閲覧数の差分とパレートの法則によるリストア後のテスト対象の限定と時間の短縮

西村 克己[1]　平尾 真斗[2]　串田 高幸[1]

概要：東京工科大学コンピュータサイエンス学部の研究室である Cloud and Distributed Systems Laboratory (以後 CDSL) の日本語サイトは WordPress で運用されている．Web サイトのページを確認する際, サイト内の全体のページ, ハイパーリンクを対象にアクセスできるかテストする．課題は, バックアップデータのコンテンツ全体のテストを実行する際に, コンテンツを限定してテストを実行した際と比べ時間がかかることである．提案では, 複数日のバックアップデータをリストアした際に生じる WordPress の閲覧数の差分を算出した．その後, パレートの法則から閲覧数に差分の大きい値上位 20%を WordPress のサイトの確認対象として定めた．評価では, 提案適用前と提案適用後における, テストの実行時間を比較した．テストの実行対象として, コンテンツ全体を計測した時間, 閲覧数に差分のあったコンテンツ全体を計測した時間, 閲覧数の差分の大きい値上位 20%のコンテンツを計測した時間をそれぞれ 10 回計測した．コンテンツの全数のテストを実行した際の時間は, 平均で約 123.46 秒であり, 閲覧数の差分のあったコンテンツ全体を検索した際の実行時間は, 平均で約 26.53 秒, 閲覧数の差分が大きい値上位 20%のコンテンツをテストの実行対象として定めた際, テストの実行時間は約 5.35 秒であった．コンテンツの全数と閲覧数の差分のあったコンテンツ全体のテストを実行した際は平均約 79%, コンテンツの全数と閲覧数に差分の大きい値上位 20%のテストを実行した際は平均約 96%実行時間を削減できた．テストの実行時間を削減することでバックアップデータに不備があった際, 原因の探索により早く取り組むことができる．

1. はじめに

背景

バックアップとは, 削除または破損したデータを回復する目的で作成及び保持される生産データのコピーである [1]. データのバックアップは, 自然災害や人為的ミス, ハードウェアやソフトウェアの障害が原因でのデータ損失を防ぐために必要不可欠である [2,3]. バックアップを作成しておくことで, 運用しているシステムのデータを損失した際に, データを復元することができる [4].

システムのバックアップを行う際には, フルバックアップ, 差分バックアップ, 増分バックアップ, 逆増分バックアップをもちいる [5]. フルバックアップとは, ファイルシステムの内容全体をバックアップデバイスにコピーすることである [6].

リストアとは, バックアップを保存しているストレージから, データを取得することである [7,8].

バックアップのテストを実行することは, 定期的に作成されるバックアップが実際に利用できるかを確認し, 作成に問題が発生した際に把握することで, バックアップが必要になる前にデータの復元の信頼性を確認することに役立つ [9,10].

東京工科大学 コンピュータサイエンス学部 Cloud and Distributed Systems Laboratory (以後 CDSL) では, 仮想マシン上に WordPress を構築して, 研究室に所属する学生が執筆するブログやテクニカルレポートを公開するサイトを運用している[*1]．CDSL の日本語サイトは, 毎日 0 時にフルバックアップを作成している．バックアップは作成後, Network Attached Storage(以後 NAS) に 1 ヶ月間保存する．CDSL の日本語サイトのデータをリストアする際, バックアップデータが保存されている NAS 上からデータの復元を行い, 復旧させる．2024 年 09 月 15 日, 2024 年 09 月 16 日時点での CDSL の日本語サイトの投稿件数は 437 件あり, 固定ページが 12 件の計 449 件が公開されている．2024 年 09 月 15 日 0:00 と 2024 年 09 月 16 日 0:00 までにそれぞれ取られたバックアップデータのリストアを行った際に, 閲覧数の差分順に並べ替えた上位 10 件の記事の結果のグラフを図 1に示す.

[1] 東京工科大学コンピュータサイエンス学部
〒 192–0982 東京都八王子市片倉町 1404-1
[2] 東京工科大学大学院バイオ・情報メディア研究科
〒 192–0982 東京都八王子市片倉町 1404-1

[*1] https://ja.tak-cslab.org/

© 2024 Cloud and Distributed Systems Laboratory

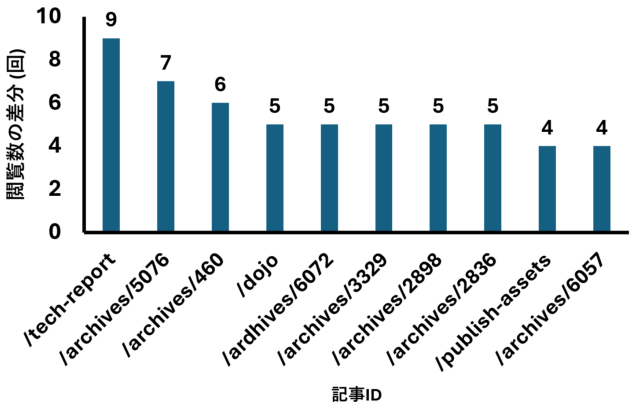

図 1　09 月 15 日と 09 月 16 日の閲覧数の差分

図 1より, 2024 年 09 月 15 日時点と 2024 年 09 月 16 日時点でのバックアップデータのリストアを行った際には, バックアップデータの作成の日時によってそれぞれの記事の閲覧数に差がみられる. 例えば/tech-report では 9 件の記事の差分, /archives/5076 では 7 件の記事の差分があった.

Web サイトのページを確認する際, ページテスト, ハイパーリンクテストがあり, サイト内の全体のページ, ハイパーリンクを対象にアクセスできるかテストする [11].

課題

課題は, バックアップデータのコンテンツ全体のテストを行うと, コンテンツを限定してテストを実行したときと比べ時間がかかることである. 本稿では, WordPress の記事や固定ページの中身が保存されているデータベースとプラグイン, 画像が保存されているディレクトリのコンテンツをバックアップデータとして指す.

バックアップデータのコンテンツ全体のテストを実行する流れを図 2に示す. 図 2は, WordPress のリストア後に行う, コンテンツ全体のテストを示している. テストを実行するソフトウェアが, 確認対象としてあげられている仮想マシン内で動作している WordPress のコンテンツファイルを確認している状況を表している. 確認対象としている WordPress の記事の件数が多くなるほど, 確認に必要な時間がかかる.

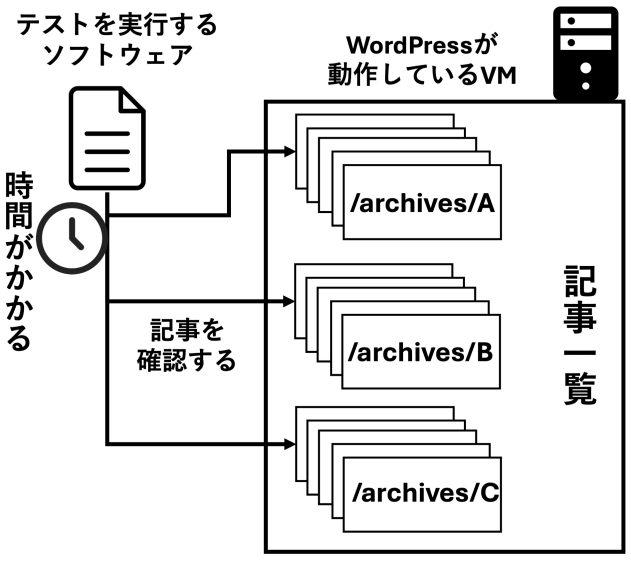

図 2　コンテンツ全体のテストを実行する流れ

各章の概要

第 2 章では, 関連研究について説明する. 第 3 章では, 課題について解決するための提案方式について説明する. 第 4 章では, 提案方式の実装について説明する. 第 5 章では, 評価実験として実験内容と実験結果と分析について説明する. 第 6 章では, 提案方式についての議論を説明する. 最後に, 第 7 章にて結論を説明する.

2. 関連研究

目的復旧時間 (以後 RTO) と目的復旧ポイント (以後 RPO) の観点から, 目的復旧要件を達成するデータバックアップが必要と主張している研究がある [12]. この研究では大規模な障害に対して回復を行うために, 組織に所属するユーザーが作成する大量のデータをサポートする必要があると言及している. 災害時 (障害発生時) からの復旧手法についての検討を行っているが, 実際にバックアップの使用可能の判断までは行っていない点に改善の余地がある.

システムが利用できなくなると, ビジネスに深刻な影響を与える. そのため, 多くの組織が災害復旧 (Disaster Recovery) を設定することを主張している研究がある [13]. この研究では, 分析モデルと障害注入実験をもちいて, 可用性, ダウンタイム, RTO, RPO の災害復旧に必要な指標の評価を行っている. バックアップと復元のデータ量によって, 環境の可用性が異なることを証明しているが, バックアップデータの確認を行っていない点に改善の余地がある.

リレーショナルデータベースを対象とした, バックアップとリカバリにおけるデータの一貫性に対する提案を述べている研究がある [14]. この研究では, データ管理システムの潜在的な問題であるバックアップとリカバリ, データの整合性に対する提案を行った. メンテナンス時の問題を解決しようとしているが, リストア時のデータの確認までは行っていない点に改善の余地がある.

3. 提案

提案方式

バックアップデータをリストアした際に, テストの実行対象を絞ることで, テストの実行時間を削減することを目的としている. 提案方式では, WordPress の閲覧数に差分がある記事を抽出する. その後, 閲覧数をもとに, バックアップデータに含まれている WordPress の記事や固定ページに対してテストを実行する対象を定めコンテンツを限定

© 2024 Cloud and Distributed Systems Laboratory

することで, テストの実行時間を短縮する. 提案方式では前提条件として, 毎日フルバックアップを作成していることをおいている.

パレートの法則にもとづいて, 閲覧数の変化件数に 1 件以上あった記事から閲覧数の差分の大きい値上位 20%をテスト対象として設定する [15]. パレートの法則とは, 別名 20:80 と呼ばれており, 約 20%の努力で約 80%の結果が得られる法則である [16,17].

バックアップデータをリストアした後のバックアップデータの比較を提案ソフトウェアが行う流れを図 3に示す.

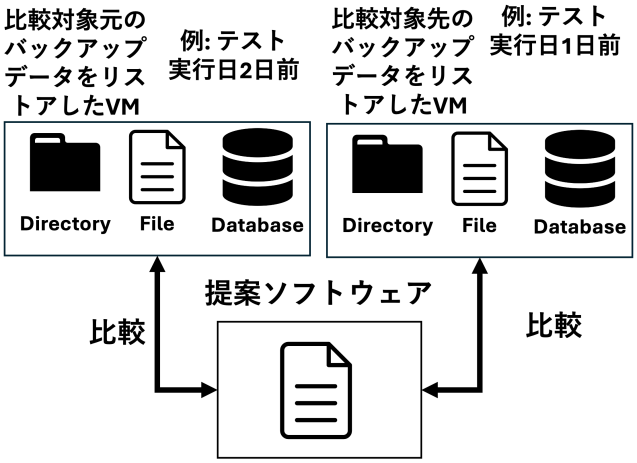

図 3 リストア後のデータを比較

比較対象元のバックアップデータと比較対象先のバックアップデータでそれぞれリストアを行った仮想マシンを用意する. 図 3では, 比較対象元のバックアップデータをテスト実行日 2 日前のデータをリストアした VM, 比較対象先のバックアップデータをテスト実行日 1 日前のデータをリストアした VM としておいている. その後, バックアップデータに保存されている記事の閲覧数が 1 件以上増加しているページを取得する.

閲覧数の差分の算出方法

テストを実行する比較対象元と比較対象先における閲覧数の差分の算出方法を式 (1) に示す.

$$D = (A_o - B_o) \tag{1}$$

D は, テストを実行する比較対象先の閲覧数からテストを実行する比較対象元の閲覧数を引いた, 閲覧数の差分の値である. A_o はテストを実行する比較対象先の閲覧数, B_o はテストを実行する比較対象元の閲覧数を示している. 例として, A という記事の比較対象先に記録されている閲覧数である A_o が 150 件, 比較対象先の閲覧数である B_o が 100 件であったとする. この場合, 閲覧数の差分の算出方法である式 (1) を適用すると, D は, 150 - 100 を計算した 50 件があてはまる.

パレートの法則からテスト対象を決定する方法

WordPress のサイトの総記事の全てに対して閲覧数の差分の値を求めた D を利用する. パレートの法則で対象を決定する際に使用した式を式 (2) に示す.

$$N = DA \times 0.2 \tag{2}$$

N は, 閲覧数に差分のあった合計を示す DA にパレートの法則にて 20%である 0.2 を掛け合わせる値で算出される値である. N の値をもとにして, 一番大きい差分順から N 件までの閲覧数に差分のあった件数を決定する. 例として, 閲覧数の差分のあった記事が 50 件あったとする.DA は 50 があてはまる. N は, 50×0.2 の 10 という値があてはまり, 閲覧数の差分のあった上位の値から 10 件検索を行うこととなる.

ユースケース・シナリオ

仮想マシン上にある WordPress で動作するブログサイトのバックアップデータをテストすることをユースケースとして想定する. バックアップデータの中身として, WordPress が動作するのに必要なファイルである /var/www/html とデータベースの一種である MariaDB-Server の Dump ファイルをあげる. ユースケース・シナリオを図 4に示す.

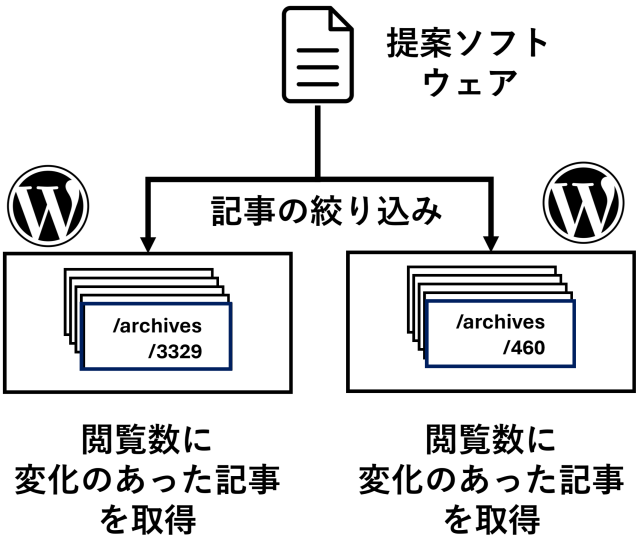

図 4 ユースケース・シナリオ

WordPress で動作するブログサイトは毎日 0 時にフルバックアップを作成しており, 1 ヶ月間データが保存されている. 図 4では, WordPress で動作するブログサイト全体のコンテンツのテストを実行していた手法を, 提案方式をもちいることによって, テストの実行対象を絞り込む状況を示している.

テストの実行時間を短縮することにより, バックアップデータに不備があった時, 原因の探索により早く取り組むことができる [18,19].

© 2024 Cloud and Distributed Systems Laboratory

4. 実装

テストを実行する対象を定めるソフトウェアを開発言語である Python を使い, 2 つ作成した. 1 つ目のソフトウェアは, WordPress のバックアップデータから, 閲覧数を取得し, データベースのテーブルに書き込む DB-Write である.

2 つ目のソフトウェアは, 1 つ目で取得した WordPress のバックアップデータから閲覧数を取得したデータを読み込み, パレートの法則にしたがって WordPress のサイトにテストを実行する FoucsCheck である. それぞれのソフトウェアを分けて説明する.

閲覧数を取得しデータを保存するソフトウェア

WordPress には, 閲覧数のデータをもとに簡単にサイトを解析できるプラグインとして, WP Statistics がある[*2]. WP Statistics は, WordPress の記事や固定ページのアクセス数を wp_statistics_pages のテーブルに保存する.

DB-Write が処理を行った後のデータベースの結果を表 1に示す.

表 1 DB-Write が処理したデータベースのテーブルの一部

id	url	total_count	title	post_status
1	/archives/460	3555	タイトル A	publish
2	/tech-report	3080	タイトル B	publish
3	/archives/3329	1103	タイトル C	publish
4	/about	807	タイトル D	publish
5	/about/member	728	タイトル E	publish

id は, 閲覧数順に並べ替えた順番を示しており, url は, 該当記事の path を示している. title は記事のタイトルを示しており, post_status は該当の記事の公開状況を示している.

DB-Write では, 記事, 固定ページのそれぞれの閲覧数の累積を合計して, 閲覧数順に並べ替えたテーブルをデータベース上に作成をしている.

表 1の total_count を比較対象元のバックアップデータ, 比較対象先のバックアップデータを仮想マシンにリストアしたデータベースに対して算出することで, 閲覧数の差分の値を取得することが可能になる.

パレートの法則にそって WordPress のサイトにテストを実行するソフトウェア

パレートの法則の処理を図 5に示す.

*2 https://wordpress.org/plugins/wp-statistics/

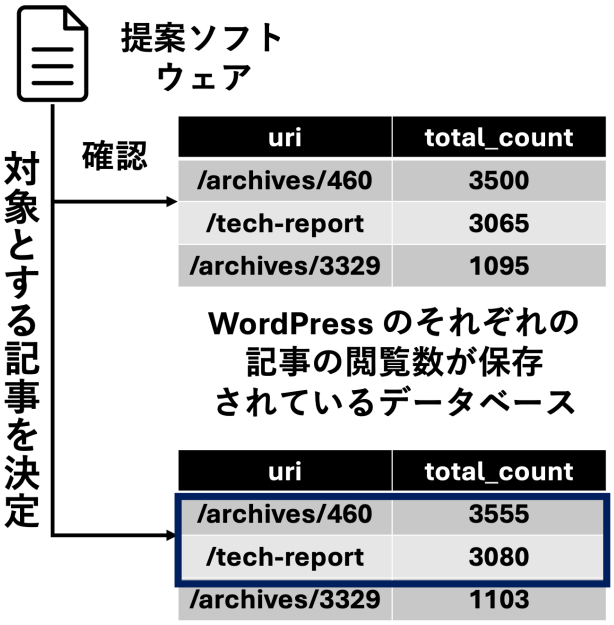

図 5 パレートの法則の処理

1 つ目のソフトウェアで説明した DB-Write の処理が完了したら, 比較元と比較先のそれぞれのバックアップデータにそれぞれ含まれている図 5で示した uri, total_count の値を取得する. その後, 比較対象元と比較対象先のバックアップデータの total_count を比較し, 比較対象元のバックアップデータの時点からの閲覧数の差分の値を計算する.

5. 評価実験

提案適用前と提案適用後における, テストの実行時間を比較して, 評価した [20]. テスト項目を (1)〜(2) で示す [21].

(1) トップページ含む該当ページ全てにアクセスできるか
(2) 該当ページに埋め込まれている画像にアクセスすることができるか

提案適用前では, 評価指標が 2 つある. 1 つ目は, 2024 年 09 月 15 日時点での CDSL の日本語サイトのバックアップデータをリストアした仮想マシンに対して, 投稿件数全体である 449 件のテストの実行時間を計測した. 2 つ目は, 2024 年 09 月 15 日と 2024 年 09 月 16 日時点での CDSL の日本語サイトをバックアップしたデータをリストアした仮想マシンに対して, 閲覧数に変化のあった 70 件の記事のテストが完了するまでの実行時間を計測した. 提案適用後では, 2024 年 09 月 15 日と 2024 年 09 月 16 日時点での CDSL の日本語サイトをバックアップしたデータをリストアした仮想マシンに対して, 閲覧数に変化のあった 70 件のうち, 閲覧数の増加が多かった上位 20%, 14 件を対象としたテストの実行時間を計測した.

CDSL では日本語サイトを毎日 0 時にバックアップを作成している. そのため, 2024 年 09 月 15 日と 2024 年 09 月 16 日の間である 24 時間の閲覧数を差分の値として算出

© 2024 Cloud and Distributed Systems Laboratory

した.

実験環境

実験環境には, 提案ソフトウェアを導入したソフトウェアを実行する仮想マシン 1 台, WordPress の 2 日分のバックアップデータをリストアした仮想マシン 2 台の計 3 台の構成で実験を行った. 以下に各仮想マシンの構成要素を示し, 図 6に実験環境を示す.

- 仮想マシン構成情報 (提案ソフトウェア実行用)
  OS: Ubuntu Server 24.04.1 LTS
  vCPU: 2Core
  RAM: 2GB
  SSD: 25GB
- 仮想マシン構成情報 (WordPress 動作用)
  OS: Ubuntu Server 24.04.1 LTS
  vCPU: 1Core
  RAM: 1GB
  SSD: 25GB

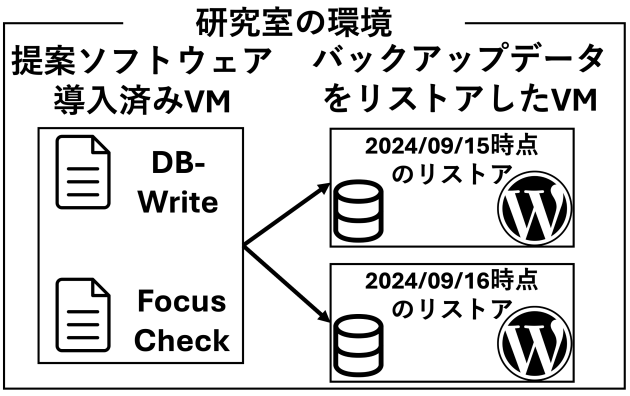

図 6 評価実験の構成

提案ソフトウェア実行用の仮想マシンでは, Python で実装した提案ソフトウェアを導入し, 動作させた. 仮想マシンは, (vCPU: 2[Core], RAM: 2[GB], SSD: 25[GB]) の性能で作成した. WordPress が動作している仮想マシンでは, Web サーバーを動作させるときに必要である Nginx, PHP, データベースサーバーである MariaDB-Server をインストールして WordPress を動作させた. 仮想マシンは 2 台とも同じ性能の (vCPU: 1[Core], RAM: 1[GB], SSD: 25[GB]) で作成した. また, 閲覧数を取得するために, WordPress のプラグインとして公開されている WP Statistics を導入し, また WordPress の記事情報を取得するために API として公開されている WP REST API を導入した[*3][*4].

実験結果と分析

バックアップデータをリストアした際の CDSL の日本語サイトにあるコンテンツ全体を対象として計測した時間, バックアップデータを比較してリストアした際に閲覧数に変化のあった記事全体を対象として計測した時間, 閲覧数に変化のあった記事からパレートの法則にあてはめ, 閲覧数の差分が大きい値上位 20%を対象として計測した時間の結果をそれぞれ示す.

2024 年 09 月 15 日と 2024 年 09 月 16 日でのバックアップデータをリストアした際に, 閲覧数の差分のあったページを取得する. その後, 閲覧数の差分の大きい記事から数えて上位 20%の記事の閲覧数を合計した値と, 残りの 80%の記事の閲覧数の合計を算出する. 算出した結果を表 2に示す.

表 2 差分の閲覧数をパレートの法則に適用した結果

項目	上位 20%の合計 (パレート適用)	残り 80%の合計 (パレート非適用)
閲覧数 (件)	71	59

2024 年 09 月 16 日の閲覧数の合計は, 2024 年 09 月 15 日の閲覧数の合計と比較し, 130 件増加していた. 2024 年 09 月 15 日と 2024 年 09 月 16 日のバックアップデータを比較し, 変化のあった上位 20%の合計は 71 件 (全体の約 54.6%) であり, 変化のあった残りの 80%の記事の閲覧数の値の合計は 59 件 (全体の約 45.3%) であった. 2024 年 09 月 15 日と 2024 年 09 月 16 日の閲覧数に変化のあった記事は 70 件あり, 閲覧数に変化のあった値の合計が 130 件あった. 閲覧数の上位 20%の算出した値の記事に対して検索した閲覧数の合計が 71 件, 残りの 80%の記事に対して検索した閲覧数の合計は 59 件あった. このことから, 閲覧数の合計の多い記事に着目することができ, 閲覧数の差分の大きいものを対象として絞り込むことができた.

テスト時間の計測

バックアップデータをリストアした際の全体の記事のテストを実行した際と, バックアップデータをリストアした際の閲覧数に変化があった記事全体のテストを実行した際の, 実行時間を 10 回計測した結果の頻度グラフを図 7で示す.

図 7は, 2024 年 09 月 15 日, 2024 年 09 月 16 日それぞれの日時の CDSL の日本語サイトを対象としたテストの実行時間を示している. 分布の青色の線は, 全数を検索した結果, 分布のオレンジの線は, 閲覧数の差分全体を検索した結果, 分布の緑色の線は, 閲覧数の差分の大きい値上位 20%を検索した結果をそれぞれ示している. 2024 年 09 月 15 日現在の CDSL の日本語サイトの投稿件数は 449 件あった. CDSL の日本語サイト全体を対象としたテストの実行時間の最小値が約 114.76 秒であり, 最大値が約 138.59 秒であった.

実装ソフトウェアである FoucsCheck は, WP REST API

*3 https://wp-statistics.com/
*4 https://developer.wordpress.org/rest-api/

© 2024 Cloud and Distributed Systems Laboratory

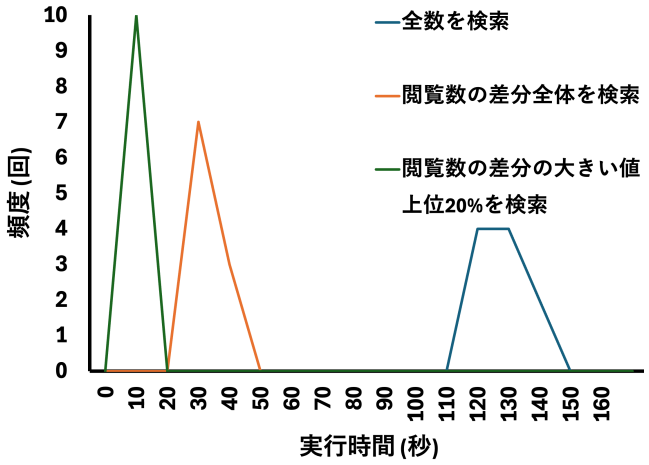

図 7 全数検索, 差分のあった記事全部, 差分の上位 20%の度数分布

を利用して, 記事の存在の確認を行っている. 全数を検索した際, CDSL の日本語サイトの投稿件数は 449 件であったため, リクエストごとに結果を返すため, 実行時間に差がみられた. また, 初回リクエストではデータベース経由で直接データを取得するため速度が遅くなり, 2 回目以降はキャッシュされたデータが利用されて実行時間に差がみられた.

2024 年 09 月 15 日と 2024 年 09 月 16 日でのバックアップデータをリストアした際に閲覧数の差分のあったページは 70 件あった. CDSL の日本語サイト全体の閲覧数の差分からテストを実行する対象を定めた際のテストの実行時間の最小値が約 20.24 秒であり, 最大値が約 34.81 秒であった. CDSL の日本語サイト全体をテストの実行対象とした際と, 閲覧数の差分の大きい値上位 20%のテストを実行した際には, 平均約 79%テストの実行時間を削減できた. 2024 年 09 月 15 日と 2024 年 09 月 16 日でのバックアップデータをリストアした際に閲覧数の差分のあったページのうち, 閲覧数の差分の大きい記事上位 20%(14 件) のテストを実行した. テストの実行時間の最小値が約 5.14 秒であり, 最大値が 5.66 秒であった. CDSL の日本語サイトのうち, 閲覧数のあった記事全体と, 閲覧数に差分のあった上位 20%のテストを実行した際には, 平均約 80%テストの実行時間を削減できた. このことから, 閲覧数の差分の大きい記事上位 20%を対象としてバックアップデータのテストを実行したときに計測した時間がコンテンツ全体と閲覧数に差分のあったコンテンツ全てを計測した時間と比較して短くなったことがわかる.

この結果から, バックアップデータのテストを実行する際, コンテンツ全体を対象とするよりも閲覧数に差分のあったコンテンツ全体を計測することが, テストの実行時間を削減できる. また, 閲覧数に差分のあったコンテンツ全体を計測するよりも, 閲覧数に差分のあった上位 20%のコンテンツを対象とすることが, テストの実行時間を削減できる. これにより, テストの実行時間を削減でき, バックアップデータに不備があった際, 原因の探索に, より早く取り組むことができる [18, 19].

6. 議論

提案方式では, 閲覧数に差分のあった上位 20%を対象とし, コンテンツを限定することでテストの実行時間を短縮した. 提案ソフトウェアがテストを実行する対象を定め, 事前に設定したテスト項目をもとにステータスコードでページの存在有無の確認を行う. ただし, ステータスコードで判断を行うため, ステータスコード上では 200 番の正常コードで返答があったとしても, ページの中身まで確認を行っていない. そのため, ページの存在は確認できても, Web ページのレイアウトが崩れることにより, ユーザーが期待する動作と異なるときがある. バックアップデータの比較を行っているため, 以前のバックアップデータと現在のバックアップデータをそれぞれリストアを行う. チェックサムやハッシュ値をもちいたファイルの比較を行うことで, テストの信頼性をあげることができる.

提案方式では, WordPress の記事の閲覧数に差分のあった期間を変数としておいている. CDSL の日本語サイトは毎日 24 時にバックアップを作成しているため, 評価実験では比較元のバックアップと比較先のバックアップの閲覧数を比較する範囲を 24 時間で設定した. ただし, 閲覧数を比較する範囲を 24 時間で設定するのは, CDSL の環境依存のため, 確実性がない. 本稿でのユースケースは, CDSL の日本語サイトをおいているが, 運用する実際の環境によって変数の値を決める必要がある. 方法として, 対象とするサイトの閲覧数を過去のバックアップやアクセスログから収集して, 閲覧数の時間帯ごとに記事のアクセス数の増減や傾向を分析する. 分析をもとに, 閲覧数の差分が現れている期間を特定する. 具体例として, 朝, 昼, 夜の各時間帯における閲覧数の傾向を比較し, 変動が集中する時間帯を特定して, 閲覧数の取得する範囲を変数として決定する.

提案方式では, 閲覧数の差分をパレートの法則にあてはめ, テストの実行対象を限定した. 2024 年 09 月 15 日と 2024 年 09 月 16 日のバックアップデータを比較した際, 閲覧数の変化のあった記事の上位 20%の合計は 71 件であり, 変化のあった残りの 80%の記事の閲覧数の値の合計は 59 件であった. 閲覧数の変化のあった上位 20%の合計は, 閲覧数の変化のあった記事全体の約 54.6%, 変化のあった残りの 80%の合計は, 閲覧数の変化のあった記事全体の約 45.3%であった. パレートの法則に適用する際, 上位 20%の合計の割合が変化のあった残りの 80%の割合 (50%:50%) と同じにならなければならない. 本稿で実験した結果は, (約 54.6%と約 45.3%) となりパレートの法則に完全には従わなかった. 提案方式では, 閲覧数の差分のあった件数の上位 20%が, 閲覧数の差分のあった合計の 80%を占めると

した. 閲覧数の差分のあった件数の上位 20%の基準を, 記事の閲覧数の合計と残りの閲覧数の合計の数が同じになるように基準を合わせることで, パレートの法則を利用して提案方式を利用することができる. 一例として, 閲覧数の差分のあった値の上位順に 50, 40, 30, 20, 10 という値があったとする. 閲覧数の変化のあった記事が 5 件あったとして, 閲覧数の差分のあった件数の値の合計が 150 であったとする. その際に, 変化のあった記事である 5 件のうち, 2 件と残りの 3 件の閲覧数の合計が同じ, もしくは 2 件の閲覧数の合計が多くなるようにして, 記事の閲覧数の上位を足した合計と残りの閲覧数の合計が同じになるように基準を合わせる.

7. おわりに

課題は, バックアップデータのコンテンツ全体のテストを行うと, コンテンツを限定してテストを実行したときと比べ時間がかかることである. 提案は, WordPress の閲覧数に差がある記事を抽出し, バックアップデータに含まれている WordPress の記事や固定ページのテストを実行する対象を定めコンテンツを限定することで, テストの実行時間を短縮する. 評価では, 提案適用前である, CDSL の日本語サイトのコンテンツ全体をテストの実行する対象として定めた際のテストの実行時間, 閲覧数に差分のあった記事全体をテストの実行する対象として定めた際のテストの実行時間, 閲覧数に差分のあった記事のうちパレートの法則に当てはめ, 閲覧数に差分のあった値の上位 20%をテストの実行する対象として定めた際のテストの実行時間を比較した. 提案適用前であるコンテンツ全体をテストの実行対象として実行時間を計測した時間, 閲覧数に差分のあったコンテンツ全体を計測した時間, 閲覧数の差分の大きい値上位 20%のコンテンツを計測した時間をそれぞれ 10 回計測した. コンテンツの全数のテストを実行した際の時間は, 平均で約 123.46 秒であり, 閲覧数の差分のあったコンテンツ全体を検索した際の実行時間は, 平均で約 26.53 秒, 閲覧数の差分が大きい値上位 20%のコンテンツをテストの実行対象として定めた際, テストの実行時間は約 5.35 秒であった. コンテンツの全数と閲覧数の差分のあったコンテンツ全体のテストを実行した際は平均約 79%, コンテンツの全数と閲覧数に差分の大きい値上位 20%のテストを実行した際は平均約 96%実行時間を削減できた. テストの実行時間を削減することでバックアップデータに不備があった際, 原因の探索により早く取り組むことができる.

謝辞 本稿の執筆にあたりご助言を賜りました, 東京工科大学大学院バイオ・情報メディア研究科コンピュータサイエンス専攻の高橋 風太さん, 東京工科大学コンピュータサイエンス学部の山野 倖平さんに御礼申し上げます.

参考文献

[1] Akbar, R., Husain, M. S. and Suaib, M.: Comparative study of various backup and monitoring techniques, *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, pp. 1530–1537 (online), DOI: 10.1109/ICGCIoT.2015.7380710 (2015).

[2] Xia, R., Yin, X., Alonso Lopez, J., Machida, F. and Trivedi, K. S.: Performance and Availability Modeling of ITSystems with Data Backup and Restore, *IEEE Transactions on Dependable and Secure Computing*, Vol. 11, No. 4, pp. 375–389 (online), DOI: 10.1109/TDSC.2013.50 (2014).

[3] Cheng, H., Ho, Y. H., Hua, K. A., Liu, D., Xie, F. and Tsaur, Y.-P.: A Service-Oriented Approach to Storage Backup, *2008 IEEE International Conference on Services Computing*, Vol. 2, pp. 413–421 (online), DOI: 10.1109/SCC.2008.132 (2008).

[4] Tamimi, A. A., Dawood, R. and Sadaqa, L.: Disaster Recovery Techniques in Cloud Computing, *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, pp. 845–850 (online), DOI: 10.1109/JEEIT.2019.8717450 (2019).

[5] Ghazi, K. and H. O. Nasereddin, H.: Business Continuity Based on Backup, *American Academic and Scholarly Research Journal*, Vol. 5, pp. 253–258 (2013).

[6] Chervenak, A., Vellanki, V. and Kurmas, Z.: Protecting file systems: A survey of backup techniques, *Joint NASA and IEEE Mass Storage Conference*, Vol. 99, Citeseer (1998).

[7] Shriwas, M. S., Gupta, N. and Sinhal, A.: Efficient Method for Backup and Restore Data in Android, *2013 International Conference on Communication Systems and Network Technologies*, pp. 693–697 (online), DOI: 10.1109/CSNT.2013.148 (2013).

[8] Chang, V.: Towards a Big Data system disaster recovery in a Private Cloud, *Ad Hoc Networks*, Vol. 35, pp. 65–82 (online), DOI: https://doi.org/10.1016/j.adhoc.2015.07.012 (2015). Special Issue on Big Data Inspired Data Sensing, Processing and Networking Technologies.

[9] Ramesh, G., Logeshwaran, J. and Aravindarajan, V.: A Secured Database Monitoring Method to Improve Data Backup and Recovery Operations in Cloud Computing, *BOHR International Journal of Computer Science*, Vol. 2, pp. 1–7 (online), DOI: 10.54646/bijcs.019 (2023).

[10] Bruderer, H. and Vakulov, A.: The Dangers of Digitization, and the Importance of Data Backup, *Commun. ACM*, Vol. 67, No. 5, p. 23–25 (online), DOI: 10.1145/3643990 (2024).

[11] Ricca, F. and Tonella, P.: Analysis and testing of Web applications, *Proceedings of the 23rd International Conference on Software Engineering. ICSE 2001*, pp. 25–34 (online), DOI: 10.1109/ICSE.2001.919078 (2001).

[12] Suguna, S. and Suhasini, A.: Overview of data backup and disaster recovery in cloud, *International Conference on Information Communication and Embedded Systems (ICICES2014)*, pp. 1–7 (online), DOI: 10.1109/ICICES.2014.7033804 (2014).

[13] Mendonça, J., Lima, R., Queiroz, E., Andrade, E. and Kim, D. S.: Evaluation of a Backup-as-a-Service Environment for Disaster Recovery, *2019 IEEE Symposium on Computers and Communications (ISCC)*, pp. 1–6 (online), DOI: 10.1109/ISCC47284.2019.8969658 (2019).

[14] Bhattacharya, S., Mohan, C., Brannon, K. W., Narang, I., Hsiao, H.-I. and Subramanian, M.: Coordinating

© 2024 Cloud and Distributed Systems Laboratory

backup/recovery and data consistency between database and file systems, *Proceedings of the 2002 ACM SIGMOD International Conference on Management of Data*, SIGMOD '02, New York, NY, USA, Association for Computing Machinery, p. 500–511 (online), DOI: 10.1145/564691.564749 (2002).

[15] Yamashita, K., McIntosh, S., Kamei, Y., Hassan, A. E. and Ubayashi, N.: Revisiting the applicability of the pareto principle to core development teams in open source software projects, *Proceedings of the 14th International Workshop on Principles of Software Evolution*, IWPSE 2015, New York, NY, USA, Association for Computing Machinery, p. 46–55 (online), DOI: 10.1145/2804360.2804366 (2015).

[16] Grachev, G. A.: Pareto Principle.

[17] Gittens, M., Kim, Y. and Godwin, D.: The vital few versus the trivial many: examining the Pareto principle for software, *29th Annual International Computer Software and Applications Conference (COMPSAC'05)*, Vol. 1, pp. 179–185 Vol. 2 (online), DOI: 10.1109/COMPSAC.2005.153 (2005).

[18] Mendonça, J., Lima, R., Andrade, E., Araujo, J. and Kim, D. S.: Multiple-criteria Evaluation of Disaster Recovery Strategies Based on Stochastic Models, *2020 16th International Conference on the Design of Reliable Communication Networks DRCN 2020*, pp. 1–7 (online), DOI: 10.1109/DRCN48652.2020.1570614925 (2020).

[19] Suguna, S. and Suhasini, A.: Overview of data backup and disaster recovery in cloud, *International Conference on Information Communication and Embedded Systems (ICICES2014)*, pp. 1–7 (online), DOI: 10.1109/ICICES.2014.7033804 (2014).

[20] Swathi, B.: Automated Test Case Prioritization and Evaluation using Genetic Algorithm, *2022 International Conference on Computing, Communication, Security and Intelligent Systems (IC3SIS)*, pp. 1–5 (online), DOI: 10.1109/IC3SIS54991.2022.9885535 (2022).

[21] Gupta, K. and Goel, A.: Requirement checklist for blog in web application, *International Journal of System Assurance Engineering and Management*, Vol. 3 (online), DOI: 10.1007/s13198-012-0116-7 (2012).

© 2024 Cloud and Distributed Systems Laboratory