supports PostgreSQL for data storage [33,43]. Importantly, these combined platforms still lack functions for task assignment and monitoring collection progress [44]. These problems may lead to erroneous data collection and waste of effort in assigning tasks, especially in large-scale data collection tasks where many task forces are distributed over a wide area, making communication extremely inconvenient [39]. Table 1 intuitively shows the detailed functions supported by the currently popular generic collection platforms. In summary, the current generic mobile collection platforms still need more versatility and only focus on field data collection, ignoring other equally important steps in field geographic information collection work. Therefore, it is necessary to continue to develop a generic geographic information collection platform with greater versatility and support for the entire field geographic information collection process.

To bridge the gap, this paper proposes a generic geographic information collection platform. The platform comprises collection software supporting multiple runtime platforms such as iOS, Android, and the web, as well as a management system compatible with various databases such as PostgreSQL, MySQL, and Oracle. Our platform adopts a comprehensive XML schema definition to address heterogeneous data collection
problems and designs an XML toolkit capable of parsing configuration files, creating database tables, interacting with databases, and rendering forms in collection software.

An innovative aspect of this platform is its diverse geographic information data collection workflow and rich set of configurable controls, thereby supporting complex data collection scenarios. Moreover, our platform supports multitasking parallelism, allowing users to handle multiple tasks simultaneously. In order to improve the adaptability of the platform, our platform also offers cross-platform compatibility, cross-database compatibility, and the ability to process data collection tasks offline or online.

Table 1. Comparison of existing software.

<table>
<thead>
<tr>
<th>Software</th>
<th>This Platform</th>
<th>QField</th>
<th>ArcGIS Field Map</th>
<th>ODK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positioning</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Trajectory Recording</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Geometric Feature Collection</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Control Selection</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Offline Data Collection</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Data Upload</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Shapefile Support</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Local Database Support on Mobile Phone</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Remote Database Support</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Support Multiple Databases for Local Central Server</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Navigation</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>WMS, WTMS, WFS</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Local Central Server</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Remote Central Server</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Statistical Analysis</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Task Allocation</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Nested Tables</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Script Operations</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>basemapConfiguration</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

“✓” indicates that the platform has this function.

2. Materials and Methods

2.1. Platform Architecture

Our generic platform is designed to support the entire process of field geographic information collection. As shown in Figure 1, it consists of collection software and a management system, which communicate via network transmission. The platform was developed using C++ 17, Java 8, ObjectC 2.0, and JavaScript ECMAScript 2018, customized to particular field deployments using reusable-definition XML-based configuration files.

The collection software leverages a custom XML toolkit and dynamic page technology to provide tailored functions for different application scenarios. To maximize compatibility across various environments, it integrates technologies such as mobile databases, runtime scripts, and GPS positioning. We offer professional applications for iOS, Android, and Web to accommodate different systems used by collectors. For volunteer geographic projects, we provide mini programs that allow volunteers to quickly collect simple spatial data.
The management system also uses a custom XML toolkit for task configuration, configuration files, and database operations. Utilizing Geohash, spatial indexing technology, and OpenLayer v6, our management system enables managers to allocate tasks, schedule personnel, and rapidly visualize heterogeneous data. Building on these functionalities, we implemented a comprehensive field geographic information collection workflow in the management system and designed multiple work modes.

![Platform architecture](https://via.placeholder.com/150)

**Figure 1.** Platform architecture.

### 2.2. XML Schema Definition

XML can effectively describe and model the specifications, structures, and functions needed to solve specific domain problems and is often combined with components and frameworks to offer quick and flexible solutions [42,43]. Currently, some collection platforms use XML configuration files to control interface generation, making the process faster and more cost-effective [28]. Therefore, we designed an XML schema for field geographic information collection tasks to define data structures, UI, and various collection tools. By reading different XML-based configuration files at runtime, our collection software can quickly adapt to different data collection scenarios. Figure 2 illustrates our XML schema design and the process of writing XML configuration files. To improve usability and reduce the learning curve, we provide a visual interface in the management system to generate these configuration files.
We designed the XML schema comprising five components: spatial editing tools configuration, positioning tools configuration, basemap tools configuration, database table configuration, and collection content configuration. Table 2 details the XML elements and functions corresponding to each part. In the spatial editing tool configuration, positioning tool configuration, and basemap configuration sections, users can set spatial shape-editing tools, positioning tools, and basemaps for tasks. In the database tables and collection content sections, they can also customize attribute data structures. In order to allow a collection page to automatically extract additional information from user behavior, we employ the observer mode to implement three runtime scripts: value script, group script, and visibility script. These scripts enable the collection interface to adapt automatically to user actions, such as changing field visibility based on user choices.

Table 2. Elements of XML schema.

<table>
<thead>
<tr>
<th>Element</th>
<th>Attributes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;config&gt;</code></td>
<td>-</td>
<td>Root element. Stores the tools needed for field geographic information collection. Its sub-elements include: <code>&lt;drawtool&gt;</code>, <code>&lt;position&gt;</code>, and <code>&lt;basemap&gt;</code>.</td>
</tr>
<tr>
<td><code>&lt;meta&gt;</code></td>
<td>-</td>
<td>Stored geometry drawing tools. Its sub-element is <code>&lt;type&gt;</code>.</td>
</tr>
<tr>
<td><code>&lt;drawtool&gt;</code></td>
<td>-</td>
<td>Store position tools. Its sub-element is <code>&lt;type&gt;</code>. Stores positioning tool or geometric feature drawing tool. The attribute <code>name</code> records the corresponding tool name.</td>
</tr>
<tr>
<td><code>&lt;position&gt;</code></td>
<td>-</td>
<td>Stores basemaps that can be used. Its sub-element is <code>&lt;layer&gt;</code>.</td>
</tr>
<tr>
<td><code>&lt;type&gt;</code></td>
<td>name</td>
<td>Stores the basemap. Its attribute type is used to store the URL of the basemap. SubType is used to record whether it is a vector layer or an image layer, and index is used to record the visible order of the basemap.</td>
</tr>
<tr>
<td><code>&lt;basemap&gt;</code></td>
<td>-</td>
<td></td>
</tr>
<tr>
<td><code>&lt;layer&gt;</code></td>
<td>name, index, sub-type, type, group</td>
<td></td>
</tr>
</tbody>
</table>
2.3. XML Toolkit

XML for dynamic page generation is widely used in UI-related studies for page layouts [45,46]. XML-based dynamic UI control and database access are mature technologies that enhance application flexibility and maintainability [34,47]. Therefore, we developed an XML toolkit. The toolkit can generate corresponding XML configuration files based on user configuration and the XML schema we defined. Based on the XML configuration file, our toolkit can control the collection interface content and operate the database. Figure 3 shows the detailed structure of the XML toolkit, its position in the system, and its interaction with other modules. It is noteworthy that in the XML toolkit, we have implemented support for the current mainstream databases.

![Figure 3. XML toolkit structure.](image-url)
As shown in Figure 3, the XML toolkit includes the XML Generator, XML Parser, and XML4DB. First, the XML Generator can create configuration files based on user settings in the visual interface. Second, the XML Parser can convert XML configuration files into an object in memory. This object defines the UI of the collection interface and interacts with the database tables. Finally, XML4DB uses the generated object to perform database operations. Notably, we have integrated popular spatial database drivers and spatial extensions into XML4DB, enabling direct data import into these databases. This integration reduces the need for data conversion. Additionally, the XML toolkit, combined with the XML configuration file, makes switching between collection tasks and collection pages easy. We will show the XML Generator interface and various collection interfaces in the collection software to demonstrate the XML toolkit’s capabilities.

2.4. Workflow Design

We implemented the necessary functions in the management system following the standard field geographic information collection process. Additionally, we designed four modes for data collection and verification to allow users to flexibly assign personnel and complete tasks. Figure 4 illustrates the detailed workflow of field data collection supported by our platform.

As shown in Figure 4, the platform’s workflow is divided into six steps: task configuration, task allocation, field data collection, field data review, backend data quality review, and data visualization and analysis. During the task configuration phase, we use the XML toolkit and XML schema to enable users to quickly customize collection pages based on specific scenarios. In the task allocation phase, we implemented a fast task area division and allocation method using the Geohash coding algorithm. This method divides the task area into appropriately sized vector tiles based on the selected partition level. The detailed process is as follows:

Firstly, this method obtains the minimum bounding rectangle of the task area. Secondly, according to the chosen partition level, the size of the vector tile corresponding to
the Geohash encoding is confirmed. Thirdly, obtain all vector tiles covering the minimum bounding rectangle. After eliminating the tiles that do not intersect with the task region, the remaining tiles are the task sub-regions. Finally, assign the collectors to each tile in order.

For the field data collection and field data review phase, we have designed multiple flexible working modes, including unrestricted collection, partitioned collection, partitioned review, and synchronized collection and review. The management system will decide whether to execute the collection or verification process and how to assign task personnel according to different modes. In the data visualization and analysis phase, we use spatial indexing technology and the Geohash algorithm in the database to facilitate rapid access to spatial data. We employ the Geohash clustering algorithm, along with window size, lower-left corner coordinates, and current scale, to enable fast filtering and clustering of the displayed data.

3. Result and Analysis

3.1. Platform Functionalities

Our platform improves field data collection by providing a range of features that specifically address the needs of field data collection in various scenarios. These features allow users to collect, review, manage, visualize, and analyze heterogeneous data with appropriate workflows. These features include:

Various spatial feature editing tools: The platform offers vector editing tools such as point, line, polygon, rectangle, circle, and so on.

Various positioning methods: This platform offers multiple positioning methods to obtain the current location and trajectories of collection personnel. These position methods include GPS, network, Bluetooth, and third-party positioning services like Baidu, Amap, and Google.

Multi-source layer loading: This platform allows the loading of various map sources, including offline data and online services. The offline data include ttpk and shapefile. The online services include Baidu, Amap, Google, and map services that comply with standards such as web feature service (WFS), web map service (WMS), and web map tile service (WMTS).

Supports the collection of various types of data: This platform supports the collection of multiple data types within a single record, including structured data, geospatial data, multimedia data, and complex nested data.

Support multiple collection controls: This platform supports those collecting structured data, text data, multimedia data, and nested data. It provides rich controls including drop-down menus, check boxes, radio buttons, a collection table, and a relation table, etc.

Supports the collection of heterogeneous data: This platform allows users to create tasks based on usage scenarios. Within a single task, users can create multiple collection interfaces, each corresponding to a type of data that needs to be collected.

Maintain data structure consistency between collection software and management system: Data can be transferred smoothly between the two parts without any format conversion or field mapping.

Script and Dynamic UI: This platform provides three types of scripts: value, visibility, and group. These scripts can calculate values automatically, toggle control visibility automatically, and manage option visibility automatically.

Support automatic generation or manual writing of configuration files: Users without programming experience can use the XML generator to create XML configuration files interactively. For more advanced users, the generator also supports downloading template files and importing user-created configuration files.

Multiple data collection and review modes: This platform provides four data collection and verification modes, including unrestricted collection, partitioned collection, partitioned review, and synchronized collection and review.
Compatible with common spatial databases: This platform can connect to common spatial databases such as PostgreSQL, MySQL, SQLite, and Oracle.

Rapid visualization and analysis of heterogeneous data: Users can quickly browse existing collected data in the management system. The system will cluster and display the data according to the current scale.

Support offline collection: The device can work offline for a long time. The data collected in the offline state will be stored in the mobile database.

Task area division and allocation: The platform supports dividing tasks into regular vector tiles and allocating the task areas covered by the tiles to task personnel.

Real-time monitoring of collection tasks: Users are able to view the collection progress, location, and trajectory of the collection personnel in real-time. The collection software will automatically remind the collection personnel who leave the assigned task area and plan a reasonable collection route for the collection personnel.

3.2. Illustrative Examples

This platform has already been successfully applied in over thirty organizations in China, such as surveying institutes and universities across more than ten provinces and cities, and has received positive user feedback. We illustrate the potential of this platform by using the example of a regional agricultural industry structure survey in Guizhou, China. Guizhou Province features rugged terrain with scattered farmland, with a total area of $1.76 \times 10^5 \text{km}^2$, making regional agricultural industry structure surveys challenging [48]. According to the Köppen climate classification, Guizhou is categorized into humid subtropical climate (cwa, cfb) and subtropical highland climate (cwb) [49]. The persistent cloudiness and humidity in some areas make it impossible to conduct agricultural surveys that rely solely on remote sensing technologies like satellite imagery or aerial photography [50]. Due to these geographic and climatic challenges, on-site data collection in Guizhou is essential. By using the field geographic information collection platform, researchers can navigate the rugged terrain, visit scattered areas of farmland, and gather valuable on-site observational data. This platform collects high-resolution data that can supplement and validate remote sensing information, offering a more comprehensive understanding of Guizhou’s agricultural industry structure.

In a broad sense, agriculture encompasses crop cultivation, livestock production, fisheries, forestry, and the processing of agricultural by-products [51]. This demonstration takes crop cultivation and livestock production as examples.

3.2.1. Defining Task Requirements

In this collection task, we faced three main challenges. Based on these challenges, we assessed and demonstrated the platform’s effectiveness and usability:

Challenge 1: The task covers a wide area with many participants. Each person has specific duties and works in different regions. Managers need to quickly configure tasks, allocate them, and monitor progress in real-time to optimize personnel management.

Challenge 2: Some remote areas have few areas of farmland, making collection difficult. Other areas near cities with good transportation and many areas of farmland make collection easier. Moreover, there may be recently collected data for some areas, which only need verification before being used directly. We need flexible collection modes for different areas to reduce workload while ensuring data quality.

Challenge 3: The task involves collecting geographic information for crop cultivation and livestock production. Data for these two industries must be stored separately. Each industry has multiple objects requiring data collection, each with different attributes. For example, in crop cultivation, we need to collect data on rice, corn, potato, soybean, and cabbage. Each crop has specific attributes about which data need to be collected.
3.2.2. Challenge 1: Support the Entire Process of Field Collection

The collection area for this task is vast, and many personnel are involved. Gathering all personnel to discuss and assign tasks is difficult and inefficient. Our platform effectively resolves these issues.

Managers configure the necessary collection tools and contents interactively on the platform, as shown in Figure 5a. Once configured, they upload the task area to the server and add all personnel. The server divides the task area evenly among the personnel based on the area, number of personnel, and selected division level. As shown in Figure 5b, managers can manually adjust these divisions on the platform. Task information is then transmitted in real-time to the collection software of the respective task personnel. During the collection process, managers can track the location of the personnel and monitor progress in real-time. They can also view the field data uploaded by the personnel and check if it has been verified by inspectors, as shown in Figure 5c.

![Figure 5](image_url)

**Figure 5.** (a) Task configuration interface; (b) task allocation interface; and (c) data management interface.

3.2.3. Challenge 2: Flexible Collection Modes

Our platform allows users to adopt various work modes to manage personnel scheduling and quality requirements in fieldwork. In this collection task, managers adjust the collection mode manually based on the difficulty of accessing the collection areas. The task allocation interface is shown in Figure 5b.

For difficult-to-reach areas, managers set the mode to partitioned collection. In this mode, only collection personnel need to gather data, and no field inspectors are required for second verification. Managers visually check the data uploaded by collection personnel in the management system.

For easy-to-reach areas, managers set the mode to synchronized collection and review. The system assigns both collection personnel and field inspectors to these areas. Figure 6a shows the spatial data collection interface for field collectors. Field review personnel download the data uploaded by collection personnel and verify it on-site. Figure 6b shows the software interface for field inspectors.

For areas where there are existing data that only need verification, managers set the mode to partitioned review. In this case, the system assigns inspectors to check if the existing data have changed and to update them if necessary.
3.2.4. Challenge 3: Customized Collection Interface

Our platform supports customizing collection pages based on application scenarios and the collection of nested data. In this task, managers first identify the various collection tools and base maps that the collectors need to use. Then, managers confirm the crops and livestock breeds needed for crop cultivation and livestock production for the agricultural structure survey in Guizhou Province, along with the necessary attributes for each breed. The collection content and corresponding control items for agriculture and livestock are detailed in Tables 3 and 4.

Control items are selected based on the type and content of each indicator. For example, “Crop Cultivation” uses a “CheckBox” to select from local crop types in Guizhou Province, reducing input errors. “Contact Information” uses a “CollectionTable” to record multiple contacts for each area of farmland, including name, phone number, and ID card. In addition, for “Crop Type”, only rice, corn, potato, soybean, and cabbage are included in the survey area. For “Livestock Breed”, only pigs, cattle, and sheep are included in the survey area.

When recording “Contact Information”, managers use the “CollectionTable” control to store nested data. For crop cultivation, the implementation is as follows: managers add
a new table named “ContactInformationTable” in the task configuration interface; they then add three text input boxes for the contact’s name, phone number, and ID card; and finally, they use the “CollectionTable” control in the crop cultivation interface to point to the “ContactInformationTable”. This allows collection personnel to enter contact information stored in the corresponding database table.

### Table 3. The collection content of crop cultivation.

<table>
<thead>
<tr>
<th>Control Description</th>
<th>Control Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spatial Information</td>
<td>Shape</td>
</tr>
<tr>
<td>Crop Type</td>
<td>CheckBox</td>
</tr>
<tr>
<td>Irrigation Method, Storage Method, Remark</td>
<td>InputBox (Text)</td>
</tr>
<tr>
<td>Photos</td>
<td>Media</td>
</tr>
<tr>
<td>Operating Entity</td>
<td>RelationTable</td>
</tr>
<tr>
<td>Contact Information</td>
<td>CollectionTable</td>
</tr>
<tr>
<td>Plant Time</td>
<td>DateSelector</td>
</tr>
<tr>
<td>Total Yield, Area, Average Yield, Grain Weight, Tuber Weight, and others</td>
<td>InputBox (Double)</td>
</tr>
</tbody>
</table>

### Table 4. The collection content of livestock production.

<table>
<thead>
<tr>
<th>Control Description</th>
<th>Control Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spatial Information</td>
<td>Shape</td>
</tr>
<tr>
<td>Livestock Breed, Feeding Method Remarks</td>
<td>CheckBox</td>
</tr>
<tr>
<td>Photos, Videos</td>
<td>InputBox (Text)</td>
</tr>
<tr>
<td>Operating Entity</td>
<td>Media</td>
</tr>
<tr>
<td>Contact Information</td>
<td>RelationTable</td>
</tr>
<tr>
<td>Pasture Area, Unit Production Cost, Total Output Value, and others</td>
<td>InputBox (Double)</td>
</tr>
</tbody>
</table>

Based on these tables and conventions, managers configure the management system and generate the corresponding XML configuration file. Figure 7 shows part of the XML file generated by the system. This file is then transmitted to the collection software of the task personnel.
Figure 7. XML Configuration file.

After the member of personnel accepts the task, the collection software will display the corresponding collection interface according to the transmitted XML configuration file and related task resources. Figure 8a shows the spatial data collection interface that appears after the task is accepted. The icon in the upper right corner allows users to quickly switch between crop cultivation and livestock production.
Figure 8. (a) Spatial element collection interface. The gray area is the work area of the collectors. (b) Crop cultivation attribute collection interface for rice. (c) Crop cultivation attribute collection interface for potato. (d) Livestock production attribute collection interface for pig. (e) Crop cultivation attribute collection interface after filling in the data. “ContactInformationTable” is a “CollectionTable” control that supports users in adding multiple data. “OperatingEntityTable” is a “RelationTable” control. (f) The attribute collection interface displays the value script. Unit yield is automatically calculated based on total yield and area.

Figure 6a shows spatial data collected by personnel, indicating that our collection software supports geometric elements like points, lines, and polygons. After drawing these elements, personnel must enter the corresponding attribute information in the attribute collection interface. Figure 8b–d display this interface dynamically based on user selections. Figure 8b,c show how different collection items appear when different crops are selected, thanks to visibility scripts. Figure 8b,d illustrate the differences between the crop cultivation and livestock production collection interfaces.

Importantly, collection personnel can record nested data in the attribute collection interface. Figure 8e shows the process of collecting “Operating Entity” and “Contact Information”. The figure demonstrates that collection personnel can add multiple contacts but only one operating entity, proving that our platform supports nested data collection.
This task also utilizes value scripts to minimize user errors. The platform allows users to define numerical relationships between different attributes, reducing workload and calculation errors. For instance, during crop cultivation data collection, entering the area and yield will automatically calculate the yield per unit area. Figure 8e showcases the functionality of these value scripts in a real-world scenario.

4. Discussion

Our platform meets common and specific field collection needs, reducing the difficulty for field workers. The collection software enables field collectors and verifiers to record on-site geographic information efficiently, even in complex environments like offline conditions. The various components of the collection software enable easy collection of data types such as images, videos, and audio. By using selection boxes, grouping scripts, value scripts, and visibility scripts, the collection software automatically inherits or calculates as much information as possible from existing data, enhancing recording speed and minimizing human errors.

It is worth noting that the platform helps the task team carry out large-scale collection activities in an orderly manner. Sharma et al. reported that in large-scale field geographic information collection tasks, team members often spend much time on tedious telephone conferences to assign tasks and allocate task-related resources [39]. In our platform, managers in the team can use the management system to quickly assign collection tasks and allocate task-related resources, such as vector files of the collection area. These features reduce the workload for managers and ensure field personnel are in designated work areas, minimizing task duplication. This ensures that there is no duplication of work between field personnel.

Next, our platform has multiple quality assurance mechanisms. Sharma and colleagues reported that managers need to spend a lot of time to ensure data quality, including integrating data delivered by collectors and visualizing and reviewing the integrated data [39]. These tasks exhaust mid-level managers responsible for field delivery. Our platform distributes the task of ensuring data quality into three processes, reducing the workload of managers who need to verify data while further strengthening data quality control. First, during field collection, the collection software will limit the type of data that collectors can enter based on the control. Second, the management system will automatically send the data uploaded by the collectors to the corresponding field data reviewers. For projects with a tight number of task personnel, managers can monitor the collection progress of the current task and reallocate task personnel according to the progress, for example, by adopting the synchronized collection and review mode so that nearby collectors can go to the field to verify each other’s data. Finally, managers can visually review the data verified on the field in real-time and store it in the database without data integration.

Finally, our platform supports the seamless flow of collected data from field devices to the final project database and avoids intermediate storage in commercial online cloud services. This addresses limitations in commonly used field collection software, such as Esri products using ArcOnline for data storage and synchronization. These limitations include slow data upload or download speeds due to limited Internet conditions [7]. In addition, our platform’s management system can run on private servers and support local area network data transmission. This can effectively prevent sensitive data, such as personal privacy, from entering the Internet or being stored in commercial online cloud servers.

5. Conclusions

This paper presents a generic geographic information collection platform designed to support the collection of heterogeneous data. The platform offers comprehensive support throughout the entire workflow, including task configuration, task assignment, geographic information collection, data review, data storage, data analysis, and data
visualization. It caters to the needs of collection personnel, data review personnel, and management personnel, promoting the generalization of geographic information collection software and the application of geographic information collection software across various fields. By standardizing the process of geographic information collection, the platform enhances data efficiency and quality.

This platform still has some problems. While the current platform can handle most geographic information collection scenarios for two-dimensional data, it currently lacks support for three-dimensional data collection and visualization. To address this limitation and keep pace with the rapid advancements in geographic big data, future developments will focus on incorporating functionalities for collecting and displaying the attribute information of three-dimensional geographic information data. Additionally, the platform incorporates an artificial intelligence interface, paving the way for integrating intelligent technologies such as geographic information large language models and image recognition. These advancements will further enhance the user experience, making the platform more convenient and intelligent.

Author Contributions: Conceptualization, S.L. and B.L.; software, S.L. and B.L.; validation, S.L. and Y.C.; resources, Z.C.; writing—original draft preparation, S.L.; writing—review and editing, S.L., B.L., Y.C., and Z.C.; supervision, B.L.; funding acquisition, Z.C. and B.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No. 42301532) and the China Postdoctoral Science Foundation (No. 2022M722442).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding author. The data are not publicly available due to privacy.

Conflicts of Interest: The authors declare no conflicts of interest.

References

32. Cavallini, P.; Ghetta, M.; Cavallini, U. From the field to the clouds: Data collection and publication with QGIS. GEOmedia 2022, 26, 6–11.


Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.