Abstract

Immediately after a widespread disaster, critical information such as census registers should be available in the disaster-affected area to confirm people’s safety. However, when a widespread disaster actually happened, such information was not available in the Great East Japan Earthquake of 2011 even though the information were protected in safe and remote site over the Internet. This is because the disaster damages both data and internet access of the area. To enable the information available even immediately after a widespread disaster occurs, we propose Risk-aware Data Replication (RDR), which replicates data to safe and nearby sites in the area without the wide area network. It evaluates the site-to-site risk of data loss, and seeks low-risk sites using mathematical programming approach. The result of a simple earthquake simulation shows that RDR improves data availability to more than 10% compared to the case of random selection of replication sites and the computation time is less than several seconds even under the 1350 sites. Our contribution is to provide an organized, effective, and feasible way to decide replication sites based on their disaster risks under massively multiple sites environment.

Keywords: Storage System, Availability, Replication, Backup System, Disaster Recovery

1 Introduction

Many kinds of natural disasters intermittently occur all over the world. A widespread disaster, such as an earthquake and tsunami, may damage information systems seriously and stop their services. However, immediately after a widespread disaster, local governments and medical institutions need critical information: census registers for confirmation of safety, medical records, and prescriptions for health-care. Therefore, our research purpose is to continue such services in a disaster-affected area.

Remote replication technique, which replicates data into a remote site over the Internet, is widely used to mitigate such disaster effects on a service. For example, SnapMirror [1] transfers data of a primary site to a remote replication site which is never damaged simultaneously, and the replication site provides the primary’s services instead when a disaster damages them.

Remote replication technique is not sufficient when a widespread disaster damages both a primary site and internet access of the disaster-affected area. This is because the recipients of services in the area cannot access the replication site over the Internet. This problem actually happened in the Great East Japan Earthquake of 2011.

Therefore, our goal is to design a highly-available storage system that enables recipients of services in disaster-affected area to access the alternative services immediately after a widespread disaster. To achieve this, we propose Risk-aware Data Replication which utilizes geographically-distributed sites which are usual in branch offices of local governments and medical institutions, and evaluates the site-to-site risk of data loss, and seeks low-risk sites using mathematical programming approach.

2 Related Work

Effective data distribution for availability has been widely studied in storage systems communities.

Dynamo [2] distributes data at a set of nodes (i.e., storage hosts) based on consistent hashing technique [3]. Dynamo system determines rep-
lication targets by using a “ring” created by the output range of circular hash function. Each node is assigned to the random value within the ring space, and replicates data at its successors. Dynamo distributes data essentially in a random manner because it partitions hash range randomly in adding a new node to the system, and never considers the node’s safety against widespread disasters. Therefore, Dynamo requires high redundancy of data if we hope for high availability against such disasters.

Cassandra [4] also distributes data at a set of nodes based on consistent hashing technique. Additionally, it provides various replication policies such as “Rack Aware” and “Datacenter Aware”. By activating these policies, Cassandra system replicates data at different racks or different datacenters from the node storing primary data. These policies enable the system to avoid failures due to power outage, cooling failures, network failures, and natural disasters. Cassandra only considers whether the replication target is installed in the same rack or the same datacenter as the primary node, and never considers the node’s safety against widespread disasters. Therefore, Cassandra also requires high redundancy of data if we hope for high availability against such disasters.

3 Problem Establishment

3.1 Target System

Our target system consists of branch offices of local governments and medical institutions, which can be considered as a class of geographically-distributed storage systems illustrated in Figure 1.

It consists of some areas. Each area is distant enough from each other not to be damaged simultaneously by an earthquake, and is connected with a Wide Area Network (WAN).

Every area has some sites connected with a Local Area Network (LAN) in it. Every site includes servers and a storage system, and provides its own services to recipients in the area. The number of sites may be more than several thousand because there are such numbers of medical institutions in a large prefecture of Japan [8].

We assume that an earthquake damages only a part of an area, that is, some of sites, a part of LAN, and a WAN connection, but each site in the same area can communicate with each other using the area’s LAN.

The target system improves data availability using the following way: each site replicates its data to other sites periodically; when an earthquake damages a part of an area; one of the surviving site takes over the services of the damaged site.

3.2 Requirements

To design the highly-available storage system under the target system, we should meet two requirements: a protection requirement and an accessibility requirement.

The protection requirement is that a primary site and its replication sites should not be simultaneously damaged by an earthquake. This requirement is obviously necessary not to lose both primary data and its replica. Conventional remote replication technique meets this requirement by replicating data to a site in other area over the Internet.

The accessibility requirement is that a primary site and its replication sites should be in the same area. This is because recipients of services in disaster-affected area cannot access its replication site in another area when an earthquake damages WAN connection.

3.3 The Issue

The issue we have to solve is to meet both the requirements above: to find pairs of two sites that are not damaged simultaneously in the same area. To meet both the requirements, there are at least two difficulties.

The first difficulty is that it is precisely unpredictable where the next earthquake occurs and how strong it is. It is rather easy to find a safe site against a particular earthquake whose location and strength are known because we can just replicate data to a sufficiently far site from the earthquake. However, the unpredictable earthquakes may cause data loss by damaging both primary data and its replicas simultaneously.

The second difficulty is that the number of
combinations with pairs of primary-replication sites may be huge. In conventional remote replications, the number of replication sites is no more than three sites, therefore, an administrator can select a replication site easily. By contrast, in our research, the number of replication sites may be more than several thousands, and the number of combination is much more, therefore it is difficult for an administrator to check all the combinations and select a proper replication site.

4 Our Approach

Replicating data to all the sites is one of the simple solutions. However, it requires a lot of storage capacities. If the storage system consists of a thousand sites and if each site needs a storage capacity of 1TB for its services, this simple solution requires each site to have its storage capacity of 1PB.

Therefore, we propose Risk-aware Data Replication (RDR). RDR quantifies and evaluates an earthquake risk of every site, and only replicates data to low-risk sites. Therefore, it can save their storage capacities with maintaining the data availability.

We construct two design policies of RDR to overcome the two difficulties described in Section 3.3.

The first policy is that RDR focus on multiple and stochastic earthquakes information such as earthquake forecast maps provided by a government agency. This policy overcomes the first difficulty, and makes RDR avoid unpredictable data loss.

The second policy is that we use the model known as mathematical optimization problem. This policy makes RDR capable to support massively multiple sites because the model provides organized procedure to decide pairs of primary-replication sites.

5 Risk-aware Data Replication

To put RDR with the two policies described in Section 4 into action, we design it with following three steps: modeling an earthquake risk, formulating the problem into the form of mathematical optimization problem, and solving it numerically to decide safe pairs.

5.1 Modeling an Earthquake Risk

In this paper, we introduce a data loss probability which indicates two different sites are simultaneously damaged in the same area due to target earthquakes in order to model the earthquake risk. This is possible since the target earthquakes are stochastic in the earthquake forecast map and the risk can be described by a form of probability.

Here, we model the data loss probability under an assumption that it is equally probable where earthquakes occur and how strong it is for the simplest case of the earthquake forecast map.

Let $S$ denote the strength of an earthquake, the data loss probability $P_{ij}$ between sites $i$ and sites $j$ satisfies the following two properties described in Figure 2:

A) $P_{ij}$ is continuous and increases as $S$ increases,

B) $P_{ij}$ is positive and converges to 1 as $S \to \infty$,

where $i, j \in \mathbb{N}, i, j \in [1, n], \mathbb{N}$ is a set of natural numbers, and $n \in \mathbb{N}$ denotes the total number of sites. These properties follow our intuitions. The property A indicates that the two sites are more likely to be damaged as the strength increases. The property B describes that all the value of the probability is more than 0 and less than 1.

One of the descriptions of the data loss probability with these intuitive properties can be derived in the following steps:

1. calculates a strength of an earthquake $S$ from the given value $d_{ij}$, which denotes the distance between sites $i$ and sites $j$, using the relation of

   $S(d_{ij}) = \log_{10} d_{ij};$

2. calculates $P_{ij}$ by using the sigmoid function $\zeta$ with given parameters $a, b$ of

   $P_{ij}(d_{ij}) = \zeta \left( a(-S(d_{ij}) - b) \right) = \frac{1}{1 + e^{a(\log_{10} d_{ij} - b)}}.$

The first step is to describe the reasonable strength $S$, and is based on an statistical earth-
quake model [6]. We see the maximum speed of ground as the strength of earthquake, and assume that the increasing rate of the probability depends on the decreasing rate of the speed. The second step is to satisfy the properties A and B. Using the sigmoid function $\zeta$ with $a > 0$, it assures $P_{ij} > 0$ and $P_{ij}$ converges to 1. Additionally, we use two design parameters $a, b$ to tune the curve of $P_{ij}$. $P_{ij}$ converges to the step function as $a \to \infty$, and we can model $P_{ij}$ with a gradual slope if we set a small $a$. Essentially, these parameters depend on the property of target earthquakes, however, the design method remains future work.

The derived data loss probability follows general intuition that a strong earthquake is likely to damage two sites which are located far from each other, and that the two sites near each other are likely to be damaged simultaneously.

5.2 Formulating into the Mathematical Optimization Problem

We formulate the problem to decide replication sites into the Integer Programming Problem (IPP) which is one of the mathematical optimization problems. The IPP consists of an objective function and constraints including integer variables.

We aim to get a combination with pairs of the primary-replication sites that optimize the data amounts surviving from an earthquake. By using $P_{ij}$ as a weight coefficient of an objective function to decide replication sites, we encourage RDR to decide safe sites as a replication sites.

We also consider the number of replicas to create and the amount of storage capacity that can be consumed by replicas. If the number of replicas is large, the availability after an earthquake is clearly large, but the storage capacity consumed by replicas is also large. To make this trace-off tunable, the number of replicas should be a design parameter. The amount of storage capacity for replicas is valuable and limited resources for many systems. If the amount is large, the replicas would concentrate in the safest site. This is not necessarily beneficial. To avoid such a case, the storage capacity for replicas should be a design parameter. These two conditions can be described into constraints of the IPP.

We show the formulation of the IPP below.

(1) Objective Function

RDR aims to protect data of sites in an area against target earthquakes. Here, we aim to minimize the total amount of data which is damaged and lost. Thus, we set an objective function of

$$f(x_{12}, \ldots, x_{n(n-1)}) = \sum_{i=1, i \neq j}^{n} \sum_{j=1}^{n} D_i P_{ij} x_{ij},$$

where $x_{ij} \in \{0, 1\}$ denotes whether a site $j$ has a replica of a site $i$ or not, and $x_{ii} = 0, \forall i$. $D_i$ denotes the primary data amounts in site $i$, and $P_{ij}$ denotes the data loss probability described in Section 5.1. This definition of $f$ denotes the total amounts of data which are expected to be damaged for a combination of the variables $x_{ij}$. The combination of $x_{ij}$ to minimize $f$ is the most highly available solution against target earthquakes.

(2) Constraints

RDR uses two constraints: a redundancy constraint and a storage capacity constraint.

The redundancy constraint is to regulate the number of replicas to create from a primary site in a replication process. This constraint is necessary because every site tries to create as many replicas as possible without it. Thus, the redundancy constraint is set to bound the maximum value of data redundancy, and described by

$$\sum_{j=1}^{n} x_{ij} = R_i, \forall i,$$

where $R_i$ denotes the number of replicas of site $i$, and is given by an administrator.

The storage capacity constraint is to regulate the number of replicas that a replication site receives in a replication process. This constraint is necessary because every site has finite storage capacities. Thus, the storage capacity constraint is set to bound the maximum value of storage capacity consumed by replicas, and described by

$$\sum_{i=1}^{n} D_i x_{ij} \leq F_j, \forall j,$$

where $F_j$ denotes the number of storage capacity of site $j$ and is given by an administrator or each sites.

5.3 Solving the IPP

To minimize $f$, we can use an existing optimal algorithm. However, IPP generally takes time to solve. Thus, we use the following two algo-
rithms.

(1) Branch-and-Bound method (BB method) [7]
BB method is an algorithm used to seek the combination of $x_{ij}$ which minimize the objective function $f$. This method guarantees that the solution is optimal, but the computation time generally increases fast as the total number of sites increases.

(2) Greedy method
Greedy method is an algorithm that is used to search for the combination of $x_{ij}$ by finding the component of $x_{ij}$ which has the largest $P_{ij}$ in turns. This method does not guarantee an optimal solution, however, the computation time is generally small.

6 Implementing RDR Simulator

We implemented RDR simulator by C to evaluate RDR. Figure 3 describes an input-output block diagram of it. The RDR simulator consists of three parts of computation: a field creator, a risk creator, and a pair creator.

The field creator simulates sites in an area, which has a flat ground, a slope, or a mountain, and outputs a field information and sites information. The field information includes the geological formation of the ground described by specified coordinates of computational mesh. The sites information includes their locations, their usage and free amounts of storage capacities.

The risk creator calculates the site-to-site risk $P_{ij}$. The risk creator receives two pairs of $(d_{ij}, P_{ij})$ as a risk hint, and calculates the parameters $a, b$ with linear interpolation in the input variable space of the sigmoid function $\zeta$. Then it outputs the risk information, the value of probabilities of all the pairs of sites.

A pair creator decides a combination with safe pairs of primary-replication sites. It formulates and solves the IPP by using constraints information and a specified algorithm. The constraints information includes data redundancy. The algorithm can be selected from BB method and Greedy method. For BB method, it invokes lp_solve 5.5, and for Greedy method, we just implemented its algorithms. Finally, it outputs the safe pairs.

7 Evaluation

We measured the RDR’s improvement of availability and computation time to decide pairs using a simple earthquake simulator, to evaluate RDR’s effectiveness and feasibility.

7.1 Earthquake Simulator

We implemented a simple earthquake simulator to evaluate RDR’s improvement on availability. The earthquake simulator simulates to damage sites in a field.

We input the earthquake’s magnitude and the location of its hypocenter including the depth to the simulator. The simulator calculates the damage probability of every sites based on the statistics of an earthquake [6][8], using the probability, the simulator decides if sites are damaged or not.

7.2 Simulation Setup

We conducted the simulation on a server with a processor (2-core 1.87 GHz, Xeon E5502; Intel Corp.) and 6 GB RAM, and set parameters of the RDR simulation and the earthquake simulation in the following.

The field is flat and 200km $\times$ 200km wide. The sites are located randomly on the field, and the number $n$ is 135 or 1350. Each site has only one data and only one storage capacity for replica, that is, $D_i = 1, F_i = 1, \forall i$, and the replicates its data only one time, that is, the redundancy $R_i = 1, \forall i$.

We simulated two kinds of earthquake to evaluate robustness of the algorithm. One is a far-field earthquake, the hypocenter of which is far away from every site. To simulate this, we set an earthquake of which hypocenter is (400, 100, −40) and its magnitude is 8. Another is a near-field earthquake, the hypocenter of which is near some sites. To simulate this, we set an earthquake of which hypocenter is (100, 100, −40) and its magnitude is 8.
We set $(10, 0.9)$ and $(50, 0.5)$ for the risk hint described in Section 6, and repeat each kind of earthquake 500 times to obtain an average of measured data.

We use not only the BB method and the Greedy method described in Section 6.3, but also Random method for algorithms in order to compare. The Random method decides pairs of primary-replication sites in a random manner, and the expectation of availability is ideally 75% in the case that the number of replicas is 1 and damaged half of the total number of sites.

### 7.3 Results

1. **Distribution of Primary-replication Pairs**
   Figure 4 shows the histogram of distances between two sites each of which has the other’s replica when the number of sites is 135.

   In the case of the BB method, the distribution is concentrated to the distance around 160, there are no pairs except 140-180, and the average of the distance is 154.6. This shows that BB method pairs sites to biasing toward its average of the distribution.

   It shows that Greedy method pairs sites which are more distant than Random method and BB method do. There are pairs of which distance is over 180 and near the maximum value of the distance 259.2. This shows that the Greedy method pairs sites which have as long distance as possible.

2. **Availability**
   Figure 5 shows the available data ratios, the percentage of data amounts that survive from an earthquake when the total number of sites is 135. The values of the BB method is more than 20%, larger than that of the Random method, but the values of the Greedy method is less than 20% smaller in the case of near-field earthquake. This shows that RDR is effective to improve data availability if the algorithm is well-designed. For a far-field earthquake, the Greedy method value is 0.6% lower than that of BB method. This shows the Greedy method to be as effective as the BB method under a far-field earthquake.

3. **Time to Decide Safe Pairs**
   Figure 6 shows the time to decide pairs. This shows that the Greedy method can select replication sites in a shorter time than the BB method does, and that the time is less than several seconds.

### 8 Conclusions

This paper presents Risk-aware Data Replication, which replicates data to nearby and safe sites based on evaluating earthquake risks. We show that a problem to decide replication sites can be formulated into a form of an Integer Programming Problem model by using a probability based on a site-to-site risk against an earthquake, and that we can solve it using existing algorithms. By simulating an earthquake, we show that, depending on the algorithm, RDR improves data availability to more than 20% larger than the case of random selection of replication sites, and the
computation time is less than several seconds even under the 1350 sites.

In the future, we plan to improve the definition and design method of a site-to-site earthquake risk. The proposed site-to-site risk seeks a weak solution against the earthquakes that occur between the primary and replication sites. To improve this, we plan to derive the site-to-site earthquake risk from an earthquake forecast map which predicts a geographical range of a hypocenter mathematically.

Acknowledgment

We thank Masatoshi Shimbori, Hiroshi Ichinomiya, and Mitsuyoshi Igai for research computing support, and Mitsuo Hayasaka, Atsushi Sutoh, and all the members of our project team for their useful comments. This work is supported as ‘Research and Development on Highly-functional and Highly-available Information Storage Technology’, sponsored by the Ministry of Education, Culture, Sports, Science and Technology in Japan.

References