

Recenzja rozprawy doktorskiej mgr. Andrzeja Jackowskiego zatytułowanej “Adapting Distributed Storage with Deduplication to Cloud Use Cases”

Pan Andrzej Jackowski złożył rozprawę doktorską poświęconą zagadnieniom implementacji rozproszonych systemów danych z deduplikacją i ich adaptacji do środowiska chmurowych (cloud computing). Rezultaty rozprawy zaprezentowane zostały wcześniej w rozpoznawalnych i uznanych w środowisku międzynarodowym miejscach: w czasopiśmie czołowych w informatyce akademickiej organizacje ACM i IEEE: ACM Transactions on Storage i IEEE Transactions on Parallel and Distributed Systems oraz w recenzowanych sprawozdaniach znanej konferencji USENIX Conference on File and Storage Technologies, FAST 2023.

Poniżej pokrótce omawiam zawartość rozprawy koncentrując się na moim zdaniem głównych aspektach ją wyróżniających. Pełne omówienie zawartości bardzo dobrze zostało przedstawione w streszczeniu rozprawy a bardziej szczegółowo we wprowadzeniach i/lub podsumowaniach poszczególnych części rozprawy.

W rozdziałach 1–3 zapoznać się można z prezentacją problemu deduplikacji w przechowywaniu i archiwizacji danych wielkoskalowych, związanych z nią wyzwań a także przegląd dotychczasowych badań, rozwiązań i problemów szeroko rozumianego rozproszonego przechowywania i archiwizacji danych. Wraz z podrozdziałami 4.1, 5.1 i 6.1 materiał ten stanowi bardzo solidne wprowadzenie i omówienie tematyki, z ogromną ilością odwołań do literatury, która w efekcie składa się z aż 351 pozycji. Chciałbym też zaznaczyć, że tę część tekstu czyta się z zainteresowaniem i uważam, że stanowi przystępne a zarazem profesjonalne wprowadzenie w tematykę, a do zrozumienia wystarczy podstawowa wiedza informatyczna.

Główne wyniki rozprawy opisane są w rozdziałach 4, 5 i 6 a w krótkim rozdziale 7 znajduje się podsumowanie wyników rozprawy wraz z sugestiami odnośnie ich konsekwencji i znaczenia oraz dalszych kierunków badań.

Rozdział 4 poświęcony jest zagadnieniu obiektowego dostępu do danych, które przechowywane są z użyciem deduplikacji zaimplementowanej na bardziej niskopoziomowej warstwie “bloków”. W tym celu zaprojektowany został ObjDedup, który z punktu widzenia “końcowego” użytkownika (programisty) jest warstwą/biblioteką oferującą interfejs dostępu do danych, również przechowywanych w chmurze. Istotnym problemem w tym problemie jak i generalnie w przypadku zastosowania deduplikacji jest rozmiar metadanych i wpływ efektywności dostępu do nich określonych przez czas i ilość obliczeń na wydajność systemu (aby oszczędności pamięci wynikające z deduplikacji nie zostały „zarnowane”). Bardzo pozytywnym elementem sposobu podejścia do rozważanego problemu, który daje się zauważyć, nie tylko w tym rozdziale, jest próba zidentyfikowania czy postawiony cel można realizować przy pomocy innych narzędzi stworzonych na potrzeby dostępnych komercyjnie lub dostępnych w otwartym dostępie (ang. open source) systemów chmurowych i lokalnych

magazynów danych¹. A także dobór danych i eksperymentów do porównań tak, aby z góry nie faworyzowały własnych rozwiązań. Szczegółowe rozwiązanie, dopasowane do specyfiki wymagań zostało przystępnie i przekonująco opisane w rozdziale 4 rozprawy, a uzyskane wyniki potwierdzają na różnych systemach pamięci rozproszonych, w tym chmurowych, że uzyskany wynik daje korzyści zarówno w zakresie przepustowości jak i czasu przetwarzania.

W rozdziale 5 opisany został drugi element o nazwie InftyDedup, w tym przypadku chodzi głównie o efektywną archiwizację danych w systemie „chmurowym”, który nie zapewnia deduplikacji. InftyDedup ma umożliwić archiwizację z deduplikacją, zapewniając oczywiście oszczędności pamięci, którą potencjalnie deduplikacja pozwala osiągnąć, bez utraty jakości innych parametrów. Ponownie pojawia się tutaj szereg wyzwań związanych z rozmiarem i koniecznością szybkiego dostępu do metadanych, ale także bardzo zróżnicowanymi systemami opłat za składowane dane, dostęp do nich, przesyłanie oraz wykonywane na nich obliczenia, wzorce dostępu do archiwów. Podobnie jak w rozdziale 4 przedstawiono rozwiązanie z bardzo dobrze uzasadnionymi wyborami projektowymi i rozważeniem, czy przedstawiona propozycja stanowi korzystną alternatywę wobec usług już dostępnych komercyjnie bądź jako projekty akademickie. Ciekawe jest selektywne wykorzystywanie alternatywnych „hot” i „cold” storage. Duże wrażenie zrobiło na mnie uwzględnienie wielu aspektów wpływających na efektywność rozwiązań, zarówno technicznych jak i związanych np. z polityką cenową rynkowych dostawców usług.

W rozdziale 5 wprowadzono też przypominającą B+-drzewa strukturę danych nazwaną OMT-drzewami (ObjectMetadataTree) z operacją scalania, wprowadzenie tej struktury zostało uzasadnione a jej podstawowe własności przeanalizowane.

Zaprojektowane i zaimplementowany system InftyDedup został też kompleksowo przetestowany i wykazano jego liniową skalowalność ze względu na rozmiar składowanych danych, z zapewnieniem akceptowalnego czasu dostępu i przetwarzania, a także oszczędności rozmiaru pamięci wynikającej z deduplikacji. Ze względu na brak innych produktów o podobnych celach utrudnione były porównania z innymi systemami. Mimo wszystko udało się znaleźć rozsądne metody porównań i ewaluacji. W mojej ocenie niedosyt powoduje jedynie wybór tych aspektów, w których eksperymenty prowadzono z myślą o dużych systemach i zbiorach danych oraz tych, gdzie skupiono się na mniejszych zasobach.

Wreszcie trzecia i zarazem ostatnia część wyników rozprawy o nazwie Derrick opisana w rozdziale 6 stanowi w pewnym sensie uzupełnienie dwóch poprzednich. Zadaniem systemu Derrick jest wykonywanie okresowych „migracji” danych, które mają zwiększyć odporność na błędy, niezawodność dostępu do nich, a także wykorzystanie zasobów (pamięci) i szybkość dostępu. Cechą wyróżniającą ten element na tle dwóch pozostałych jest precyzyjne sformułowanie go jako rodziny problemów optymalizacyjnych. W pierwszej kolejności celem jest wyznaczenie nowego rozmieszczenia danych optymalizującego różne metryki, czasem ze sobą sprzeczne. W tym przypadku chodzi głównie o zbalansowanie wykorzystania pamięci (np. podobna zajętość różnych urządzeń) i utrzymania niezawodności (różne kopie tych samych danych na różnych nośnikach fizycznych). Ale w praktyce liczba metryk jest dużo większa i konieczne jest rozważanie ich jako przestrzeni „wektorów”, w której dążymy do minimalizacji np. wg porządku leksykograficznego. Rozwiązanie optymalne wymagałoby prawdopodobnie

¹ Nie jest to tak oczywiste jak w przypadku badań teoretycznych, gdzie rozwiązuje się problemy o dość zwięzłym sformułowaniu wyrażonym w sformalizowanej terminologii danej dziedziny.

wykładniczych obliczeń, dlatego stosuje się odpowiednio dobrane heurystyki (w tym przypadku algorytmy z rodziny *hill climbing*).

Drugim elementem Derricka jest algorytm dla innego ciekawego problemu optymalizacyjnego: poszukiwania sposobu migracji danych (sekwencji kroków) o minimalnym koszcie, prowadzącym do uzyskania nowego rozmieszczenia danych, których oryginalne lokalizacje stanowią dane wejściowe do problemu. W rozprawie zaproponowany został nie tylko sam problem i koncepcja jego heurystycznego rozwiązania, ale także formalizacja w terminologii matematycznej i sformułowanie kluczowych własności następnie formalnie udowodnionych. (Np. możliwość ograniczenia przestrzeni „jednokrokowych” zmian rozmieszczenia danych z zachowaniem możliwości uzyskania końcowego rozwiązania). Ten element wyróżnia prezentowane badania – nieczęsto się zdarza, że wyniki nastawione na weryfikację eksperymentalną w rzeczywistych systemach poparte są też formalnymi dowodami ich własności o potencjalnie szerszym zastosowaniu. Derrick zawiera też trzeci algorytm służący do migracji danych w sytuacjach wyjątkowych – awarii urządzeń przechowujących dane.

Uwagi

Z uwagi na kluczowe znaczenie implementacji komputerowej zaprojektowanych rozwiązań, przygotowania odpowiednich zestawów danych i narzędzi analizy wyników eksperymentów na tych danych i symulacji, istotny element pracy nad rozwiązaniem postawionych problemów i weryfikacją jego jakości stanowią kody źródłowe utworzonych narzędzi, czy dostępne zestawy danych. (Inaczej niż np. w informatyce teoretycznej, gdzie treść publikacji i badań zazwyczaj w pełni dokumentuje wyniki pracy nad postawionymi i rozwiązywanymi problemami.) Z uwagi na komercyjny charakter systemu HYDRAsTOR, dla którego tworzone były przedstawione narzędzia, pełen dostęp do kodów źródłowych nie może w tym przypadku mieć miejsca. Niemniej, interesująca dla czytelnika byłaby wiedza na temat potencjalnej dostępności użytych danych testowych i utworzonych narzędzi do weryfikacji, porównywania bądź wizualizacji wyników. A bardziej ogólnie możliwości samodzielnej weryfikacji przedstawionych wyników badań. Sama rozprawa i publikacje naukowe, na których jest oparta nie dostarczają takich możliwości.

Formułując powyższą uwagę chciałbym zaznaczyć, że poddając swoje wyniki ocenie fachowców z komitetów redakcyjnych branżowych czasopism i komitetów programowych konferencji, autorzy uzyskali potwierdzenie wartości swoich wyników z najbardziej zainteresowanej i kompetentnej strony. Nie mam więc zastrzeżeń do wiarygodności uzyskanych wyników, zwracam jedynie uwagę na kwestię reprodukowalności wyników i dostępu do danych, ponieważ dobre praktyki w tym zakresie służą też lepszej rozpoznawalności i uznaniu wyników oraz pozwalają na szybszy rozwój badań w danym obszarze.

Wskazane powyżej duże znaczenie implementacji przedstawionych rozwiązań i koncepcji wiąże się też z liczbą autorów publikacji naukowych, na których oparta jest rozprawa (5 w ACM Trans. on Storage, 9 w IEEE Trans. Parall. and Distr. Systems, 6 na FAST). Ta dość duża liczba autorów publikacji, na których oparta jest rozprawa może budzić wątpliwości co do „udziału jednostkowego” doktoranta. Jednak trzy publikacje będące podstawą treści rozprawy nie odzwierciedlają rozmiaru projektów, które prowadziły nie tylko do wypracowania koncepcji rozwiązań postawionych problemów, ale też implementacji i eksperymentalnej ewaluacji trzech narzędzi tworzących końcowe rozwiązanie. Niemniej, w przypadku ewentualnej dyskusji nad wyróżnieniem rozprawy, istotne może być poznanie własnego wkładu doktoranta w przedstawione w rozprawie wyniki.

Od strony technicznej sędzę, że docenić warto fakt, że pomimo czysto aplikacyjno-eksperymentalnego charakteru prowadzonych badań, włożony został też pewien wysiłek

w analizę złożoności i poprawności proponowanych rozwiązań algorytmicznych w terminologii matematycznej jak i formalizacja części z postawionych problemów jako problemów algorytmicznych i złożonościowych. Dotyczy to szczególnie rozdziału 6, ale elementy analizy złożoności obliczeniowej wykorzystywanych struktur danych i algorytmów są konsekwentnie prowadzone również w rozdziale 4 i 5.

Ciekawe jest wprowadzenie i implementacja uogólnienia B+-drzew nazwanego OMT-trees (ObjectMetadataTree) z operacją scalania, uwzględniającego specyfikę rozważanych zastosowań. W kontekście (pod)problemów dotyczących wyboru nośnika na przechowywane/składowane dane o różnych charakterystykach czasów odczytu, zapisu, dostępu itp. chciałbym zarazem zwrócić uwagę na obszar badań algorytmicznych nazywany szeroko *cache-oblivious algorithms and data structures*. Historycznie nazwa tych struktur odwołuje się tylko do korzystania z pamięci podręcznej procesora (ang. *cache memory*), niemniej rozwiązania adaptują się do wielopoziomowych hierarchii różnych typów pamięci. Przypuszczam, że niektóre rozwiązania z tego obszaru mogą być inspirujące w problemach rozważanych np. w rozdziale 4 czy 5, a przynajmniej warto się odnieść do stosowalności tych technik lub braku takich zastosowań, np. przy rozważaniach o wyborze między „hot” i „cold” cloud storage.

Ocena rozprawy

Odnosząc się do formalnego wymogu ustawowego mówiącego, że *rozprawa doktorska prezentuje ogólną wiedzę teoretyczną kandydata w dyscyplinie albo dyscyplinach* nie mam wątpliwości, że rozdziały 1 – 3 oraz 4.1, 4.2 i 4.3 dowodzą spełnienia tych warunków przez rozprawę. Natomiast wymóg *umiejętności samodzielnego prowadzenia pracy naukowej lub artystycznej* jest z nawiązką wypełniony przez uzyskane wyniki z rozdziałów 4 – 6 i ich prezentację w rozprawie. W szczególności rozprawa przedstawia „oryginalne rozwiązanie problemu naukowego, oryginalne rozwiązanie w zakresie zastosowania wyników własnych badań naukowych w sferze gospodarczej”, czyli zbiór rozwiązań kompleksowo adresujący kluczowe aspekty przechowywania i archiwizacji danych wielkoskalowych z użyciem deduplikacji w centrach danych oraz w ramach dostępnych usług „chmurowych” (cloud storage).

W mojej ocenie wyniki rozprawy można ocenić jako bardzo dobre a może wręcz wyśmienite, czego potwierdzeniem są zarówno dobre miejsca ich publikacji, implementacja i wdrażanie w komercyjnym systemie HYDRAsor, kompleksowa ewaluacja z wykorzystaniem rozległych eksperymentów na rzeczywistych systemach i danych (np. z FSL) oraz symulacji jak i czytelna prezentacja wyników tych eksperymentów w rozprawie. Na uwagę zwraca fakt, że projektując eksperymenty porównujące wydajność różnych systemów starano się uwzględnić ewentualne przewagi własnego rozwiązania wynikające wprost z przyjętych założeń i priorytetów. Użyte dane i struktura eksperymentów bądź symulacji zostały tak dobrane, aby neutralizować ewentualne naturalne przewagi własnych rozwiązań. Sądzę, że docenić też warto formalizację problemów rozważanych w rozdziale 6 problemów w terminologii matematyczno-algorytmicznej i wykazanie poprawności zastosowanych rozwiązań w formalnej metodologii dowodów matematycznych.

Wyróżniającą cechą złożonej rozprawy jest bardzo dobra jakość prezentacyjno-edycyjna co świadczy o profesjonalnej redakcji rozprawy. Rozprawa napisana jest nienagannym, bogatym ale zrozumiałym językiem, trudno jest znaleźć nawet drobne błędy językowe, literówki, omyłki językowe bądź techniczne.

Podsumowanie

Uważam, że rozprawa doktorska Pana Andrzeja Jackowskiego jest bardzo dobra a może nawet wyśmienita w każdym aspekcie, zarówno formalnym jak i merytorycznym. Dotyczy to zarówno samych koncepcji rozwiązań opisanych w rozprawie jak i ich implementacji w rzeczywistym systemie działającym jako komercyjny produkt. Bez wątpienia zatem rozprawa spełnia ustawowe i zwyczajowe wymagania stawiane doktoratom, w szczególności wymogi dla uzyskania stopnia *doktora w dziedzinie nauk ścisłych i przyrodniczych, w dyscyplinie informatyka*. Wnioskuje bez wahania o **pozytywne** zaopiniowanie rozprawy i dopuszczenie do dalszych etapów postępowania w sprawie nadania stopnia doktora.



Signed by /
Podpisano przez:

Tomasz Jurdziński

Date / Data:
2024-05-06
12:49

