# TeamDR: A Data Repository Management System for Research Teams (Postprint)

**Authors:** Liu Feng, Li Jianhui, Zhang Jin: Han Fang, Liu Ang

**Date:** 2017-10-11T00:00:00+00:00

## Abstract

【Purpose】To address the issue of scattered research data in scientific teams lacking effective storage, management, and reusability, we developed a specialized data repository management system named TeamDR. 【Application Background】 TeamDR is a convenient Web-based application tool that supports research team users such as research groups in organizing, storing, managing, and collaboratively sharing research data; it uses Java as the primary programming language and offers two versions: a cloud service version available for immediate use upon registration, and a local installation version. 【Method】 To address the organization and management of diverse research data types, we designed a dynamic metadata content template and, concurrently, to ensure scalable data storage capacity and high query performance, adopted MongoDB as the storage solution. 【Results】 TeamDR has implemented key functionalities for research team data storage and management, including dynamic metadata templates, hierarchical sharing controls, and full-text metadata retrieval, with trial feedback demonstrating that it satisfies users' pressing needs in data storage and management. 【Conclusion】 The TeamDR system can effectively address the urgent fundamental requirements for team research data storage and management, sharing and collaboration, and discovery and association, though there remains room for further improvement in functional convenience, completeness, and extensibility.

## Full Text

## TeamDR: A Data Repository Management System for Research Teams

**Liu Feng[1],[2],[3], Li Jianhui[1], Zhang Jin[1], Han Fang[1], Liu Ang[1]**
[1](Computer Network Information Center, Chinese Academy of Sciences, Beijing 100190, China)
[2](National Science Library, Chinese Academy of Sciences, Beijing 100190,

China)
[3](University of Chinese Academy of Sciences, Beijing 100049, China)

## Abstract

[**Objective**] To address the problem of scattered research data in scientific teams lacking effective storage, management, and reusability, we developed a specialized data repository management system called TeamDR. [**Context**] TeamDR is a convenient Web application tool that supports research teams such as project groups in organizing, storing, managing, and collaboratively sharing research data. Developed primarily in Java, it offers both a cloud-based version available upon registration and a standalone local installation version. [**Methods**] To address the challenge of organizing and managing diverse research data types, we designed dynamic metadata content templates. To ensure scalable data storage capacity and high query performance, we adopted MongoDB as our storage solution. [**Results**] TeamDR implements key functions for research team data storage and management, including dynamic metadata templates, hierarchical sharing controls, and full-text metadata search. Trial feedback demonstrates that it meets users' urgent needs in data storage and management. [**Conclusions**] The TeamDR system effectively addresses the fundamental and urgent needs of research teams in data storage and management, sharing and collaboration, and discovery and linking. However, there remains room for improvement in terms of functional convenience, completeness, and extensibility.

**Keywords:** Scientific research teams; Data management; Data repositories; TeamDR

The flourishing development of data-intensive research has made data management an integral component of scientific activities, thereby accelerating the rapid development of scientific data repositories (DR) [1]. Currently, data repositories are mainly classified into four categories: institutional repositories, disciplinary repositories, multidisciplinary repositories, and project-specific repositories [2]. In terms of openness, disciplinary and multidisciplinary repositories demonstrate the strongest accessibility as they serve broad research communities, while institutional and project repositories are often confined to specific institutions or projects. Regarding depth of disciplinary service, disciplinary repositories, which typically provide long-term services for specific fields, exhibit stronger systematic and specialized service capabilities. In terms of breadth of disciplinary coverage, multidisciplinary and institutional repositories hold clear advantages [1]. However, overall, the construction of data repositories both domestically and internationally has primarily focused on public service DRs centered on storing, processing, and publishing data. Less attention has been paid to building repositories for the relatively scattered storage of research process data. In light of this, we developed TeamDR [3], a tool focusing on data storage and management issues for research teams, employing dynamic template design and MongoDB-based storage solutions to help research teams effectively accomplish data management tasks.

### 2.1 Current Status

With the development of digital environments both domestically and internationally, scientific and production data have rapidly proliferated and accumulated. To meet the processing needs of data professionals, various data management and sharing application tools have emerged as important auxiliary means for effective data management and research [4]. To address diverse needs in research data storage and management, foreign developers have created a series of data management systems and storage tools, including CKAN [5], a powerful open-source data portal platform developed by the non-profit Open Knowledge Foundation; UC3 Merritt [6], a novel repository service system developed by the University of California Digital Library Curation Center; Figshare [7], a platform for storing and freely sharing research data; and authoritative ecological data repositories such as Dryad, DataStage, DataBank, and Scholar Sphere [8].

(1) Figshare represents a new approach to sharing open scientific data, built on the principles of discoverability, shareability, and citability. On Figshare, researchers can publish data in a citable and searchable manner. All uploaded images and media are marked with CC licenses, while all datasets are released under CC0. Another important feature of Figshare is its encouragement of publishing unpublished negative data and images, which prevents other researchers from duplicating unsuccessful work [9]. By employing Amazon' s cloud-based data management system, Figshare ensures the security and reliability of data storage.

(2) Scholar Sphere is an open repository service system developed by Pennsylvania State University, enabling faculty, students, and various personnel to collect and store research outputs while creating a persistent, readable, and citable record. Supported resource types include papers, presentations, publications, datasets, survey analyses, academic journals, data, technical reports, audio-visual materials, annual reports, briefings, and other achievements. Researchers can also use this service to fulfill funding organizations' requirements for sharing and managing research data.

Domestically, a series of excellent data management platforms have also emerged, such as Team Document Library, Baidu Cloud Drive, Baihui Creator, Goukuai, and Jiandaoyun. (2) Jiandaoyun is a tool that implements data collection and management through forms, allowing users to build their own data management software through form customization [11]. It primarily provides functions for data collection, data management, and application customization. Its permission control functions enable internal management using forms. Jiandaoyun can create various online registration forms, feedback forms, and survey forms for public data collection, as well as complete data organization and collaboration based on enterprise internal data, while providing rich chart components for data analysis.

A comparison of key features among these different data management applications is presented in Table 1 . The comparison results indicate that although

existing data management tools for research teams each have distinctive features in terms of storage and functionality, they fail to adequately meet the most fundamental needs of research data storage and organizational management in terms of comprehensive services.

### 2.2 Application Requirements

Investigation and analysis reveal that research teams currently have urgent application needs in data management, prominently manifested in the following aspects: (1) Research data is scattered among individual researchers, making continuous storage and accumulation impossible and leading to high risk of loss. (2) The research process is separated from research data, lacking effective electronic integrated storage and management tools. (3) Research process management is non-standardized, with severe lack of detailed metadata for research data, making data difficult to understand, trace, verify, and reuse. (4) Research data sharing and service functions are extremely inadequate, with monolithic data sharing models, difficult data retrieval and synchronization, reducing research efficiency.

### 2.3 Positioning and Design

Based on the above analysis and requirements, TeamDR is positioned as a tool for storing, managing, and collaboratively sharing research data from diverse sources and of multiple types for research teams. It currently offers both cloud-based and local versions. The tool focuses on three main aspects: (1) storage and management of research data; (2) providing data sharing and collaboration for project groups; and (3) facilitating effective data discovery and linking.

(1) **Storage and Management** As a data repository management system for research teams, TeamDR focuses on the continuous accumulation and effective management of various research data resources for project groups, achieving long-term preservation and inheritance of data resources through file or relational database storage. The data sources and storage type structure for building project group data resource repositories are shown in Figure 1 [Figure 1: see original paper].

For organizational management, TeamDR employs a structure based on directory hierarchies and data collections. Data is organized and archived according to classification directories, with research-topic-oriented data collections serving as the basic storage units. The system provides dynamically customizable functions for dataset metadata items and settings. The organizational structure is illustrated in Figure 2 [Figure 2: see original paper]. TeamDR is also compatible with relational data, supporting online collaborative editing and retrieval of data, as well as import and export between Excel forms and relational database tables.

TeamDR supports data collaboration and sharing within project groups. Group members can collaboratively create data content for collections and relational

databases, achieving flexible sharing of various data through permission settings and tag annotations while ensuring system security. The project group data repository is divided into three parts: personal space, project team space, and project group space. Personal space stores user private data, project team space stores collaborative work data organized by project (sub-project), and project group space stores public project group data. User permission settings for data sharing across these spaces are shown in Figure 3 [Figure 3: see original paper].

To meet project group members' needs for sharing public research data, the system provides basic templates for data websites. Project group administrators can quickly publish data sharing websites through simple configuration, as shown in Figure 4 [Figure 4: see original paper].

(3) **Discovery and Linking** Each type of data collection in TeamDR features customizable detailed metadata information. Research users can flexibly customize different types of metadata content templates according to specific disciplinary needs, freely configuring various input attributes of data templates to meet domain-specific data management requirements [12]. Through linking, original data from research topic processes can be quickly located. The system provides full-text search functionality, making research data understandable, verifiable, and traceable. The metadata template customization process is shown in Figure 5 [Figure 5: see original paper].

### 2.4 Technical Approach

TeamDR adopts a modular design approach, planning the system into multiple service modules, each further subdivided into sub-service modules. System functions include three main components: storage management of data resources, retrieval of data resources, and sharing of data resources. Auxiliary functions also include data publishing and message management.

The overall system design employs an MVC architecture, layering model maintenance, data presentation, and request-response handling. It uses the Spring MVC development framework and adopts the powerful and easily integrable Apache Shiro security framework for access control. For user interaction, it borrows view patterns from mainstream Web-based resource management applications, providing a desktop explorer-like interface. The user interface utilizes Ajax asynchronous refresh technology to deliver a seamless browsing experience without page jumps. Bootstrap serves as the basic framework for front-end page design, enabling good display across devices with different resolutions and ensuring a secure and friendly front-end interface. To address the diversity and uncertainty of research data, MongoDB is used to store metadata information and structured data, with indexing established for file-based data. Cloud storage technology is employed for file-based data storage, facilitating maintenance and backup.

### 3.1 System Architecture Design

TeamDR is a B/S architecture application system that offers advantages such as easy deployment and minimal client-side load, while also featuring a typical 3-tier layered structure. All user-data interactions are established and conducted through the middle service layer, ensuring a low-coupling design philosophy. Figure 6 [Figure 6: see original paper] illustrates the system architecture of TeamDR, which can be logically divided into three layers: the storage layer, the service layer, and the application layer.

(1) **Storage Layer** The storage layer forms the foundation of system data storage management, where user research data, metadata, and system data are persisted. Depending on data type, structured data is stored using MongoDB database clusters, while unstructured file-based data is managed using cloud storage file systems. This approach fully leverages the advantages of each storage system to ensure efficient data storage and retrieval.

(2) **Service Layer** The service layer constitutes the system' s core business logic, implementing modules around main functions such as research data storage, browsing, and sharing. Data access uses the Spring Data MongoDB framework to support interaction between the service layer and MongoDB databases, while file system storage employs a three-layer Hash directory storage operation interface. The service layer functional modules consist of data resource management, data retrieval, data interfaces, and other utility classes. Data resource management primarily provides service support for user operations such as data uploading, browsing, and sharing, while ensuring coordination among sub-module functions. Data retrieval is mainly responsible for providing search functions based on data collection names, data types, and metadata. Data interfaces primarily provide open access to project group data resources for other applications, including functions such as obtaining resource lists under directories, retrieving data collection metadata, and accessing data resources, as well as restricted data collection creation, data uploading, and publishing.

(3) **Application Layer** The application layer comprises applications developed based on the functional interfaces provided by the service layer, ranging from data collection, data synchronization, and data query to data publishing, aiming to provide users with richer functional experiences. Currently, TeamDR is developing mobile data entry applications, research results experimental data association and search applications, data synchronization clients, and research data publishing integration.

### 3.2 Key Technical Designs

(1) **Three-Layer Hash Directory Storage Design** TeamDR employs a three-layer Hash directory storage scheme, establishing a root directory for each project group with data files stored under their respective root

directories. Since different hard disk formats support varying maximum numbers of files, and considering file indexing and reading efficiency, it is impractical to store all files within a project group in a single directory. Therefore, a directory indexing scheme based on Hash algorithms is established. For each file, the unique index generated by MongoDB serves as the file name, while original file attributes are extracted as metadata and stored in MongoDB. A three-layer Hash algorithm is applied to file names to calculate corresponding three-layer directory paths. The specific algorithm is as follows:

```
hash(filename)=HashCode(ObjectId)

path1=hash (filename)&255
path2=(hash(filename)>>8)&255
path3=(hash(filename)>>16)&255
path=Contact(path1,path2,path3)
```

Perform HashCode operation on the unique ObjectId generated by MongoDB to obtain the Hash value of the filename; Perform bitwise AND operation between the filename' s Hash value and 255 to obtain the first-level folder name; Right-shift the filename's Hash value by 8 bits, then perform bitwise AND with 255 to obtain the second-level folder name; Right-shift the filename' s Hash value by 16 bits, then perform bitwise AND with 255 to obtain the third-level folder name; Concatenate the addresses of the three-level directories to obtain the relative path address to the project group' s storage directory.

The three-layer Hash storage design prevents the problem of excessive file counts in single folders while incurring minimal performance overhead in file addressing. Since the three-layer Hash storage design is project group-based, it prevents chaotic and disorderly data storage across project groups, ensuring that project group file data is easy to maintain and backup.

(2) **HTTP Protocol-Based Large File Upload Design** The large file upload problem has long troubled many B/S architecture systems. Uploading through a single HTTP connection often results in connection timeout issues. Even modifying server-side timeout limits, errors during upload require restarting the entire process from scratch. Current solutions for large file uploads primarily rely on third-party components such as Java Applets or ActiveX upload controls, but these approaches are less than ideal. TeamDR effectively solves the large file upload problem by leveraging HTML5 technology, MD5 algorithms [13], and file splicing techniques. Figure 7 [Figure 7: see original paper] illustrates the flowchart of TeamDR' s file upload module.

The specific process for TeamDR' s large file upload handling is as follows: After users select upload files in the browser, the HTML5 File API obtains file size. Files smaller than 10MB are uploaded directly; For files larger than 10MB, the HTML5 Blob API calculates the file' s MD5 value and obtains file

type and size. Since calculating MD5 values for large files is time-consuming, TeamDR employs a simplified MD5 calculation method that extracts the first 64KB and last 64KB of data from the file, calculating the MD5 value of these 128KB. The specific calculation method is as follows:

```
S1 = file.slice(0,65536)
S2 = file.slice(file.size-65536,file.size)
MD5(file) = MD5(S1+S2)
```

This algorithm is faster and can uniquely identify files in combination with file type and size. Query the server for incomplete uploads using the MD5 value, file type, and file size. If the server finds that the file was previously uploaded but not completed, it retrieves the size of the uploaded portion and uses it as the starting point for continued upload. If the server finds no relevant file, it first stores the filename, file size, file type, and MD5 value on the server, generating a unique ID to facilitate resumption after upload failure. After the browser obtains the unique ID, it splits the file into segments no larger than 10MB and uploads them sequentially. Upon receiving file segments, the server locates the previously uploaded file using the unique ID and appends the newly uploaded segment to the file' s end.

(3) **HTML5-Based Document Online Preview Design** The document online preview function enables users to browse documents through browsers without installing corresponding software. TeamDR supports online preview of Word, PowerPoint, Excel, PDF, and other document formats. Currently popular document online preview solutions are mostly Flash-based, such as Docin. TeamDR employs HTML5 technology to display document content, eliminating Flash limitations and providing better page display effects. It also allows developers to manipulate its objects using JavaScript to draw graphics and images on web pages [14], whereas HTML requires Flash plugins for such functionality. TeamDR uses PDF.js, supported by Mozilla Labs, as the tool for document rendering in browsers, with secondary development integrated with TeamDR' s file upload module, data storage module, and others to provide smooth and feature-rich document online preview services. The specific process is shown in Figure 8 [Figure 8: see original paper].

After users upload documents through the TeamDR system, the database stores various document attributes. If the document is an Office document such as Word, Excel, or PowerPoint, TeamDR converts it to PDF format. TeamDR has set up a document conversion thread pool in the background, capable of converting multiple documents simultaneously. TeamDR currently supports preview of Office documents and PDF documents, including full-screen display and other functions. The page employs HTML5 technology, compatible with all major browsers, and eliminates dependence on Flash plugins, enabling normal display even in browsers with high security settings.

### 4.1 Implementation and Testing Environment

The development and testing of TeamDR' s cloud version primarily rely on the Haiyun Innovation Test Environment Platform of the Computer Network Information Center, Chinese Academy of Sciences. This platform features 1.2PB of scalable online storage capacity and network performance reaching 150Mb/s for single-connection WebService transmission. Users can conduct self-service application for services, fully enjoying the scalable and extensible characteristics of the cloud computing service environment, along with professional technical support from a 7×24 hour operations and maintenance team. Development based on this platform brings great convenience for TeamDR deployment and expansion, reduces costs, and provides end users with more stable and secure services. The software development is primarily based on languages such as Java, JavaScript, HTML, and CSS, using mainstream and stable development frameworks like Spring and Bootstrap.

### 4.2 Application Effectiveness

Following clarification of research team user data management requirements and research and implementation of key technologies, TeamDR' s cloud version officially launched in early August 2015, with the local version made available for download and service at the end of August. After product launch, preliminary promotion was conducted among specific scientific database construction units, and the system was introduced at scientific data conferences and exchange seminars for research institutions. By the end of October 2015, cloud-based registered users exceeded 90, local version downloads surpassed 80, and cloud-based project groups reached over 70. Current primary users and project groups are concentrated in chemistry, atmospheric environmental science, and biology disciplines. TeamDR training was provided to key groups including the Key Laboratory of Separation and Analysis Chemistry at Dalian Institute of Chemical Physics, Chinese Academy of Sciences, Biotechnology Division Group 1810, the Resource Geography and Land Resources Research Office of the Institute of Geographic Sciences and Natural Resources Research, and the Nuclear Magnetic Resonance Research Group of the Institute of Psychology, among others, with local versions formally deployed for use. Figure 9 [Figure 9: see original paper] displays the disciplinary distribution of cloud-based project groups.

Additionally, at the "2015 Scientific Data Conference," the most influential innovation conference in China' s scientific data field, the "Project Data Treasure"product (TeamDR) attracted the attention of numerous attending scientific data experts, scholars, and researchers during its first promotional appearance, ultimately receiving unanimous recognition from the conference' s expert committee and winning the "Best Demonstration Award." Researchers from various fields also provided numerous suggestions and ideas for TeamDR regarding data management workflows, security, and other aspects.

## 5 Conclusions and Future Outlook

With continuous scientific research and development across various disciplines, the scale of heterogeneous data—including experimental data, observational data, and literature data generated during research processes—is continuously expanding. How to continuously accumulate and effectively manage these research data assets and build high-quality project data repositories has become an increasingly important issue. TeamDR is dedicated to helping research teams effectively store and manage research process data, which will greatly contribute to solving these problems.

TeamDR still has deficiencies in systematic functional completeness and requires further improvement. Main areas include: interface integration between TeamDR and existing authoritative data repositories; real-time synchronization between TeamDR local data storage and cloud data; design of TeamDR mobile data collection and synchronization APP based on cameras, GPS positioning, and gravity sensing; and TeamDR data resource version control.

Through in-depth understanding of research user needs and continuous product function enhancement, TeamDR will become an indispensable tool for research team data management. We also hope that the development of TeamDR can expand new ideas and directions for scientific data repository construction practice.

## References

[1] Liu Feng, Zhang Xiaolin, Kong Lihua. Research Review on the Research Data Repositories [J]. New Technology of Library and Information Service, 2014(2): 25-31.
[2] Pampel H, Vierkant P, Scholze F, et al. Making Research Data Repositories Visible: The Re3data.org Registry[J]. PLoS One, 2013,8(11). DOI: 10.1371/journal.pone.0078080.
[3] TeamDR [EB/OL].[2015-07-13]. http://www.teamdr.cn.
[4] Ma Jianling, Cao Yuezhen. Research on the Development of Research Data Management Tools[J]. Research on Library Science, 2014(15): 40-47.
[5] CKAN [EB/OL]. [2015-11-08]. http://ckan.org/about/.
[6] UC3 Merritt [EB/OL]. [2015-11-08]. https://merritt.cdlib.org/.
[7] Figshare [EB/OL]. [2015-11-08]. http://figshare.com/.
[8] What is Scholar Sphere [EB/OL]. [2015-11-08]. https://scholarsphere.psu.edu/.
[9] Zhang Jing. Comparative Analysis of Figshare Platform and CNKI Academic Picture Library [J]. Science-Technology & Publication, 2015(1): 63-66.
[10] Online Team Document Library of Scientific Research[EB/OL]. [2015-11-08]. http://ddl.escience.cn/.
[11] Professional Data Collection and Management Tools: JianDaoyun [EB/OL]. [2015-11-08]. https://www.jiandaoyun.com/.
[12] Li Rui. Design and Implementation of Metadata Management Tool Based on Semantic Analysis [D]. Wuhan: Huazhong University of Science and

Technology, 2012.
[13] Mao Yi, Chen Na. Research and Improvement of MD5 Algorithm [J]. Computer Engineering, 2012, 38(24): 111-114.
[14] Xia Cuijuan, Zhang Yan. The New Chance of Library Mobile Reading Services: HTML5 & CSS3 [J]. New Technology of Library and Information Service, 2012(5): 16-25.

**Conflict of Interest Statement:** All authors declare no conflict of interest.

**Author Contributions Statement:**
Liu Feng: Proposed overall ideas and framework, participated in overall scheme design and content analysis, manuscript writing and revision;
Li Jianhui: Proposed paper improvement ideas, participated in content analysis and manuscript revision;
Zhang Jin: System implementation framework design and organization, key technology research organization, related chapter writing and revision;
Han Fang: Investigation of domestic and international application status, system positioning and design, related chapter writing and revision;
Liu Ang: System key implementation technology research, related chapter writing and revision.

**Supporting Data:** Supporting data [1-2] can be found in the journal' s online version at http://www.infotech.ac.cn; supporting data [3] is self-archived by the authors, E-mail: hanfang@cnic.cn.
[1] Liu Feng, Li Jianhui, Zhang Jin, Han Fang, Liu Ang. CloudService_url. Cloud version service related links.
[2] Liu Feng, Li Jianhui, Zhang Jin, Han Fang, Liu Ang. Standalone_url. Standalone version related links.
[3] Liu Feng, Li Jianhui, Zhang Jin, Han Fang, Liu Ang. DisciDistri.xls. Cloud version usage disciplinary distribution table.

**Received Date:** 2015-09-29
**Revised Date:** 2015-11-16

*Note: Figure translations are in progress. See original paper for figures.*