A cloud-based remote sensing data production system

Accepted Manuscript

A cloud-based remote sensing data production system

Jining Yan, Yan Ma, Lizhe Wang, Kim-Kwang Raymond Choo, Wei Jie

PII: S0167-739X(17)30303-5
DOI: http://dx.doi.org/10.1016/j.future.2017.02.044
Reference: FUTURE 3362

To appear in: *Future Generation Computer Systems*

Received date: 14 November 2016
Revised date: 23 January 2017
Accepted date: 24 February 2017

A Cloud-based Remote Sensing Data Production System

Jining Yan[a,b], Yan Ma[a], Lizhe Wang[c,a,*], Kim-Kwang Raymond Choo[d], Wei Jie[e]

[a] *Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing 100094, P. R. China*
[b] *University of Chinese Academy of Sciences, Beijing 100049, P. R. China*
[c] *School of Computer Science, China University of Geoscience, Wuhan 430074, China*
[d] *Department of Information Systems and Cyber Security, University of Texas at San Antonio, San Antonio, TX 78249-0631, USA*
[e] *School of Computing and Engineering, University of West London*

Abstract

The data processing capability of existing remote sensing system has not kept pace with the amount of data typically received and need to be processed. Existing product services are not capable of providing users with a variety of remote sensing data sources for selection, either. Therefore, in this paper, we present a product generation programme using multisource remote sensing data, across distributed data centers in a cloud environment, so as to compensate for the low productive efficiency, less types and simple services of the existing system. The programme adopts "master-slave" architecture. Specifically, the master center is mainly responsible for the production order receiving and parsing, as well as task and data scheduling, results feedback, and so on; the slave centers are the distributed remote sensing data centers, which storage one or more types of remote sensing data, and mainly responsible for production task execution. In general, each production task only runs on one data center, and the data scheduling among centers adopts a "minimum data transferring" strategy. The logical workflow of each production task is organized based on knowledge base, and then turned into the actual executed workflow by Kepler. In addition,

☆Fully documented templates are available in the elsarticle package on CTAN.

*Corresponding author
Email address: lizhe.wang@gmail.com (Lizhe Wang)

the scheduling strategy of each production task mainly depends on the Ganglia monitoring results, thus the computing resources can be allocated or expanded adaptively. Finally, we evaluated the proposed programme using test experiments performed at global, regional and local areas, and the results showed that our proposed cloud-based remote sensing production system could deal with massive remote sensing data and different products generating, as well as on-demand remote sensing computing and information service.

Keywords: remote sensing, Cloud Computing, Big Data
2016 MSC: 00-01, 99-00

---

1. Introduction

In recent times, there has been a sharp increase in the number of active and passive remote sensors being sent to space. Those sensors generally have characteristics such as hyper spectral, high spatial resolution, and high time resolution; thus, resulting in a significant increase in the volume, variety, velocity and veracity of data. Due to the richness of the data collected, their applications have also expanded. There are, however, limitations in existing remote sensing data management, processing, production and service pattern systems to adequately deal with the increased demands.

In data processing system, for example, the data processing capability has not kept pace with the amount of data typically received and need to be processed. Similar observation is reported by Quick and Choo [1], who remarked that "Existing forensic software solutions have evolved from the first generation of tools and are now beginning to address scalability issues. However, a gap remains in relation to analysis of large and disparate datasets. Every year the volume of data is increasing faster than the capability of processors and forensic tools can manage". For example, the amount of data received from a GF-2 satellite PSM1 sensor is approximately 1.5 TB per day, and the correction of an image (7000*7000*4 pixels) includes billions of floating point operations that require several minutes or even up to an hour to complete using a single workstation. It is clear that the processing time is not appropriate for various real-world applications.

In a data production system, existing remote sensing product services primarily support moderate-resolution imaging spectroradiometer (MODIS) [2] and Landsat [3, 4] production, and these services are not capable of providing users with a variety of remote sensing data sources for selection. If users need a multi-source remote sensing data product or another specific product, then they need to search for and download related data and use their own workstation. However, if the user's computing capability and/or knowledge on remote sensing (as well as their computer skills) are limited, then it would be challenging for such user to obtain the remote sensing products they need [5, 6]. Hence, this motivates the need for a well-designed platform with new data processing system architectures and service patterns to provide products and services for both skilled and unskilled researchers.

Cloud computing has been identified as a potential solution to address some of the big data challenges in remote sensing [7, 8, 9, 10] and big data computing [11, 12, 13, 14], such as allowing massive remote sensing data storage and complex data processing, providing on-demand services [15, 16, 17], and improving the timeliness of remote sensing information service delivery. For example, Lv, Hu, Zhong, Wu, Li and Zhao [18] demonstrated the feasibility of using MapReduce and parallel K-means clustering for remote sensing image storage and processing. Also using MapReduce, Almeer [19] built an experimental, high-performance cloud computing system in the Environmental Studies Center at the University of Qatar. Lin, Chung, Wang, Ku and Chou [20] proposed and implemented a framework desigend to store and process massive remote sensing images on a cloud computing platform. Similarly, Wang, Wang, Chen and Ni [10] compared the use of rapid processing methods and strategies for remote sensing images, using cloud computing and other computing paradigms. However, the focus of these studies is only on the storage and computational capabilities using cloud computing, rather than product generation and information service of remote sensing.

Above all, the existing remote sensing systems are facing the following major issues: (1) data processing capability has not kept pace with the amount of data typically received and need to be processed; (2) product services are not capable of providing users with a variety of remote sensing data sources for selection, and a well-designed platform is urgently needed to provide products and services for both skilled and unskilled researchers; (3) the current cloud-based remote sensing computing is less focused on product generation and information service. Therefore, in order to tackle these complex challenges, in this paper, we present a product generation programme using multisource remote sensing data, across distributed data centers in a cloud environment. This allows us to achieve massive data storage, high-performance computing, virtualization, elastic expansion, on-demand services and other cloud-inherent characteristics. We also provide an easy-to-use multi-source remote sensing data processing and production platform [21]. Finally, we demonstrate the utility of our approach using data processing and production generation experiments.

The remainder of this paper is organized as follows. In the next section, we provide an overview of the background and related work. Section 3 introduces the proposed cloud-based programme framework, system architectures, business logic and service patterns. Section 4 describes the experiments and study cases. Finally, in Section 5, we provide a summary and conclude the paper.

2. BACKGROUND AND RELATED WORK

This section briefly review remote sensing products and production system architectures, and related work.

2.1. Remote Sensing Products

Remote sensing products can be broadly categorized into fine processing products, inversion index products, and thematic products.

2.1.1. Fine Processing Products

Fine processing products mainly include geometric normalization products, radiometric normalization products, mosaic products, and fusion products.

- Geometric normalization products refer to geometric-registered image collection, where using geometric precision corrections, the images are turned into space seamless remote sensing products [22].
- Radiometric normalization products are quantitative remote sensing products (essentially, products obtained after radiometric cross-calibration, long time series radiometric normalization, atmospheric correction, etc) [23].
- Mosaic products can be explained simply as stitching two or more orthorectified satellite images with an overlapping area if the images from the satellite do not include atmospheric effects. To create a mosaic of two or more optical satellite remote sensing images, we geometrically correct the raw optical remote sensing dataset to a known map coordinates system (e.g., geographic coordinates system or projected coordinates system) as well as preprocessing the atmospheric corrections (e.g., image-based model, empirical line model and atmospheric condition model). However, it is important to consider apparent seasonal changes in order to mosaic images obtained from different seasons, due to the difficulties in acquiring high-resolution optical images in the same season for a number of reasons such as adverse weather conditions [24].
- Fusion products, one of the most commonly used remote sensing data, integrate information acquired with different spatial and spectral resolutions from sensors mounted on satellites, aircraft and ground platforms, and they contain more detailed information than each of the sources [25]. Fusion techniques are useful for a variety of applications, ranging from object detection, recognition, identification and classification, to object tracking, change detection, decision making, etc [26].

2.1.2. Inversion Index Products

Inversion index products generally refer to various inversion products of geophysical parameters, which reflect variation in characteristics of the land, sea and weather, such as Normalized Difference Vegetation Index (NDVI) [27], Normalized Difference Water Index (NDWI) [28], Normalized Difference Drought Index (NDDI)[29], Normalized Difference Build-up Index (NDBI)[30], and Normalized Difference Snow Index (NDSI)[31].

2.1.3. Thematic Products

Thematic products are application-oriented products or maps, such as thematic land-use and mineral thematic maps. Thematic products are generally obtained through remote sensing image interpretation and remote sensing inversion model, as well as expert knowledge.

In general, the above mentioned remote sensing products have upper and lower hierarchical relationship. That is to say, if we wish to obtain a remote sensing thematic product, the fine processing or inversion index products may be generated first. According to this hierarchical relationship, we build a knowledge base of remote sensing products and their corresponding production parameters, which guide the products generation.

2.2. Remote Sensing Data Production System

The remote sensing production system architecture can be broadly categorized into (1) personal computers (PCs) or a single workstation that acts as the remote sensing processing system, and (2) high-performance cluster-based remote sensing processing system. The stand-alone processing system uses the computational resources of single computer to independently perform remote sensing data processing and production process via human-computer interaction. As computing technologies develop over the years, the stand-alone processing system has evolved from big- and medium-sized computers specializing in remote sensing data processing to super-minicomputers specializing in remote sensing data processing to PC-based universal remote sensing data processing systems with multi-core processors to the current hyper-threading Graphics Processing Units (GPUs) [32, 33] universal remote sensing data processing systems [34].

Cluster-based remote sensing processing system generally consists of a number of identical PCs connected to multiple networks, tied together with channel bonding software. Thus, the networks act like one network running at many times the speed [35, 36]. Notable examples include “Pixel Factory” (France's massive remote sensing data processing platform), the Grid Processing on Demand (G-POD) European Space Agency project [37], Global Earth Observation System of Systems (GEOSS) [38], and Parallel Image Processing System (PIPS) of the Chinese Academy of Sciences Institute of Remote Sensing and Digital Earth [39, 40].

With a significant increase in remote sensing data, methods for remote sensing data processing and product generation are often limited by the sheer volume of data and computational demands that far exceed the capability of single workstations. Cluster-based High-performance computing (HPC) had been used to rapidly analze very large data sets ($>$ 10 Terabytes) can be rapidly analyzed; thus, in this paper, we build an HPC cluster with Open MPI in the cloud environment. This allows us to process global data sets, detect environmental change, and generate remote sensing products.

3. PROPOSED CLOUD-BASED REMOTE SENSING DATA PRODUCTION SYSTEM

3.1. Program Framework

Our proposed Cloud-based Remote Sensing (Cloud RS) programme adopts a master and multiple slaves' architecture (see Figure 1). The master center is mainly responsible for the production order receiving and parsing, task and data scheduling, results feedback, and so on. The slave centers are the distributed remote sensing data centers, which store one or more types of remote sensing data. These slave centers are also mainly responsible for production task exe-

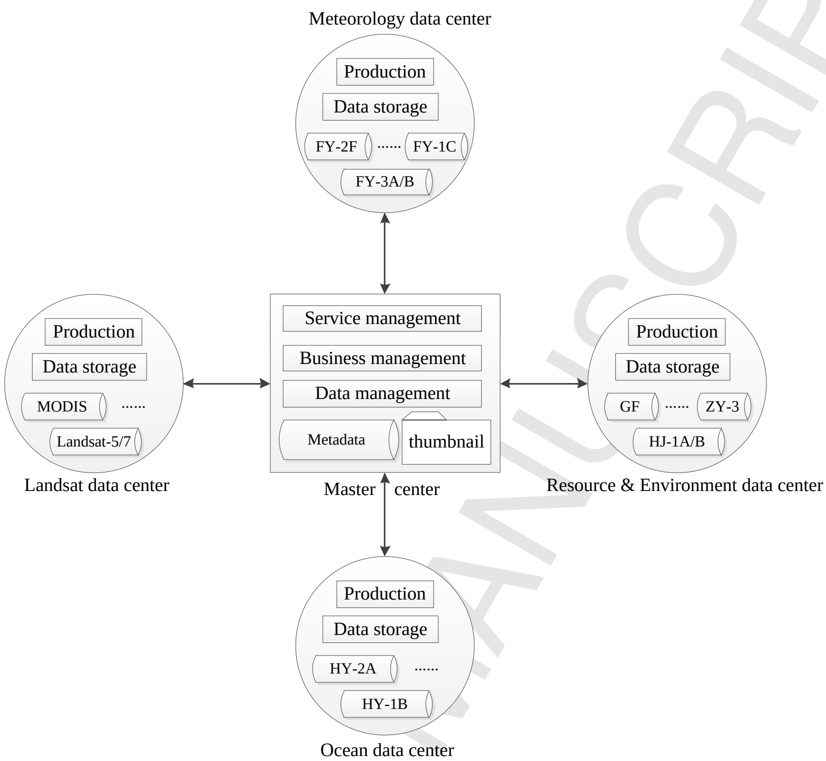


Fig. 1. Cloud RS framework

cution. In addition, the master center ingests metadata and thumbnail of the remote sensing data from each slave centers, and the service portal, including data service, production service and cloud storage service, distributed on the master center.

3.2. System Architecture

The architecture of remote sensing production system in the cloud environment consists of five layers, namely: resources, management, computing, business and service (see Figure 2).

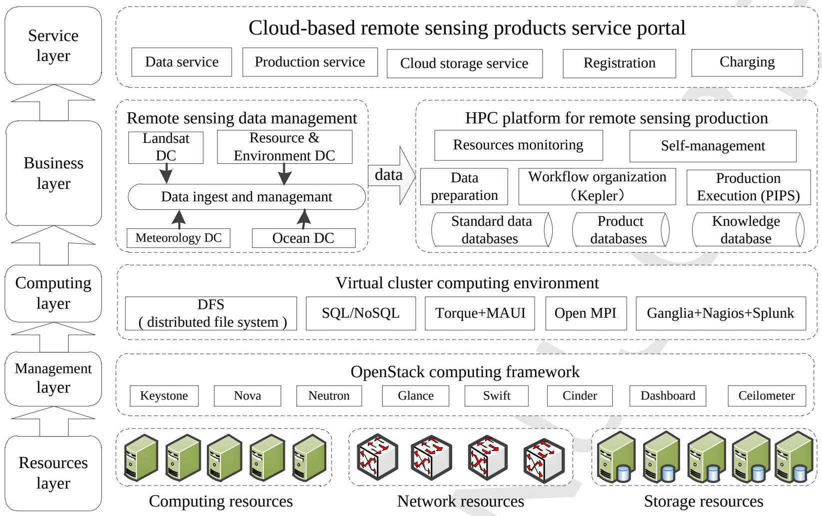


Fig. 2. Cloud RS system architecture

3.2.1. Resources Layer

A number of computing resources, network resources and storage resources connected by multiple networks are established as a pool of virtualized resources using the hypervisor (e.g., KVM, QEMU). Through unified management and scheduling, the virtualized pool can provide the standard and unified logical Central Processing Unit (CPU), logical memory, logical storage space and logical network interfaces. Thus, differences between multiple physical machines are minimal, and the virtualized resources used by all end users are consistent in the measurement, supply and scheduling aspects. End users just follow their own normal operations and make the necessary resource calls, without worrying about the distance and number of the physical devices [41, 42].

3.2.2. Management Layer

This layer mainly adopts the OpenStack cloud computing framework, and using its core components to manage all virtual resources. The OpenStack-Keystone component provides a single point of integration for managing authentication, authorization, and service catalog services. The OpenStack-Nova component creates virtual machines (VMs) according to the user's demand, managing the lifecycle of these VMs or virtual clusters. The OpenStack-Neutron component provides virtual network connection services, defining virtual networks, virtual subnets, virtual routers, etc. The OpenStack-Glance component enables users to discover, register and retrieve VM images, supporting a variety of VM image formats, as well as creating, uploading, editing and deleting VM images. The OpenStack-Swift component is used to implement object storage in the large-scale extensible system with built-in redundancy and a high fault-tolerant mechanism. The OpenStack-Cinder component primarily runs the virtual instances, providing block storage services. The OpenStack-Dashboard (also known as Horizon) is a web interface that enables cloud administrators and users to manage various OpenStack resources and services. The OpenStack-Ceilometer component is responsible for the telemetry service, collecting event and metering data by monitoring notifications sent from services, and creating alarms when collected data breaks a defined rule [43].

3.2.3. Computing Layer

This layer mainly provides a virtual cluster computing environment, such as remote sensing data storage container, cluster computing and scheduling capacity, computing environment monitoring and other services. Massive remote sensing data could be stored in the OpenStack-Swift multi-user object storage system. Each remote sensing image is stored as an object, and can be accessed only through the domain name address, which is assigned by the file identifier of each image. The metadata of the remote sensing image, thumb image and quick view image are stored in the NoSQL database MongoDB, a document-oriented storage non-relational database [44]. Millions of metadata can be located in one-tenth of a second using the powerful MongoDB search engine. The retrieval capability may be improved by building an index for each column independently via the MongoDB multi-level index [45]. It is worth noting that the NoSQL-based object storage mechanism eases management and retrieval of massive remote sensing data, but also avoids a single-point failure (commonly associated with traditional distributed file system).

Cluster scheduling of the virtual computing environment is controlled by the OpenStack-Nova and the storage capacity expansion of each node is managed by the block storage system OpenStack-Cinder. Block storage adds persistent storage to each virtual machine and provides volumes for instances.

It should also be noted that parallel computation in the virtual computing environment adopts transferring messages based on the Open Message Passing Interface (Open MPI) naked parallel programming model [46, 47]. The scheduling solution of the computing cluster is a combination of TORQUE and MAUI, so as to provide the resource allocation and scheduling ability for data processing and products produced in the virtual cluster. Furthermore, TORQUE and MAUI provide the scheduling strategy according to the monitoring software Ganglia, which is a scalable distributed system monitor tool for high-performance computing systems (e.g., clusters and grids).

The log and abnormal accurate monitoring includes Splunk-based log monitoring and mining, and the Nagios-based abnormal resource alarm. The specific Splunk-based log monitoring and mining process is as follows. The system log of distributed compute and storage nodes are recorded by the Syslogd server, before being sent to the collecting server syslog-ng at regular intervals. The collected system log is then classified and cleaned, so as to generate the log statements and conduct warning analysis. The Nagios-based abnormal resource alarm is raised under conditions of system resource overload, system performance deterioration or system outage. According to the alarm information, virtual instances would be adjusted and administrators could conduct system survey and analysis [48].

3.2.4. Business Layer

This layer mainly includes two parts, namely: (1) remote sensing data management across distributed data centers and (2) HPC platform for remote sensing production.

(1)Remote Sensing Data Management

Remote sensing data management across distributed data centers, mainly includes multi-source data ingest, metadata index and data retrieval. Data ingest from distributed data centers is mainly based on a crawler, which will launch itself at regular intervals, and push metadata to the master center. Then, the metadata will be indexed in the master center, based on the global subdivision mechanism. Finally, all of the indexed remote sensing data will be retrieved and located, providing data sources for the HPC production system.

(2)HPC Platform for Remote Sensing Production

The HPC platform for remote sensing production primarily realizes the service logic of remote senisng products generation. The core processing unit of the HPC platform is parallel image processing system (PIPS) [49, 50, 51], which was developed by the PIPS research group of the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences. PIPS is a large-scale, geographically distributed, and high-performance remote sensing data processing system. PIPS provides more than 100 kinds of serial and parallel remote sensing image processing algorithms, including level 0-2 remote sensing data pre-processing, fine processing products, inversion index products and thematic products generation, so as to provide production services for agriculture, forestry, mining, marine and other remote sensing industries.

The scheduling engine of PIPS adopts the Kepler scientific workflow [21, 52], and each production workflow mainly includes three parts: (1) data preparation (2) workflow organization and (3) production task execution.

Data preparation denotes the input data selection of each production order. In general, the production needed data is the standard remote sensing data, which is geometric and radiometric normalized original data, and their metadata are stored in the standard data database. It's important to note that the original data is level 1B or level 2 remote sensing data, and they all have been after inter-detector equalizations (sometimes referred to as radiometric correction) and systematic geometry correction. But as for some higher-level inversion

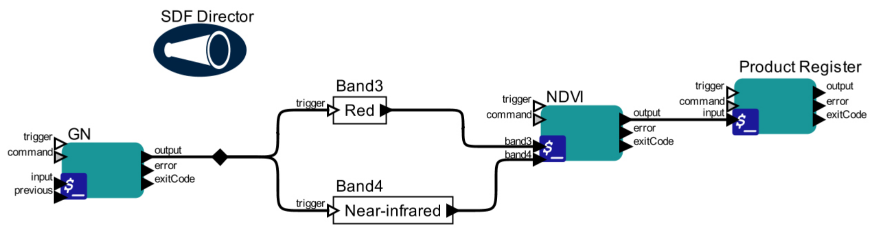


Fig. 3. The Kepler workflow of NDVI

index products, their required data may be some lower-level inversion products, and the data selection strategy may be very complex in this situation. For example, the required data sources of Net Primary Productivity (NPP) are Photosynthetic Active Radiation (PAR) and Leaf Area Index (LAI), but PAR and LAI still need some vegetation indexs (VIs), and the VIs require standard remote sensing data as input data sources, and this is very complex.

Workflow organization is essentially to determine the execution sequence of each process unit. In Kepler, the process unit is viewed as an Actor, and it was linked together by Relation and Link. Link determines the input and output of each Actor, and Relation shows the upper and lower hierarchical relationship of each Actor. For example, Figure 3 shows the Kepler model corresponding to the NDVI workflow. For the clarity of the paper, we omitted several steps from the original workflow. There are two input of the process unit NDVI calculation module, red and near-infrared bands, and output is a product. The upper module of NDVI is GN module, and lower module is product register. With the same procedure other vegetation indexs or image processing workflows can be implemented. However, there are so many kinds of remote sensing products that we cannot be able to list all of their Kepler workflows, or some users may want to define their production workflows. Therefore, we should provide a workflow auto-building capacity, then the knowledge database was built. And the knowledge base will be detailed in the next sub section.

After data preparation and workflow organization, the production tasks enter the implementation phase. In general, the production task runs only on one of data providing center, and the data scheduling adopts the the minimum data transferring strategy if one center cannot satisfy the data demand of the production task. As for the computing nodes selection on each data center, the total number is determined by the production task priority, or user determined, and which nodes assigned are determined by the scheduling policy of TORQUE and MAUI, as well as the resources monitoring results of Ganglia. In addition, in order to improve the robustness of the whole production system, self-management function is added.

3.2.5. Service Layer

Cloud services [53, 54] provided by the remote sensing cloud system include user registration and authentication services, user charging service, remote sensing data and product services, cloud storage service, etc. Remote sensing data denotes the original data, which is after inter-detector equalizations (sometimes referred to as radiometric correction) and systematic geometry correction. Remote sensing data service mainly include data retrieval, data order, data settlement, data download and unloading to cloud storage. Production service denotes the remote sensing data processing and generation services, including order submission, products download and unloading to cloud storage, et.al. Cloud storage service refers to the object storage that is built by the OpenStack-Swift component. The remote sensing data and products ordered by each user can be stored in their object storage space, and storage size can be dynamically grown or shrunk to meet user needs.

3.3. Knowledge Base and Inference Rules

Our knowledge base mainly includes three parts: (1) the upper and lower hierarchical relationship database (2) input/output database of every kind of remote sensing product and (3) inference rules for workflow organization.

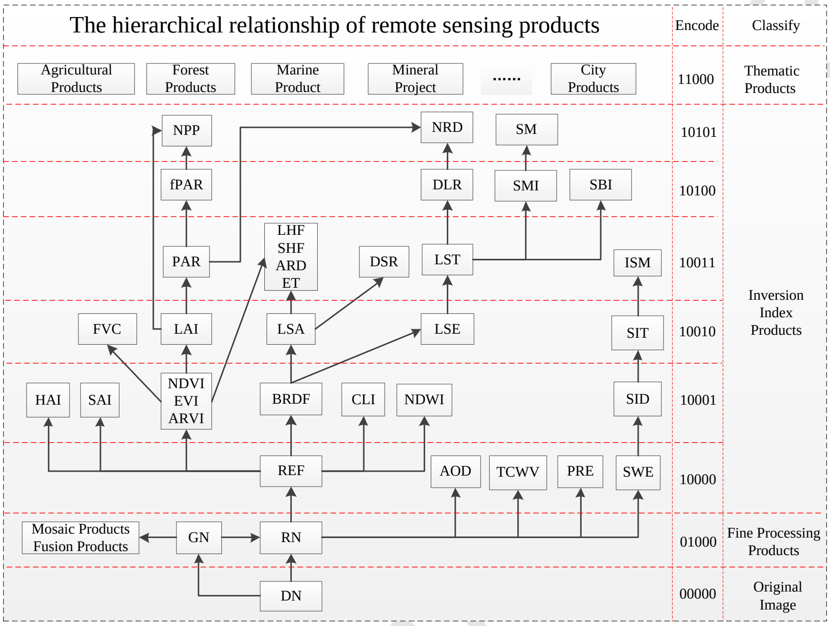


Fig. 4. The upper and lower hierarchical relationship of remote sensing products

3.3.1. The Upper and Lower Hierarchical Relationship Database

As mentioned before, remote sensing products mainly include three classifies, fine processing, inversion index and thematic products. As for each level, there are still some sub level products (Figure 4). In order to better use the hierarchical relationship of remote sensing products to organize Kepler production workflow, we encoded each product, and the coding rules are as following.

- The first layer coding includes two binary codes, '0' and '1', and '00', '01', '10', '11' denote the 'original image', 'fine processing products', 'inversion index products' and 'thematic products' separately.

- The second layer coding mainly aimed for the sub level products of each type. In order to as much as possible include all of sub level products, we chose three binary codes, and the concrete coding results are as shown in the right part of Figure 4.

- The third layer coding is for the product in each sub level, i.e., this layer coding is an ID of each product. This layer coding uses four binary codes.

Above all, as for each remote sensing product, the whole coding will include 9 binary codes, and the final results are as shown in Table 1.

3.3.2. Input/output Database of Every Kind of Remote Sensing Product

In addition to the upper and lower hierarchical relationship, we need to consider the input and output of each remote sensing product, so as to construct the production workflow. As mentioned above, remote sensing production needs one or more types of standard remote sensing data, and some inversion index products. Therefore, we build the input/output database for every kind of remote sensing product. Taking NPP as an example, the input/output database is shown in Figure 5

3.3.3. Inference Rules for Production Demand Data Selection

In order to prepare data sources for production, we established a set of inference rules, and it is as shown in Figure 6.

As shown in Figure 6, each production order need to parse inputParametersData, inputParametersProducts and auxilizryData three types parameters, and they are corresponding to the standard data, products and auxilizry data (Auxilizry data is only ready for some particular products). After parsing and reasoning, each type data name and location will be return, preparing for production scheduling.

3.3.4. Inference Rules for Workflow Organization

Production algorithms all have a one-to-one correspondence relationship with remote sensing products. Hence, the organization of remote sensing production workflow is based on the upper and lower hierarchical relationship database and input/output database. We also established inference rules in order to provide guidance for Kepler workflow self-organization (Figure 7).

```xml
<inputParametersData>
    <data>
        <inputdatatype>1</inputdatatype>
        <satellite>TERRA/AQUA </satellite >
        <sensor>MODIS</sensor>
        <productweight>0.9</productweight>
    </data>
</inputParametersData>
<inputParametersProducts>
    <product>
        <productTag>0</productTag>
        <productID>LAI</productID>
    </product>
    <product>
        <productTag>1</productTag>
        <productID>PAR</productID>
    </product>
    <product>
        <productTag>2</productTag>
        <productID>FPAR</productID>
    </product>
</inputParametersProducts>
```


Fig. 5. The input/output database for remote sensing products (NPP as an example)

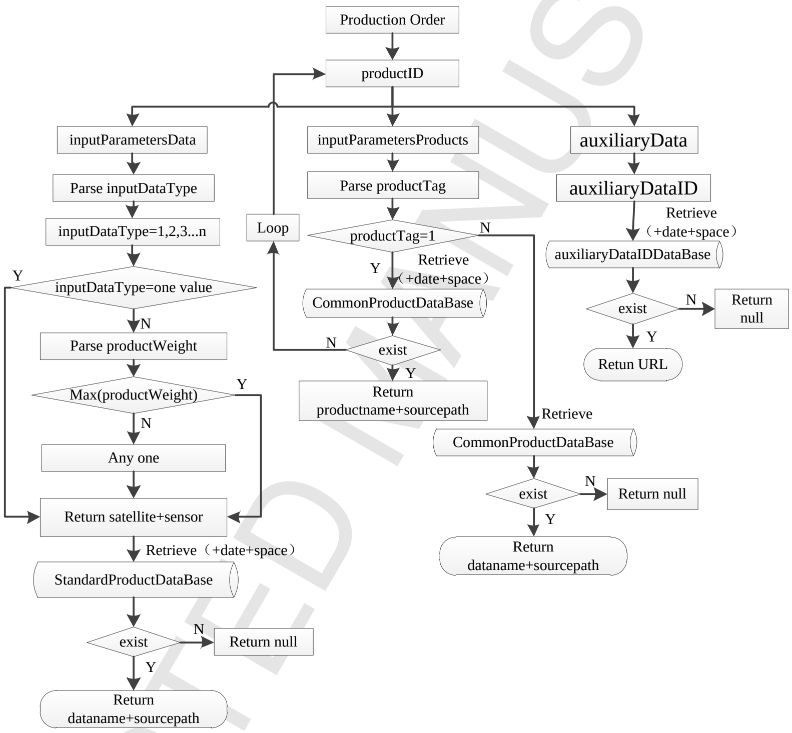


Fig. 6. Inference rules for production demand data selection

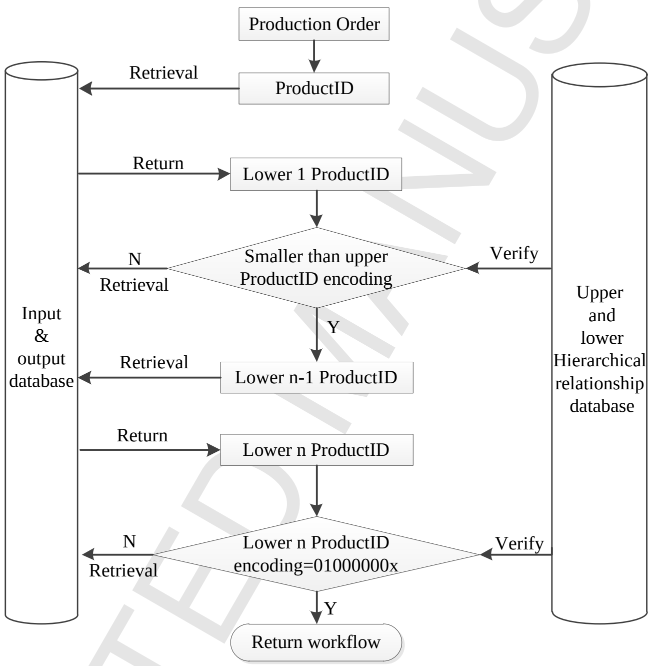


Fig. 7. Inference rules for workflow organization

3.4. Business Logic

The business logic of remote sensing production system mainly includes order submit, order analysis, completeness analysis, data preparation, products generation, products management and other related processes (see Figure 8)).

Order analysis business module analyzes the feasibility of user-submitted product orders, which depends upon whether there is potentially needed original remote sensing data in the database, whether there are potentially needed products in the products database, and whether there are potentially needed workflows in the workflows database. If the result of the order analysis is successful, then the remote sensing data preparation will arrive.

Remote sensing data preparation is aimed at determining and preparing the kinds and amounts of potentially needed original data according to the order analysis results. In essence, the original data refers to the preprocessed remote sensing data, which is available after radiometric correction and systematic geometry correction. It should be noted that data preparation is based on a good data management mechanism. In our system, the data management includes the original remote sensing database, database index, distributed storage strategy of multiple copies of remote sensing data, etc.

After data preparation, completeness analysis for the prepared remote sensing data is essential. Completeness analysis includes time range completeness analysis and space range completeness analysis. If the result of the completeness analysis is true, then the prepared remote sensing data will be transferred into the products generation module. But first, the prepared data should be standardized, including radiometric and geometric normalized. After normalization, the product orders will be processed. Remote sensing products generation business module includes workflow and corresponding algorithm selection, computing resource allocation, production task self-management and so on. Workflow and corresponding algorithm selection are determined by the products knowledge base, which have been detailed earlier. The computing resource allocation is mainly based on the workflow complexity index and real-time resource monitoring information. The workflow complexity index is calculated with the

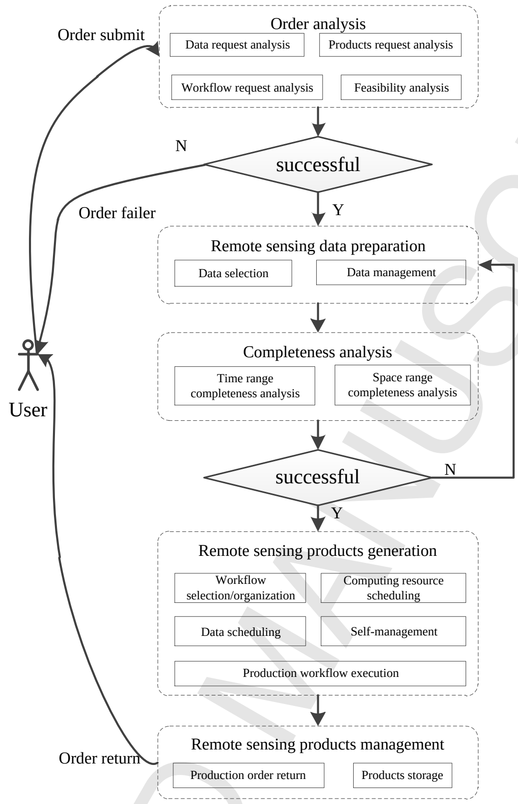


Fig. 8. Business logic of Cloud RS production system

input data volume, space and time complexities of the algorithms. Real-time resource monitoring information is obtained by Ganglia cluster-monitoring software. Production task self-management includes running task state monitoring, system log monitoring, fault-tolerance of the running job, etc.

Finally, the products will be checked-in and saved in the products management system. Products check-in refers to writing the metadata of the processed products into the products database for direct user product retrieval, so as to avoid repeated production processes.

3.5. Active Service Patterns

In general, the traditional service pattern of remote sensing production is the Build to Order (BTO) mode [55], sometimes referred to as Made to Order (MTO). This is a production approach whereby products are not built until a confirmed order is received, and the steps of its concrete realization are as following.

- By browsing the service portal, users can learn about remote sensing products information provided on the server side. Then, by the aid of a search engine, users' wanted products will be located, and it is generally accomplished based on the "product type + time range + space range" retrieval mode.
- After retrieving the desired products, users can select products and submit their production orders. If the remote sensing products have been produced, then the FTP URLs of the products will be returned. If not, the FTP URLs of the corresponding original data will be returned, and the production request will be generated and submitted.
- When the production system on the server side receives the production request, the production task will be executed and real-time feedback of the production schedule will be presented to the users. After the production task finishes, users will receive the FTP URL of the products for download to their computers, or transfer to their cloud storage system. In addition, the metadata of the products will be written into the product database for the next retrieve by other users.

BTO is the most appropriate approach for highly customized or low volume products. However, BTO is a passive service mode, and in today's competitive market, it is unable to meet market demand. Therefore, we should design an active service mode and recommend the remote sensing product service to users, based on their registration information and network behavior.

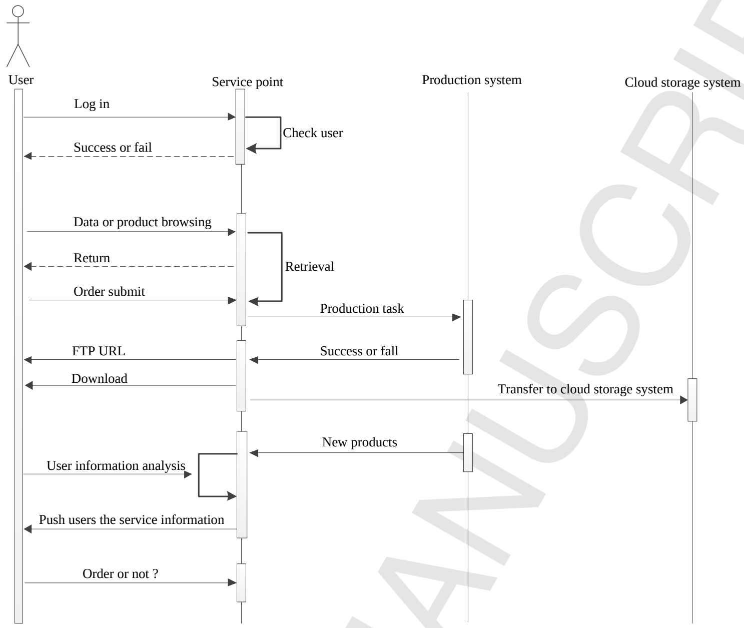


Fig. 9. UML service pattern for the Cloud RS production system

Therefore, based on the traditional BTO mode, we proposed an active service mode for remote sensing production, and its differences from BTO and main ideas will be detailed in the following. Based on user registration information, past web history and other log information, we may determine the types of users and topics that they already want. If user-related remote sensing data, products or other services realized in our platform, we would push the service information actively to users through their registered email or cellphone, thereby improving customer satisfaction for our remote sensing service.

In our cloud-based remote sensing producttion system, the implementation procedures of the active remote sensing production service can be generated using the Unified Modeling Language (UML), as shown in Figure 9).

As can be seen in Figure 9), the cloud-based service pattern differs from the BTO mode in two respects:

- Once users' requested products have been prepared, users can choose to save the products into their cloud storage, which is provided by the production system. The advantages of this pattern not only avoid the trouble caused by limited user storage capacity, but also improve the level of remote sensing data sharing.

- The individuation active recommendation information service enhances the utilization efficiency of the new generated products, while the active service pattern can provide personalized services for remote sensing users.

4. EXPERIMENT AND CASE STUDY

We evaluated the proposed programme using test experiments performed at global, regional and local areas.

4.1. Global Scale Remote Sensing Production

At the global scale remote sensing production, we chose the higher level inversion product NPP as an example. This is the quantity of carbon dioxide vegetation consumed during photosynthesis excluding the quantity of carbon dioxide the plants release during respiration (metabolizing sugars and starches for energy). In our test experiment, NPP was generated every 5 days in 2014, with 1 kilometer spatial resolution, and its input data sources are mainly MODIS L1B 1KM data from Landsat remote sensing data center. The production workflow is shown in Figure 10).

In order to realize the annual global scale NPP production, the volume of required MODIS L1B 1KM data is about 11 Terabyte (TB). Therefore, we provide a virtual multi-core cluster with 10 nodes. Each node is a x-large type of OpenStack instance, with 8 Virtual CPUs (VCPUs) and 16 GigaByte (GB) memory. The operating system was CentOS 6.5, the C++ compiler was GNU C++ Compiler with optimizing level O3, and the MPI implementation was MPICH. The total runtime was about 135 hours, and finally 74 global NPP products in 2014 were obtained. In order to examine the quality of the generated

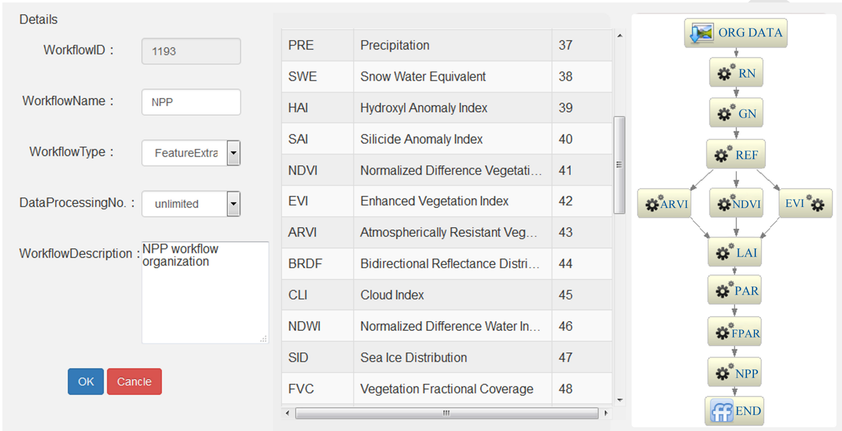


Fig. 10. NPP production workflow organization

NPP products, we selected 6 of them in different months, and they are shown in Figure 11).

As observed in Figure 11), in mid-latitudes, NPP is clearly tied to seasonal change, with productivity peaking in each hemispheres summer. The Boreal Forests of Canada and Russia, for example, experience high productivity in July and then a slow decline through fall and winter. Year-round, tropical forests in South America, Africa, Southeast Asia, and Indonesia have high productivity, not surprising with the abundant sunlight, warmth, and rainfall. This was well adaptive in natural animal growing, and also proved the practicability of our production system.

4.2. Region scale mosaic production

We selected 7 bands Landsat-TM image as the region scale mosaic data source, and the spatial scope is the north and east China (113°02′E- 123°32′ E, 30°45′N- 42°21′N). The total image number is 28, and the total volume is about 10 GB. The mosaic algorithm adopts parallel computing solution, and the total runtime with increasing numbers of virtual processors are as shown in Figure 12).

As can be seen in Figure 12), the total runtime decreases sharply when scale

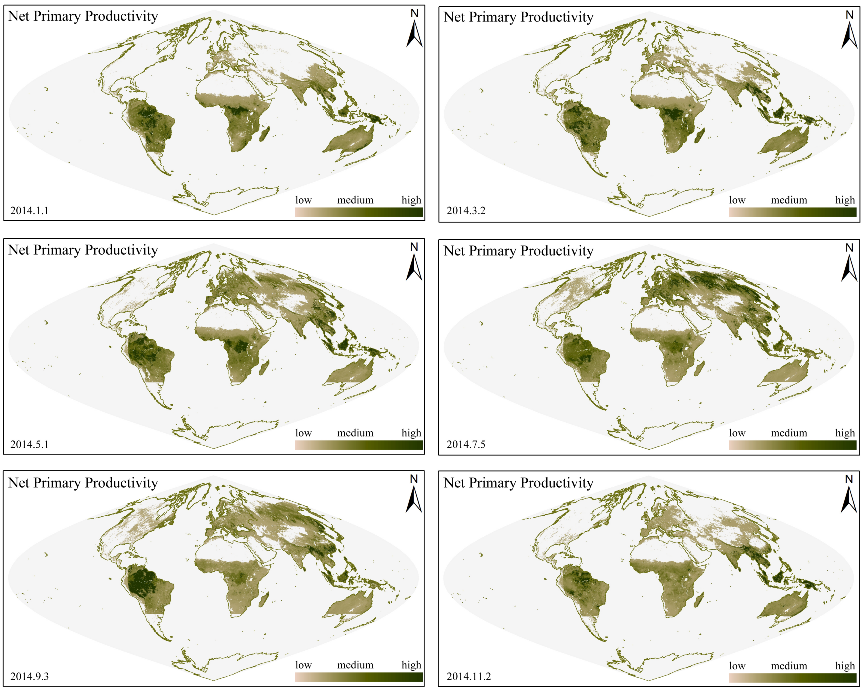


Fig. 11. The global NPP maps in different months

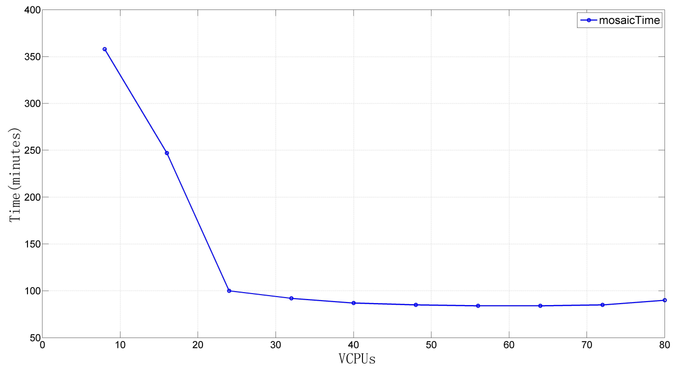


Fig. 12. Total runtime of mosaic production with scaling virtual processors

to less than 32 VCPUs. However, the decrease rate is much slower when scale from 40 VCPUs to more, and the runtime increases even up to 80 VCPUs. This is probably because the communication time between nodes occupies a great deal. The bands 4,3 and 2 false color composite image of the final region scale mosaic product is as shown in Figure 13).

In order to verify the effect of the mosaic production, we selected a 400 x 400 pixels region, and compared the visual effect before and after mosaicing. As can be seen in Figure 13), compared with the mosaicing before image, except for the partially color change, mosaic image can preserve the border structures efficiently. The color change may be because the color balance among all of the input images, during the process of mosaic, and this is difficult inevitable. Therefore, comprehensive considering the runtime and mosaic effect, our cloud-based production system is powerful.

4.3. Local scale change detection

Further more, in order to realize the time-series remote sensing production, we provide a data cube technology. But first, we should introduce the remote sensing data cube concept.

4.3.1. Remote sensing data cube

After geometric and radiometric normalization, remote sensing data are essentially becoming quantitative image 'tiles', which are two dimensional space seamless grids. Repeated observations of the same area at regular intervals produce a sequence of satellite image 'tiles'. If we collect these 'tiles' in time sequences covering the same ares of ground, it can be visualised as a three dimensional data set with the time axis as the third dimension. This is informally referred to as a 'cube'. The cube can be analysed and used to detect changes in the environment, so as to inform government about the effects of land degradation, flood damage and deforestation.

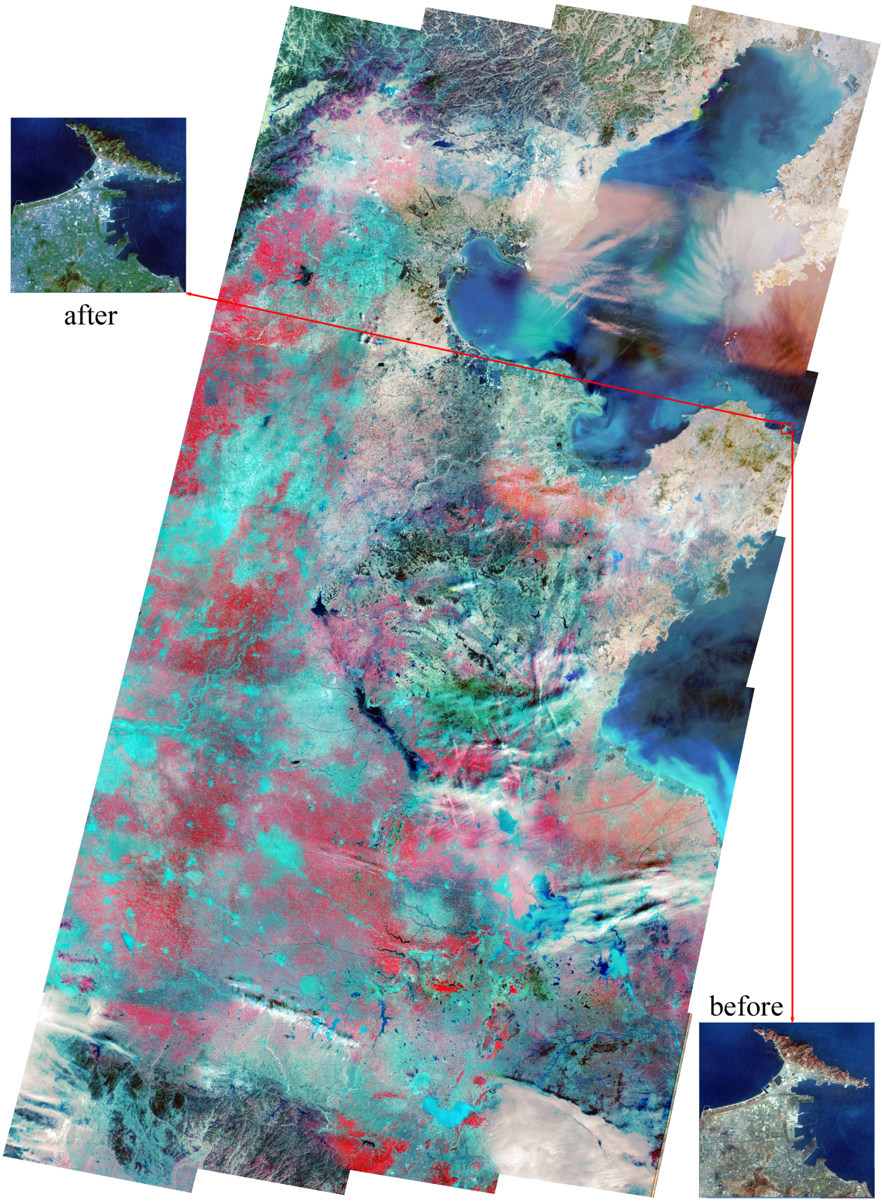


Fig. 13. Landsat-TM mosaic product of north and east China(R:band4, G:band3, B:band2)

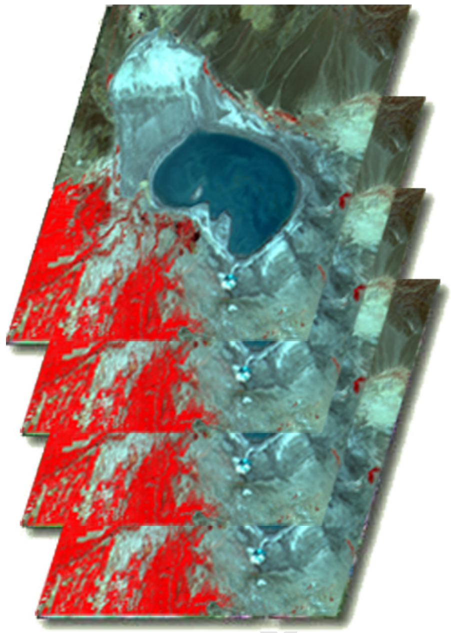


Fig. 14. HJ1A/B-CCD data cube of Aibi Lake (R:band4, G:band3, B:band2)

4.3.2. Local scale time-series production

At the local scale time-series production, we chose the Aibi Lake, which is in the northwest of China, as the study area. From September 18, 2008 to September 18, 2016, the HJ1A/B-CCD remote sensing data of that area, about total number of 800 original scenes, were ordered from China Center for Resources Satellite Data and Application (CRESDA). After geometric and radiometric normalization, a subset of each image area (2000 pixels x 2000 pixels x 4) was extracted, which covers the Aibi Lake. After weeding out the poor quality data, the left subset images were collected in time sequences, and the HJ1A/B-CCD data cube of Aibi Lake would be obtained (Figure 14)).

The prepared HJ1A/B-CCD data cube contains 421 scenes images, each image with 4 bands, 2000 x 2000 x 4 pixels, and the whole cube is about 26.4 GB. Using the NDWI production workflow in our system, with 10nodes virtual computing instances, after about 5 minutes, the final product was as shown in Figure 15).

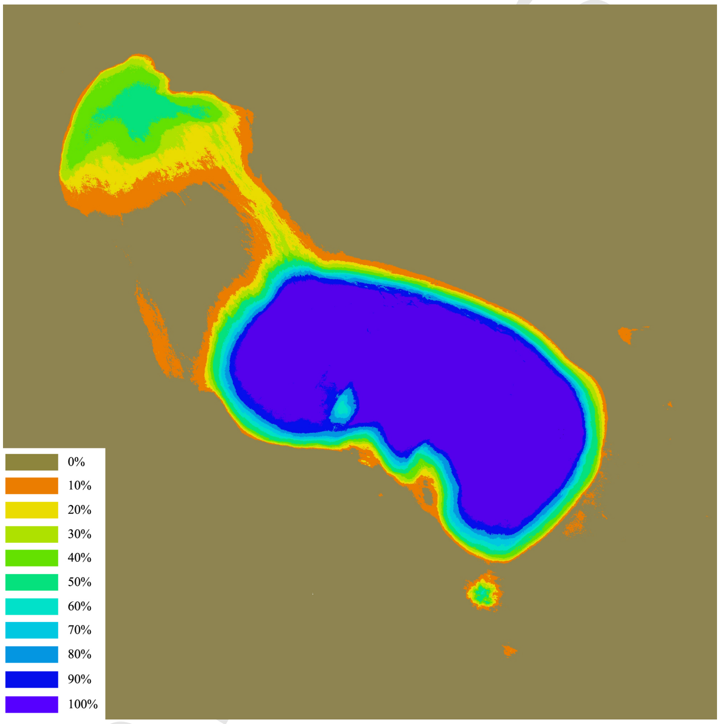


Fig. 15. The 9-year time series results of Aibi Lake (R:band4, G:band3, B:band2)

As observed in Figure 15), the blue area denotes the frequent water region of Aibi Lake, and the deeper blue color, the higher frequency. Some regions do not have persistent water, and this may be caused by seasonal variation. This is consistent with the natural laws, and further confirmed the practicability of our production system.

5. CONCLUSIONS

In this paper, we briefly reviewed remote sensing products and production system architectures, prior to presenting our cloud-based production system, across distributed data centers. We also described the system architectures, business logic and service patterns. Specifically, the proposed system has a five-layer architecture, which integrates several web and cloud computing technologies. Leveraging the benefits afforded by the use of cloud computing, we are able to support massive remote sensing data storage and processing. Each user can use the virtual machine and cloud storage conveniently; thus, reducing information technology resource costs. Moreover, the system adopts the individuation active recommendation information service.

Finally, findings from the test experiments (i.e. global scale production, regional scale mosaic, and local scale time-series analysis) demonstrated the powerful computing capabilities and advantages of our proposed programme. In other words, the proposed cloud-based remote sensing production system can deal with massive remote sensing data and different products generating, as well as on-demand remote sensing computing and information service. Future work includes extending, implementing and evaluating a prototype of the proposed system in a real-world scenario.

Acknowledgements

Dr. Lizhe Wang's work in the paper is funded by National Natural Science Foundation of China (No. 41471368).

References

[1] D. Quick, K.-K. R. Choo, Impacts of increasing volume of digital forensic data: A survey and future research challenges, Digital Investigation 11 (4) (2014) 273–294.

[2] A. M. Wilson, B. Parmentier, W. Jetz, Systematic land cover bias in collection 5 modis cloud mask and derived products global overview, Remote Sensing of Environment 141 (2014) 149–154.

[3] M. Hansen, A. Egorov, P. Potapov, S. Stehman, A. Tyukavina, S. Turubanova, D. Roy, S. Goetz, T. Loveland, J. Ju, et al., Monitoring conterminous united states (conus) land cover change with web-enabled landsat data (weld), Remote sensing of Environment 140 (2014) 466–484.

[4] D. P. Roy, J. Ju, K. Kline, P. L. Scaramuzza, V. Kovalskyy, M. Hansen, T. R. Loveland, E. Vermote, C. Zhang, Web-enabled landsat data (weld): Landsat etm+ composited mosaics of the conterminous united states, Remote Sensing of Environment 114 (1) (2010) 35–49.

[5] K. A. Al Mamun, M. Alhussein, K. Sailunaz, M. S. Islam, Cloud based framework for parkinson disease diagnosis and monitoring system for remote healthcare applications, Future Generation Computer Systems.

[6] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, Internet of things (iot): A vision, architectural elements, and future directions, Future Generation Computer Systems 29 (7) (2013) 1645–1660.

[7] L. Wang, M. Kunze, J. Tao, G. von Laszewski, Towards building a cloud for scientific applications, Advances in Engineering Software 42 (9) (2011) 714–722.

[8] L. Wang, C. Fu, Research advances in modern cyberinfrastructure, New Generation Computing 28 (2) (2010) 111–112.

[9] L. Wang, R. Ranjan, Processing distributed internet of things data in clouds., IEEE Cloud Computing 2 (1) (2015) 76–80.

[10] P. Wang, J. Wang, Y. Chen, G. Ni, Rapid processing of remote sensing images based on cloud computing, Future Generation Computer Systems 29 (8) (2013) 1963–1968.

[11] Z. Deng, Y. Hu, M. Zhu, X. Huang, B. Du, A scalable and fast OPTICS for clustering trajectory big data, Cluster Computing 18 (2) (2015) 549–562.

[12] Y. Wang, Z. Liu, H. Liao, C. Li, Improving the performance of GIS polygon overlay computation with mapreduce for spatial big data processing, Cluster Computing 18 (2) (2015) 507–516.

[13] Z. He, C. Wu, G. Liu, Z. Zheng, Y. Tian, Decomposition tree: a spatio-temporal indexing method for movement big data, Cluster Computing 18 (4) (2015) 1481–1492.

[14] Y. Chen, F. Li, J. Fan, Mining association rules in big data with NGEP, Cluster Computing 18 (2) (2015) 577–585.

[15] M. Dou, J. Chen, D. Chen, X. Chen, Z. Deng, X. Zhang, K. Xu, J. Wang, Modeling and simulation for natural disaster contingency planning driven by high-resolution remote sensing images, Future Generation Computer Systems 37 (2014) 367–377.

[16] P. Wang, J. Wang, Y. Chen, G. Ni, Rapid processing of remote sensing images based on cloud computing, Future Generation Computer Systems 29 (8) (2013) 1963–1968.

[17] L. Wang, T. Kurze, J. Tao, M. Kunze, G. von Laszewski, On-demand service hosting on production grid infrastructures, The Journal of Supercomputing 66 (3) (2013) 1178–1193.

[18] Z. Lv, Y. Hu, H. Zhong, J. Wu, B. Li, H. Zhao, Parallel k-means clustering of remote sensing images based on mapreduce, in: International Conference on Web Information Systems and Mining, Springer, 2010, pp. 162–170.

[19] M. H. Almeer, Cloud hadoop map reduce for remote sensing image analysis, Journal of Emerging Trends in Computing and Information Sciences 3 (4) (2012) 637–644.

[20] F.-C. Lin, L.-K. Chung, C.-J. Wang, W.-Y. Ku, T.-Y. Chou, Storage and processing of massive remote sensing images using a novel cloud computing platform, GIScience & Remote Sensing 50 (3) (2013) 322–336.

[21] S. Bendoukha, D. Moldt, H. Bendoukha, Building cloud-based scientific workflows made easy: A remote sensing application, in: International Conference of Design, User Experience, and Usability, Springer, 2015, pp. 277–288.

[22] J. Wang, Y. Ge, G. B. Heuvelink, C. Zhou, D. Brus, Effect of the sampling design of ground control points on the geometric correction of remotely sensed imagery, International Journal of Applied Earth Observation and Geoinformation 18 (2012) 91–100.

[23] L. Zhang, C. Wu, B. Du, Automatic radiometric normalization for multitemporal remote sensing imagery with iterative slow feature analysis, IEEE Transactions on Geoscience and Remote Sensing 52 (10) (2014) 6141–6155.

[24] J. Choi, H.-S. Jung, S.-H. Yun, An efficient mosaic algorithm considering seasonal variation: Application to kompsat-2 satellite images, Sensors 15 (3) (2015) 5649–5665.

[25] X. Li, L. Wang, On the study of fusion techniques for bad geological remote sensing image, J. Ambient Intelligence and Humanized Computing 6 (1) (2015) 141–149.

[26] J. Zhang, Multi-source remote sensing data fusion: status and trends, International Journal of Image and Data Fusion 1 (1) (2010) 5–24.

[27] B. Zheng, S. W. Myint, P. S. Thenkabail, R. M. Aggarwal, A support vector machine to identify irrigated crop types using time-series landsat ndvi data,

International Journal of Applied Earth Observation and Geoinformation 34 (2015) 103–112.

[28] H. Sun, X. Zhao, Y. Chen, A. Gong, J. Yang, A new agricultural drought monitoring index combining modis ndwi and day–night land surface temperatures: a case study in china, International Journal of Remote Sensing 34 (24) (2013) 8986–9001.

[29] A. Zhang, G. Jia, Monitoring meteorological drought in semiarid regions using multi-sensor microwave remote sensing data, Remote sensing of Environment 134 (2013) 12–23.

[30] K. Xue, Y. Zhang, H. Duan, R. Ma, S. Loiselle, M. Zhang, A remote sensing approach to estimate vertical profile classes of phytoplankton in a eutrophic lake, Remote Sensing 7 (11) (2015) 14403–14427.

[31] D. Roy, V. Kovalskyy, H. Zhang, E. Vermote, L. Yan, S. Kumar, A. Egorov, Characterization of landsat-7 to landsat-8 reflective wavelength and normalized difference vegetation index continuity, Remote Sensing of Environment.

[32] L. Wang, J. Tao, L. G. von, H. Marten, Multicores in cloud computing: research challenges for applications, Journal of computers 5 (6) (2010) 958–964.

[33] P. Liu, T. Yuan, Y. Ma, L. Wang, D. Liu, S. Yue, J. Kołodziej, Parallel processing of massive remote sensing images in a gpu architecture, Computing and Informatics 33 (1) (2014) 197–217.

[34] C. Song, Y. Li, B. Huang, A gpu-accelerated wavelet decompression system with spiht and reed-solomon decoding for satellite images, IEEE Journal of selected topics in applied earth observations and remote sensing 4 (3) (2011) 683–690.

[35] S. K. Garg, S. Versteeg, R. Buyya, A framework for ranking of cloud computing services, Future Generation Computer Systems 29 (4) (2013) 1012–1023.

[36] Y. Xia, X. Li, Z. Shan, Parallelized fusion on multisensor transportation data: a case study in cyberits, International Journal of Intelligent Systems 28 (6) (2013) 540–564.

[37] R. Cossu, E. Schoepfer, P. Bally, L. Fusco, Near real-time sar-based processing to support flood monitoring, Journal of Real-Time Image Processing 4 (3) (2009) 205–218.

[38] L. Ménard, I. Blanc, D. Beloin-Saint-Pierre, B. Gschwind, L. Wald, P. Blanc, T. Ranchin, R. Hischier, S. Gianfranceschi, S. Smolders, et al., Benefit of geoss interoperability in assessment of environmental impacts illustrated by the case of photovoltaic systems, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 5 (6) (2012) 1722–1728.

[39] L. Wang, Y. Ma, J. Yan, V. Chang, A. Y. Zomaya, pipscloud: High performance cloud computing for remote sensing big data management and processing, Future Generation Computer Systems.

[40] C. A. Lee, S. D. Gasster, A. Plaza, C.-I. Chang, B. Huang, Recent developments in high performance computing for remote sensing: A review, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 4 (3) (2011) 508–527.

[41] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, et al., A view of cloud computing, Communications of the ACM 53 (4) (2010) 50–58.

[42] H. Yao, C. Bai, D. Zeng, Q. Liang, Y. Fan, Migrate or not? exploring virtual machine migration in roadside cloudlet-based vehicular cloud, Con-

currency and Computation: Practice and Experience 27 (18) (2015) 5780–5792.

[43] OpenStack, http://www.openstack.org/.

[44] R. Giachetta, A framework for processing large scale geospatial and remote sensing data in mapreduce environment, Computers & Graphics 49 (2015) 37–46.

[45] C. Li, W. Yang, The distributed storage strategy research of remote sensing image based on mongo db, in: Earth Observation and Remote Sensing Applications (EORSA), 2014 3rd International Workshop on, IEEE, 2014, pp. 101–104.

[46] H. Gaojin, X. Wei, C. Luo, Q. Wu, N. Jing, A mpi-based parallel pyramid building algorithm for large-scale remote sensing images, in: Geoinformatics, 2015 23rd International Conference on, IEEE, 2015, pp. 1–4.

[47] Y. Ma, L. Wang, D. Liu, T. Yuan, P. Liu, W. Zhang, Distributed data structure templates for data-intensive remote sensing applications, Concurrency and Computation: Practice and Experience 25 (12) (2013) 1784–1797.

[48] M. Shi, R. Yuan, Mad: A monitor system for big data applications, in: International Conference on Intelligent Science and Big Data Engineering, Springer, 2015, pp. 308–315.

[49] L. Wang, Y. Ma, A. Y. Zomaya, R. Ranjan, D. Chen, A parallel file system with application-aware data layout policies for massive remote sensing image processing in digital earth, IEEE Transactions on Parallel and Distributed Systems 26 (6) (2015) 1497–1508.

[50] Y. Ma, L. Wang, P. Liu, R. Ranjan, Towards building a data-intensive index for big data computing–a case study of remote sensing data processing, Information Sciences 319 (2015) 171–188.

[51] Y. Ma, L. Wang, A. Y. Zomaya, D. Chen, R. Ranjan, Task-tree based large-scale mosaicking for massive remote sensed imageries with dynamic dag scheduling, IEEE Transactions on Parallel and Distributed Systems 25 (8) (2014) 2126–2137.

[52] P. Yue, L. Di, Y. Wei, W. Han, Intelligent services for discovery of complex geospatial features from remote sensing imagery, ISPRS journal of photogrammetry and remote sensing 83 (2013) 151–164.

[53] W. Ren, L. Zeng, R. Liu, C. Cheng, F2AC: A lightweight, fine-grained, and flexible access control scheme for file storage in mobile cloud computing, Mobile Information Systems 2016 (2016) 5232846:1–5232846:9.

[54] W. Ren, uleepp: An ultra-lightweight energy-efficient and privacy-protected scheme for pervasive and mobile wbsn-cloud communications, Ad Hoc & Sensor Wireless Networks 27 (3-4) (2015) 173–195.

[55] S. A. Mansouri, D. Gallear, M. H. Askariazad, Decision support for build-to-order supply chain management through multiobjective optimization, International Journal of Production Economics 135 (1) (2012) 24–36.

Table 1
The coding results of remote sensing products (part).

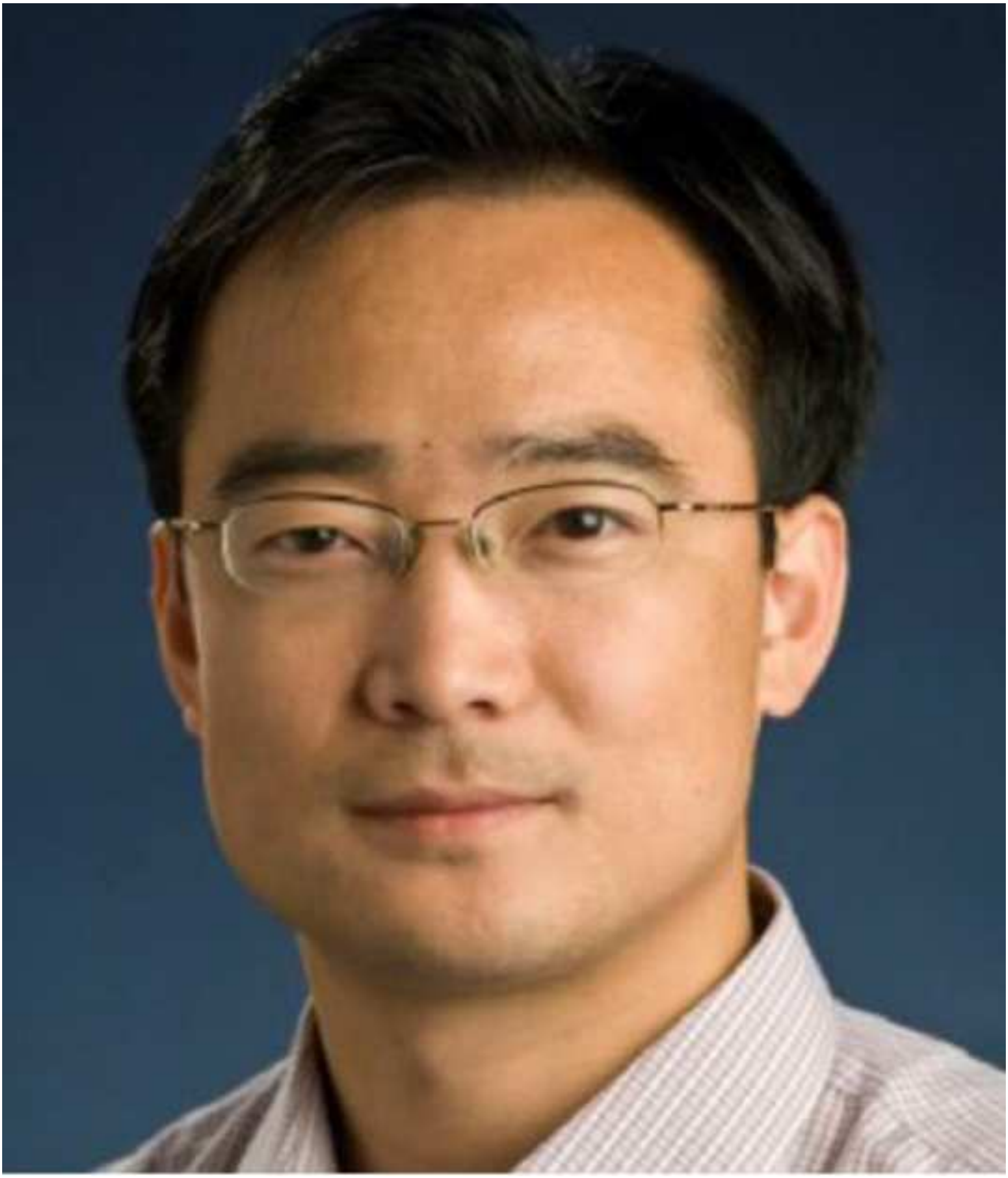
product name	short name	coding
Digital Numbe	DN	000000000
Radiometric Normalized	RN	010000000
Geometric Normalized	GN	010000001
Mosaic Products	Mosaic Products	010000010
Fusion Products	Fusion Products	010000011
Surface Reflectance	REF	100000000
Silicide Anomaly Index	SAI	100010001
Normalized Difference Vegetation Index	NDVI	100010010
Enhanced Vegetation Index	EVI	100010011
Atmospherically Resistant Vegetation Index	ARVI	100010100
Bidirectional Reflectance Distribution Function	BRDF	100010101
Normalized Difference Water Index	NDWI	100010111
Sea Ice Distribution	SID	100011000
Vegetation Fractional Coverage	FVC	100100000
Leaf Area Index	LAI	100100001
Land Surface Albedo	LSA	100100010
Photosynthetically Active Radiation	PAR	100110000
Evapotranspire	ET	100110001
Aerodynamic Roughness Length	ARD	100110010
Sensible Heat Flux	SHF	100110011
Downward Shortwave Radiation	DSR	100110101
Land Surface Temperature	LST	100110110
Ice Snow Mass Change	ISM	100110111
Fraction Of Photosynthetically Active Radiation	fPAR	101000000
Soil Moisture Index	SMI	101000010
Soil Brightness Index	SBI	101000011
Net Primary Productivity	NPP	101010000

Jining Yan is a Ph.D. student in the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences. His main research interests include remote sensing data processing and information service, high performance and cloud computing in remote sensing.

Lizhe Wang is a "ChuTian" Chair Professor at the School of Computer Science, China University of Geosciences (CUG), and a Professor at the Institute of Remote Sensing & Digital Earth, Chinese Academy of Sciences (CAS). Prof. Wang received his B.E. & M.E from Tsinghua University and Doctor of Eng. from University Karlsruhe (Magna Cum Laude), Germany. Prof. Wang is an IET Fellow and Fellow of the British Computer Society. Prof. Wang is an Associate Editor for IEEE T. Computers, IEEE T. on Cloud Computing, and IEEE T. on Sustainable Computing. His main research interests include HPC, e-Science, and remote sensing image processing.

Kim-Kwang Raymond Choo is an associated professor of Cyber Security and Digital Forensics in University of South Australia and Visiting Scholar at INTERPOL Global Complex for Innovation (IGCI). He was Named one of 100 Emerging Leaders (Innovation) in Weekend Australian Magazine/Microsoft's Next 100 series - 2009; and recipient of 2010 ACT Pearcey Award, 2010 Consensus IT Professional ward, 2009 Fulbright Scholarship, 2008 Australia Day Achievement Medallion, & British Computer Society's Wilkes Award - Computer Journal (2007)'s best paper, Highly Commended in 2014 ¡°Best Chapter in a Book¡± category by ANZPAA NIFS, and ESORICS 2015 Best Paper Award. He has authored 2 books, 9 refereed monographs, 166 refereed book chapters and articles.

Dr. Wei Jie is a Senior Lecturer at Computing, School of Computing and Technology, univeristy of west London. He is a Senior Member of The Institute of Electrical and Electronics Engineers (IEEE) and a Fellow of the Higher Education Academy (FHEA). Dr. Wei Jie has been active in a broad spectrum of areas in parallel and distributed computing, in particular, grid and cloud
computing, computing security technologies, e-science and e-research. Dr. Jie has been actively involved in professional services. He is the General Chair of the IEEE workshop on Security in e-Science and e-Research, and has served as Program Committee member for more than 40 international conferences and workshops. Dr. Wei Jie has published approximately 50 papers in nternational journal and conferences and has edited three books.



*Biographies (Photograph)
Click here to download high resolution image

(1) A cloud-based automatic remote sensing production programme is proposed

(2) A “master-slave” data processing architecture across large-scale distributed data centers.

(3) Test cases in real production system validates the practicability and capabilities of the proposed programme

ACCEPTED MANUSCRIPT