Network Research Workshop
Proceedings of the Asia-Pacific Advanced Network 2014 v. 38, p. 1-11.
http://dx.doi.org/10.7125/APAN.38.1 ISSN 2227-3026

Ceph as WAN Filesystem – Performance and Feasibility Study through Simulation

Ming-Tat Wong, Mohd Bazli Ab Karim, Jing-Yuan Luke, and Hong Ong

MIMOS Berhad, Technology Park Malaysia, 57000 Kuala Lumpur, Malaysia

E-Mails: {mt.wong, bazli.abkarim, jyluke, hh.ong}@mimos.my

Tel.: +60 3 8995 5000

Abstract: Recent development in object based distributed file systems (DFS) such as Ceph, GlusterFS as well as the more established ones like Lustre, GPFS, etc. have presented new opportunities to setup next generation of storage infrastructure for cloud computing, big data, and Internet of Things (IoT). However, existing DFSs are typically deployed to Local Area Network (LAN) and generally used for high-performance computing. Extending these DFSs into geographically distributed sites such as Campus Area Network (CAN) and Wide Area Network (WAN) for enterprise applications presents completely different set of challenges and issues. Unlike most implementations that choose a traditional multi sites deployment, i.e., each site implements a virtual storage (via LAN) and links through RESTful APIs (via WAN), we attempt to create a single virtual storage over WAN using Ceph. In this paper, we demonstrate that a properly designed and configured virtualized environment is a valuable tool for researchers to simulate a distributed files system over WAN without an actual physical environment. By following a few guidelines, the read and write performance results in a simulated environment can indicate the trending of the read and write performance in the actual physical environment. This implies that the storage design can be verified prior to actual deployment and establish a performance baseline. An obvious benefit is the initial investment of a storage solution is lower. Furthermore, this paper discuss about the challenges of setting up such environment, the feasibility of using Ceph as a single virtual store, and some possible future works.

Keywords: distributed file system; software defined storage; wide area network; virtualized environment.

1. Introduction

The arrival of cloud computing as well as big data paradigms had given the hope for more scalability, flexibility as well as redundancy to computing and storage resources and new methods in data analysis. However, both cloud computing and big data analysis have something in common, the success of their deployment in various areas depends on choosing the right storage technology or infrastructure [1, 9, 10].

New object based distributed file systems such as Ceph, GlusterFS as well as the more matured solutions like Lustre, GPFS and so on have presented a unique opportunity to serve both the areas mentioned above as they can be deployed in a relatively cost effectively compared to some of the more traditional (and proprietary) Storage Area Network (SAN) and Network-Attached Storage (NAS) based storage solutions [8].

While it is common to deploy such distributed file systems in a LAN or a single datacenter scenario, expanding such storage infrastructure into geographically distributed sites such as CAN or WAN presents totally different challenges and issues. While most would choose a traditional multi sites implementation, for example using de-duplication, remote replication methods and so on, the authors explore an alternate solution to create a single virtual storage pool using Ceph over a WAN instead. As a test environment of data transmission in WAN, a total of three datacenters which consists of two datacenters in headquarter (TPM-DC1 and TPM-DC2) and one datacenter in a branch (KHTP-DC3), where these datacenters are separated roughly over 350km are used.

At the conference PRAGMA 26, authors demonstrated the process of obtaining data and measurements over WAN in these datacenters by using Ceph virtual storage pool. To simulate this physical environment consists of three datacenters in virtualized environment, a combination of open source solution such as QEMU/KVM [11, 12], Open vSwitch (OVS) [13] and TC are used to achieve what is observed in the actual physical environment as 250Mbps, 14ms latency and approximately 1ms jitter WAN connection as discussed in Section 3.

As this simulator can be used to verify the design of storage topology of the physical environment and produce data and measurements without implementing on a physical environment, the cost in machine purchasing and the overhead of human resource is reduced. In short, this research provides an alternative preview of pre-implemented physical environment.

In Section 4, results from the virtualized environment compared to those obtain from the actual physical setup will be presented and discussed together with some issues and possible solutions in the process of creating the virtualized environment. Section 5 provides a conclusion and discusses potential future works.

2. Overview of Storage Technologies

The complexity of data management grows as the demand of large storage capacity has increased drastically as the growth of human population and the use of Internet has reach more ends of the world. In order to expand the capacity of storage, various techniques are introduced to optimize storage.

2.1. Block storage and object based distributed file systems

In order to allocate data into scalable storage, there are two major techniques available which are block storage and object based distributed file systems. Block storage is typically used in SAN environment where data is stored in evenly-sized blocks, which is also known as volume. Each volume is abstracted from the storage and then being exported as a Logical Unit Number (LUN) to the server. For server to access the storage, Fibre Channel (FC), Fibre Channel over Ethernet (FCoE) or iSCSI protocols is used. Block storage works well on enterprise application such as file systems and databases.

As database currently is optimized on block storage, block storage is widely used in most enterprise services such as structure aware file service, e-mail, and business intelligent service [16]. However, with the increasing benefits of the flexibility and scalability of cloud computing, the demand of deploying database on object based distributed file system has increased due to the nature of its programmability in cloud computing environment[8]. Moreover, due to the configuration complexity and the deployment cost of block storage, object based distributed file systems such as Ceph and Gluster are used as an alternative storage solution [2, 3].

In object based distributed file systems, each data is stored as an object. An object consists of data, metadata, and a globally unique identifier (UUID). The advantages of object based distributed file systems over block storage are less deployment cost, more scalable and its self-healing ability.

In the conference PRAGMA 25 [14], two popular object based distributed file systems, Ceph and Gluster were chosen to compare the speed of sequential write and sequential read. According to the result obtained, Ceph gives higher speed for both sequential write and sequential read. Hence, Ceph is used in this research.

2.2. Architecture of Ceph

In Ceph, objects are stored in object-based storage devices (OSDs) which are referred to a storage device on a server such as a single disk or a volume as in a RAID [4, 5]. The overall health and status of a Ceph cluster is maintained by the monitors (MONs) which manage a number of information in the form of maps. To access storage in Ceph, there are three methods

available which are CephFS, Rados Block Device (RBD) and Rados Gateway (RGW). When CephFS is used, Metadata servers (MDSs) are required to provide metadata service to clients. CRUSH is a scalable pseudo-random data distribution function. In Ceph, CRUSH Map contains a map with the information of rules or policies on placing and replicating of data. When an OSD fails, CRUSH map is used to locate replicated data in order to recover the lost data.

3. From Physical Environment to Virtualized Environment

As discussed earlier, an actual physical setup was already deployed as per Figure 1 and initial measurements and benchmarks had been obtained [15]. However, as this setup is also sharing the authors institute's WAN bandwidth, further benchmark activities to identify areas of optimization would be difficult. Therefore the idea of simulating this particular setup in a virtual environment is proposed.

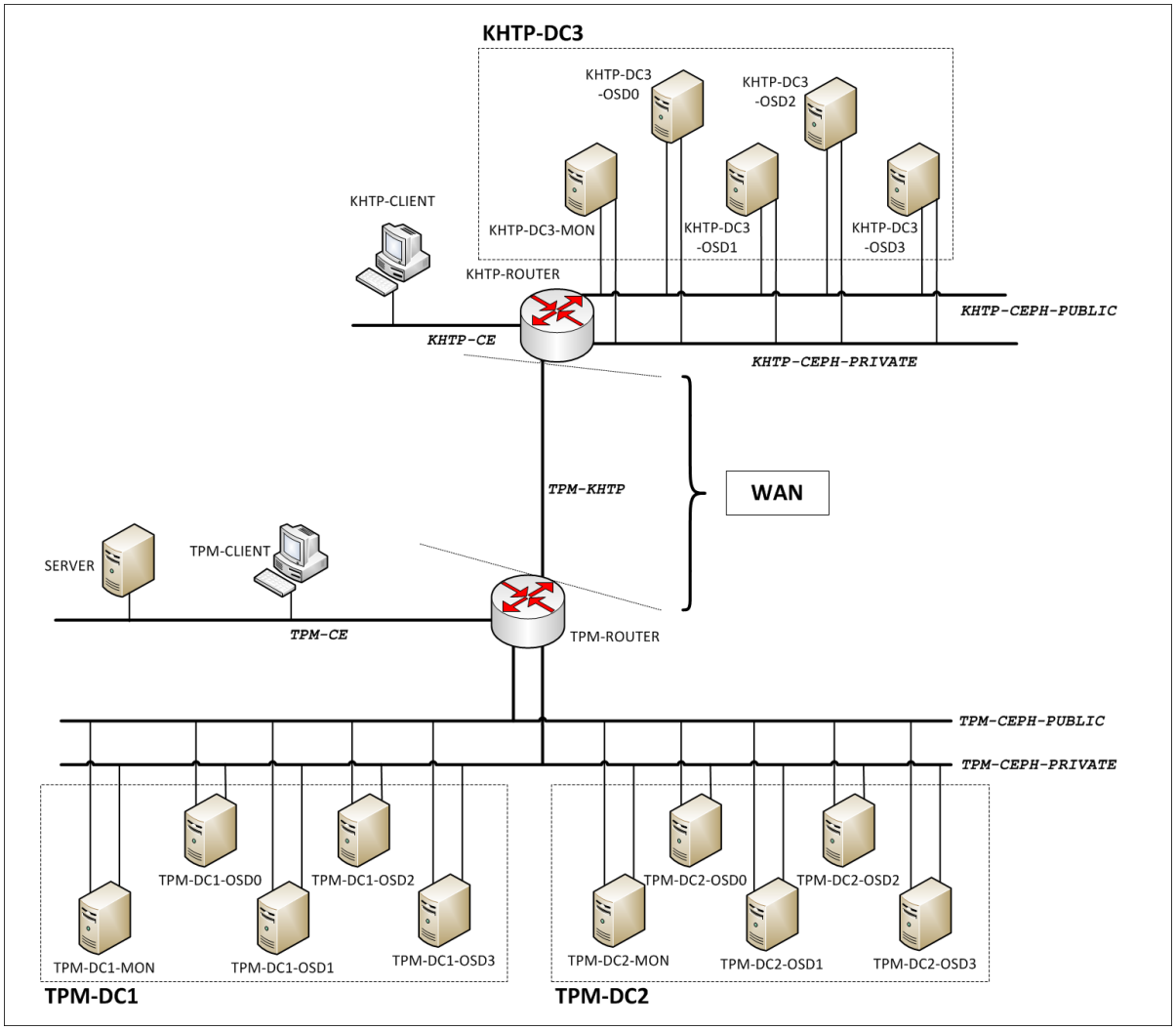

Figure 1. Physical environment topology.

3.1. Gathering Measurements from the Physical Environment

In a physical environment, three measurements, namely the throughput of the bandwidth, latency, and jitter, are tested on three data centers. Each measurement is tested on the connection between TPM-DC1 & TPM-DC2, TPM-DC1 & KHTP-DC3, and also TPM-DC2 & KHTP-DC3.

In order to obtain the measurements between the connections of these data centers, iperf, a traffic generator tool is used to test the throughput of the bandwidth while ping is used to determine the latency and jitter. According to the result obtained, the measurements of the connection between the data centers are shown as the table below:

TPM-DC1 & TPM-DC2 (1Gbps)	Bandwidth throughput	86%
	Latency	0.490ms
	Jitter	0.136ms
TPM-DC1 & KHTP-DC3 (250Mbps)	Bandwidth throughput	96%
	Latency	13.491ms
	Jitter	0.684ms
TPM-DC2 & KHTP-DC3 (250Mbps)	Bandwidth throughput	96%
	Latency	14.004ms
	Jitter	1.079ms

Table 1. Measurements obtained from physical environment.

3.2. Virtualized Environment Setup

A host machine is set up for virtualized environment which has following specification:

CPU	2 x Intel Xeon E5-2630 CPU with clock speed 2.3 GHz, 6 cores with hyper threading.
RAM	96GB DDR3
HDD	7 x 300GB 10000RPM SAS 4 disk in RAID 5 as OS Remaining 3 disk are setup individually, each storing different VM images for different datacenters
Operating System	Ubuntu 12.04.4 LTS Precise Pangolin with kernel 3.11.0-20
Hypervisor	QEMU/KVM Version 1.5.0
Open vSwitch (OVS)	Version 2.0.0
Traffic Control (TC)	Included in iproute version 20111117-1

Ceph	0.72.2
IOzone	397-2

Table 2. Specifications of host.

3.2.1. Preparing the Virtual Network for the Virtualized Environment

To simulate virtualized environment from physical environment, KVM is used as hypervisor. As the virtual switch in KVM does not provide functions such as traffic shaping and Quality of Service (QoS), OVS is used to replace this virtual switch as OVS provides functionalities to support the configuration needed in simulating physical environment in virtualized environment.

From the measurements obtained on physical environment, the bandwidth between TPM-DC1 & TPM-DC2 is 1Gbps, whereas the bandwidth between TPM-DC1 & KHTP-DC3 and TPM-DC2 & KHTP-DC3 are 250Mbps. In order to simulate the bandwidth as such on virtualized environment, the default configuration of OVS is configured. Iperf is then used to obtain the bandwidth of the connections between data centers on virtualized environment.

	Bandwidth between data centers		
	TPM-DC1 & TPM-DC2	TPM-DC1 & KHTP-DC3	TPM-DC2 & KHTP-DC3
Physical environment	1 Gbps	250 Mbps	250 Mbps
Default OVS Configuration on virtualized environment	12.1 Gbps	10.5 Gbps	10.3 Gbps
Configured OVS QoS with default queue (1 Gbps) on virtualized environment	954 Mbps	952 Mbps	952 Mbps
Configured OVS QoS with queue (250Mbps) and assigned to connection between TPM and KHTP on virtualized environment	954 Mbps	239 Mbps	239 Mbps

Table 3. Bandwidth between data centers.

According to the table 3, by using default OVS configuration, the bandwidth obtained between all the connections of data centers are not similar to physical environment. Thus, QoS is configured in OVS, and default queue which is configured as 1 Gbps is assigned to QoS. All of the connections between the data centers are then assigned to default queue. A result of this configuration is obtained from iperf as shown in table 3.

As compared to the physical environment and configured OVS QoS with default queue of 1 Gbps on virtualized environment, the bandwidth between TPM-DC1 & TPM-DC2 on virtualized environment is now similar to physical environment.

However, to achieve bandwidth of 250 Mbps between the connection of TPM-DC1 & KHTP-DC3 and TPM-DC2 & KHTP-DC3, a QoS queue with 250 Mbps is configured and assigned to the connection between TPM and KHTP. As the result shown in table 3, the bandwidth of all of the connections on virtualized environment is similar to physical environment.

	Round-trip time (ms)					
	TPM-DC1 & TPM-DC2		TPM-DC1 & KHTP-DC3		TPM-DC2 & KHTP-DC3	
	Latency	Jitter	Latency	Jitter	Latency	Jitter
Physical environment	0.490	0.136	13.491	0.684	14.004	1.079
Without using TC on virtualized environment	0.506	0.071	1.495	0.215	1.473	0.206
Configured TC with latency of 12ms on virtualized environment	0.499	0.062	13.525	0.169	13.580	0.269
Configured TC with latency of 12ms and jitter of 2ms on virtualized environment	0.517	0.076	13.385	1.154	13.423	1.132

Table 4. Round-trip time between data centers.

As for simulating the latency and jitter on virtualized environment, TC is used. The round-trip time which obtained via Ping from both physical environment and virtualized environment is shown in table 4. Without using TC on virtualized environment, the latency and jitter of the connection between TPM-DC1 & TPM-DC2 is similar to the physical environment.

Hence, TC with the latency of 12ms is configured in the connection between TPM and KHTP to obtain the similar latency result on the virtualized environment for connections between TPM-DC1 & KHTP-DC3 and TPM-DC2 & KHTP-DC3. Then, TC with the latency of 12ms and jitter of 2ms is configured to achieve similar results for both latency and jitter which is shown in table 4.

4. Experiment Setup, Results and Discussions

4.1. Experiment Setup

Once the network bandwidth, latency and jitter of the virtualized environment are configured to be similar to the physical environment, the Ceph storage cluster was setup and configured to be the same in terms of the CRUSH Map used in the physical environment. A Ceph CRUSH Map is basically a map with the information of how data can be placed and replicated according to a set or rules or policies. In this particular setup, the CRUSH Map was designed to make sure that data are placed by first making sure that all data centers must have a copy or replica of that data.

IOzone is used to conduct the benchmark of the Ceph cluster file system in the virtualized environment. Block sizes of 4kB, 8kB, 16kB, 32kB, 64kB, 128kB and 256kB and various file sizes of 1MB, 10MB and 100MB are used to conduct Sequential Read and Sequential Write benchmarks by IOzone. The results collected from the virtualized environment are then compared to those collected in an earlier benchmark of the physical environment and they are presented in the following Figure 2 and Figure 3 respectively.

4.2. Experiment Results

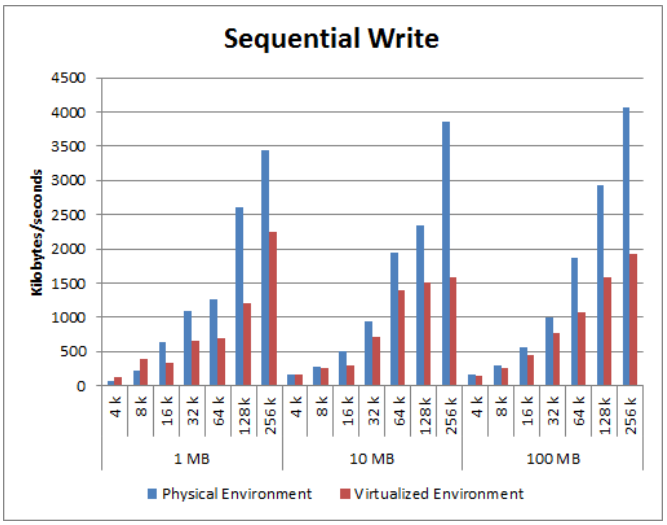

Figure 2. Result for Sequential Write.

From Figure 2, the result of sequential write on both physical environment and virtualized environment is shown and similar graph patterns are observed. The difference in the speed is most likely attributed to the fact additional overheads are introduced by the hypervisor as well as multiple VMs sharing same disks as well as possibly other effects such as caching for example.

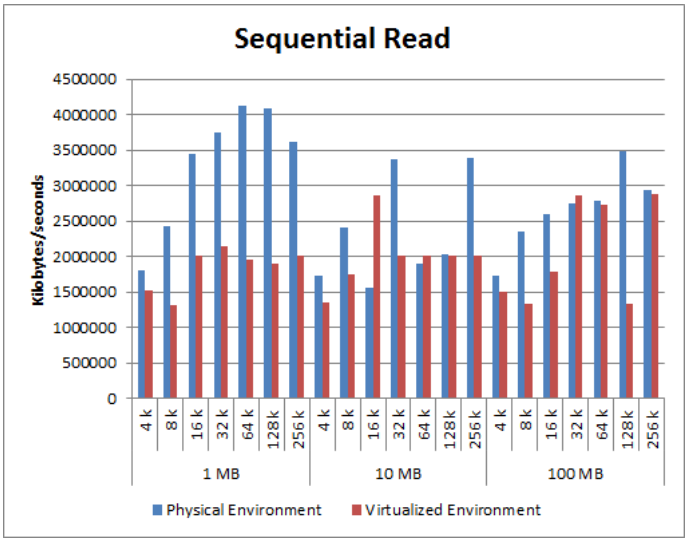

Figure 3. Result for Sequential Read.

From Figure 3, the speed of sequential read on both physical environment and virtualized environment are rather inconsistent. However, in certain test scenarios, the reading speed between physical environment and virtualized environment are very similar.

4.3. Experiment Discussions

These differences compared to the write scenarios is likely to be caused by the fact the write process for the Ceph cluster typically requires sync from all three datacenters to ensure data integrity thus show similar patterns but lower speed. On the other hand, read scenarios do not require such requirement therefore given there a 3 replicas, a read can be done from the closest datacenter or can be from the furthest datacenter which may have caused the observation above more so since the whole virtualized environment was setup within a single host.

5. Conclusions and Future Work

As discussed in Section 4, this paper has demonstrated that a properly designed and configured virtualized environment is a valuable tool for researchers for simulating a distributed file system over WAN without an actual physical environment. The performance results indicated a very similar read and write performance trends if systematic approach were applied and proper measurements were obtained from the physical environment during the design phase.

While the choice of hypervisors and the overhead introduced may cause certain loss of performance, a virtualized environment can give a basic view of how a real world setup may

behave. For future work, we intend to study the database performance on Ceph over WAN as well fine-tuning the simulation steps to get a better accuracy of the testing results.

References

1. Du, D.H.C., Recent Advancements and Future Challenges of Storage Systems. *Proceeding of the IEEE* **2008**, Volume 96, Issue 11, pp. 1875-1886.
2. Devulapalli, A.; Dalessandro, D.; Wyckoff, P.; Ali, N. ; Sadayappan, P., Integrating parallel file system with object-based storage devices. *Proceedings of ACM/IEEE Conference on Supercomputing* **2007**, 1 - 10.
3. Shuibing He; Dan Feng, Implementation and Performance Evaluation of an Object-based Storage Device. *Proceedings of SNAPI 2007 Fourth International Workshop on Storage Network Architecture and Parallel I/Os* **2007**, 129-136.
4. Sage A. Weil; Scott A. Brandt; Ethan L. Miller; Darrell D. E. Long; Carlos Maltzahn, Ceph : a scalable, high-performance distributed file system. *OSDI '06 Proceedings of the 7th symposium on Operating systems and implentation* **2006**, 307 - 320.
5. Sage A. Weil; Scott A. Brandt; Ethan L. Miller; Carlos Maltzahn, CRUSH: Controlled, Scalable, Decentralized Placement of Replicated Data. *Proceedings of the ACM/IEEE SC 2006 Conference* **2006**, 31 - 42
6. Yu Hua; Yifeng Zhu; Hong Jiang; Dan Feng; Lei Tian, Supporting Scalable and Adaptive Metadata Management in Ultralarge-Scale File Systems. *IEEE Transactions on Parallel and Distributed Systems* **2011**, Volume 22, 580 – 593
7. Brent Welch; Garth Gibson, Managing Scalability in Object Storage Systems for HPC Linux Cluster. *Proceeding of 21st IEEE/12th NASA Goddard Conf. Mass Storage Systems and Technologies (MSST)* **2004**, 433-445
8. Michael Factor; Kalman Meth; Dalit Naor; Ohad Rodeh; Julian Satran, Object storage : the future building block for storage systems. *Proceedings of 2nd International IEEE Symposium on Mass Storage Systems and Technologies* **2005, 119 – 123**
9. Christina Delimitrou; Sriram Sankar; Kushagra Vaid; Christos Kozyrakis, Decoupling Datacenter Storage Studies from Access to Large-Scale Applications. *IEEE Computer Architecture Letters*, **2012,** Volume 11, 53 - 56
10. Tzong-Jye Liu; Chun-Yan Chung; Chia-Lin Lee, A High Performance and Low Cost Distributed File System. *Proceedings of 2011 IEEE 2nd International Conference on Software Engineering and Service Science (ICSESS)* **2011**, 47 - 50
11. IBM Best practices for KVM, 2nd Ed,; 2012.
12. Alexandru C.B.; Razvan D., KVM IO Profiling. *Proceedings of 2013 RoEduNet International Conference Networking in Education and Research* **2013**, 1 - 6

13. Ben Pfaff; Justin Pettit; Teemu Koponen; Keith Amidon; Martin Casado; Scott Shenker, Extending Networking into the Virtualization Layer. *Proceedings of HotNets-VIII* **2008.**
14. Mohd Bazli Ab Karim; Jing Yuan Luke; Hong Hoe Ong; Johanes Joseph, Preliminary Study of Two Distributed File Systems for Cloud Infrastructure Storage: Ceph vs. GlusterFS. *25th Pacific Rim Application and Grid Middleware Assembly, PRAGMA25,* 17-18 Oct 2013 (Poster).
15. Mohd Bazli Ab Karim; Jing-Yuan Luke; Hong-Hoe Ong; Ming-Tat Wong, Challenges of Deploying Wide-Area-Network Distributed Storage System under Network and Reliability Constraints – A Case Study. *26th Pacific Rim Application and Grid Middleware Assembly, PRAGMA26,* 9-11 Apr 2014 (Poster).
16. Daehee Kim; Sejun Song; Baek-Young Choi, SAFE: Structure-Aware File and Email Deduplication for Cloud-based Storage System. *Proceedings of 2013 IEEE 2nd International Conference on Cloud Ntworking (CloudNet)* **2013**, 130-137.