ONTAP i TrueNAS

Jak trzej założyciele Network Appliance zbudowali „toster” do NFS-u z systemem plików, który nigdy nie nadpisuje danych w miejscu, jak ten sam pomysł stanął przed sądem w sporze z Sunem o ZFS — i jak po drugiej stronie rynku hobbystyczny FreeNAS przeszedł z FreeBSD na Linuksa i stał się TrueNAS-em.

Ta strona opowiada o dwóch rodzinach systemów, które administrator spotyka wtedy, gdy dyski przestają być częścią serwera, a stają się osobnym urządzeniem w szafie. Pierwsza to ONTAP, system macierzy NetApp, który zaczął się w 1992 roku jako oprogramowanie jednozadaniowego serwera plików, a dziś działa na klastrach kontrolerów, w chmurze Amazona i Microsoftu i — co mało kto wie — na FreeBSD. Druga to FreeNAS i jego następca TrueNAS: projekt jednego człowieka, przejęty przez firmę sprzedającą serwery z FreeBSD, który w latach 2022–2025 przeniósł się na Debiana. Między nimi stoi ZFS, system plików Suna, o który NetApp w 2007 roku wytoczył proces — zakończony trzy lata później bez wyroku. Najwięcej miejsca zajmuje to, co z tego wynika w 2026 roku: cykle wsparcia obu systemów, aktualizacja klastra bez przestoju, migracja z TrueNAS CORE, zabezpieczenia przed ransomware i to, ile pamięci i jakiego kontrolera ZFS potrzebuje w rzeczywistości, a ile według forów.

> Umiejscowienie w serii: to strona księgi „Serwerownia”, o systemach pamięci masowej. Skąd się wziął ZFS — prace w Sunie, Bonwick i Moore, OpenZFS — opisuje strona **Solaris i SunOS**; tutaj biorę z niej tylko tyle, ile trzeba do sporu z NetApp i do historii FreeNAS. FreeBSD, na którym stoi ONTAP i na którym przez dekadę stał FreeNAS, ma własną stronę **FreeBSD**, a systemy BSD w urządzeniach — macierzach, routerach, konsolach — zbiera strona **BSD w praktyce**. ZFS w innych systemach BSD pojawia się na stronie **NetBSD i DragonFly BSD**, a jądro, na którym działa dziś TrueNAS, na stronie **Linux i GNU**. Macierze NetApp i serwery TrueNAS najczęściej obsługują hipernadzorców — o nich jest strona **VMware ESX i ESXi** — a kontenery, które TrueNAS uruchamia jako „aplikacje”, opisuje strona **Kontenery: od chroot do Kubernetesa**. Inne systemy urządzeń sieciowych z tej samej księgi to **Cisco IOS i Junos** oraz **FortiOS i RouterOS**.

Skróty użyte na tej stronie (87)

Skrót	Co znaczy
8-K	raport bieżący spółki składany w SEC
A-SIS	deduplikacja w Data ONTAP 7G i 8
ACL	Access Control List: lista uprawnień do pliku
AFF / ASA / FAS	rodziny macierzy NetApp (np. FAS270C, FAS3050, FAS6070, FAS8200)

Skrót	Co znaczy
AFP	Apple Filing Protocol: protokół udostępniania plików Apple
AGD	artykuły gospodarstwa domowego
AMD	producent procesorów Opteron
ANDU	automatyczna aktualizacja ONTAP bez przestoju
API / REST	interfejs programistyczny; REST — interfejs przez HTTP
ARC / ZIL	pamięć podręczna ZFS w RAM-ie i dziennik zapisów synchronicznych ZFS
ARP / ARP/AI	Autonomous Ransomware Protection: wykrywanie ransomware w ONTAP; ARP/AI — model od 9.16.1
ATA	interfejs tańszych dysków, używanych m.in. w NearStore
ATG	dział NetApp, na którego stronie są publikacje techniczne
AWS / EC2 / EBS / GP2	Amazon Web Services, jego maszyny wirtualne i dyski blokowe; GP2 — typ dysku EBS
BASIC	język programowania
BSD	Berkeley Software Distribution: rodzina systemów uniksowych i licencja
BYOL	bring your own license: własna licencja w chmurze
CE	Community Edition: darmowe wydanie TrueNAS od 25.04
CHM	Computer History Museum w Mountain View
CIFS / SMB	protokół udostępniania plików Windows; CIFS to jego dawna odmiana
CVE	numer luki w rejestrze podatności
CVO	Cloud Volumes ONTAP: ONTAP w AWS i Azure
ECC	pamięć z korekcją błędów
EDL	inicjały sędzi w sygnaturze sprawy NetApp przeciw Sunowi
ESF	Enterprise Storage Forum: serwis branżowy
ESX	hipernadzorca VMware
FAST	USENIX Conference on File and Storage Technologies
FTP	protokół przesyłania plików
GA / RC / P-release	wydanie ogólnie dostępne, kandydujące i poprawkowe
GAAP	amerykańskie zasady rachunkowości
GB / MB / KB / TB / PB / EB	giga-, mega-, kilo-, tera-, peta- i eksabajt
GELI	szyfrowanie dysków we FreeBSD

Skrót	Co znaczy
GEOM	podsystem warstw dyskowych FreeBSD
GNU	projekt wolnego oprogramowania
GX	Data ONTAP GX: klastrowa linia ONTAP z lat 2006–2009
HA	high availability: wysoka dostępność, para kontrolerów z przełączaniem
HBA	kontroler dysków bez funkcji RAID
HDS / HP / IBM / DS / ESS / XP	producenci i macierze wirtualizowane przez gFiler (np. IBM DS4000)
HEAD	bieżąca gałąź rozwojowa FreeBSD
HPC	obliczenia wielkiej skali
I/O / IOPS	wejście-wyjście i liczba operacji na sekundę
ILM	Industrial Light & Magic: studio efektów specjalnych
IOS	system Cisco
ISO	obraz płyty instalacyjnej
JSON	tekstowy format danych
KVM / QEMU	hipernadzorca w jądrze Linuksa i emulator maszyn
LADDIS / SPEC SFS	test wydajności serwerów NFS
LFS	Log-Structured File System
LIDAR	pomiar terenu impulsami lasera
LTD	Limited: w nazwie firmy Wheel LTD
LTS	long-term support: wydanie z długim wsparciem
LUN	wolumin blokowy udostępniany przez SAN
LWN	LWN.net: serwis o Linuksie
LXC	Linux Containers: kontenery systemowe
MIPS	MIPS Computer: producent procesorów, dawny pracodawca Hitza
MTTDL	średni czas do utraty danych
NAS / SAN	pamięć dostępna przez sieć plikowo i blokowo
NASA / NBC	klienci iXsystems
NFS	Network File System: uniksowy protokół plików sieciowych
NIS	Network Information Service: katalog użytkowników
NTAP	przedrostek numerów biuletynów bezpieczeństwa NetApp

Skrót	Co znaczy
NVMe / RDMA / TCP	interfejs dysków SSD; dostęp do nich przez sieć RDMA albo TCP
NVRAM	pamięć nieulotna na dziennik żądań
PEKA	system biletowy ZTM w Poznaniu
PHP	język skryptowy, w którym napisano FreeNAS 0.7
RAID / RAID-DP / RAID-TEC / RAIDZ	macierz dysków z redundancją; podwójna i potrójna parzystość NetApp; grupy z parzystością w ZFS
RAM	pamięć operacyjna
RDP	Row-Diagonal Parity: algorytm RAID-DP
SAS	interfejs dysków serwerowych
SCC / VCC	zegary zgodności SnapLock: systemowy i wolumenowy
SEC	Securities and Exchange Commission: amerykańska komisja papierów wartościowych
SED	dysk samoszyfrujący
SUSE	dystrybucja Linuksa
SVM / DR	maszyna wirtualna z danymi w klastrze ONTAP; odtwarzanie po awarii
TPM	moduł kryptograficzny
TR	Technical Report: raport techniczny NetApp (TR3002)
USB	pamięć przenośna
USD	dolar amerykański
USENIX	stowarzyszenie organizujące konferencje o systemach
USPTO	amerykański urząd patentowy
VSA	virtual storage appliance: wirtualne urządzenie pamięci masowej
WAFL	Write Anywhere File Layout: system plików ONTAP
WORM	write once, read many: zapis jednokrotny
WWW	interfejs przez przeglądarkę
XOR	alternatywa wykluczająca: operacja liczenia parzystości
ZFS	system plików Suna, dziś OpenZFS
ZTM	Zarząd Transportu Miejskiego w Poznaniu

□□ Leksykon: nazwy i pojęcia z tej strony (42)

Nazwa	Co to jest
agregat	zbiór dysków ONTAP, na którym tworzy się woluminy FlexVol
Angelfish … Goldeye	nazwy wydań TrueNAS: 22.02, Bluefin, 24.04 Dragonfish, 24.10 Electric Eel, 25.04 Fangtooth, 25.10 Goldeye
AutoSupport	mechanizm powiadomień ONTAP wysyłanych do NetApp
bhyve	hipernadzorca FreeBSD, oddany przez inżynierów NetApp
Cloud ONTAP	ONTAP jako instancja EC2 (2014), później CVO
Community Edition	linuksowy TrueNAS od 25.04, wspólny kod z Enterprise
copy-on-write	zapis zmian w nowym miejscu zamiast nadpisywania
CORE	TrueNAS na FreeBSD (12.0–13.x), od 2026 roku bez aktualizacji
Data ONTAP 7G	wydanie z 2004 roku z FlexVol i FlexClone
Fast Dedup	szybsza deduplikacja w OpenZFS 2.3
filer	serwer plików NetApp
FlexClone	zapisywalny klon woluminu bez kopiowania danych
FlexGroup	wolumin ONTAP rozłożony na wiele agregatów
FlexVol	wolumin oderwany od fizycznych dysków
FreeNAS	system NAS na FreeBSD, od 2005 roku; od 12.0 TrueNAS CORE
gFiler	urządzenie NetApp wirtualizujące cudze macierze FC
i-węzeł główny	korzeń drzewa bloków WAFL; jego kopia to migawka
Incus	narzędzie do kontenerów i maszyn wirtualnych w TrueNAS 25.04 („Instances”)
jail	izolowane środowisko FreeBSD, używane przez wtyczki CORE
Klara Systems	firma wspierająca NetApp przy FreeBSD i TrueNAS przy OpenZFS
m0n0wall	zapora, z której kodu wyrósł FreeNAS
MetroCluster	synchroniczna replikacja ONTAP między dwiema lokalizacjami
migawka (snapshot)	stan systemu plików z danej chwili
multi-admin verification	zatwierdzanie operacji przez kilku administratorów
NAS4Free / XigmaNAS	kontynuacja FreeNAS 0.7 na FreeBSD
NearStore	macierze NetApp na dyskach ATA
ONTAP	system operacyjny macierzy NetApp (Data ONTAP do 2016 roku)
ONTAP One / ONTAP Select	pakiet licencji ONTAP; ONTAP w maszynie wirtualnej

Nazwa	Co to jest
OpenMediaVault	system NAS na Debianie Volkera Theile
OpenZFS	wspólny projekt ZFS-a dla wielu systemów
punkt spójności	chwila zapisania nowego i-węzła głównego WAFL
resilvering	odbudowa danych na wymienionym dysku ZFS
SCALE	linuksowy TrueNAS 2022–2025, potem Community Edition
scrub	przegląd całej puli ZFS z kontrolą sum
shadow paging	zapis zmienionych stron obok starych w bazach danych
SnapLock	woluminy WORM w ONTAP: Compliance i Enterprise
SnapMirror	replikacja migawek ONTAP
Spinnaker Networks	firma z Pittsburgha, kupiona w 2004 roku, źródło klastra ONTAP
System Manager	interfejs graficzny ONTAP
tamperproof snapshot locking	blokada migawek przed usunięciem, od 9.12.1
TrueNAS Connect	usługa iXsystems: Foundation, Plus, Business
WebShare / TrueSearch	udostępnianie plików przez przeglądarkę i wyszukiwanie w TrueNAS 26

> Zastrzeżenie zakresu: nie jest to podręcznik administracji ONTAP ani TrueNAS. Polecenia podaję tylko te, które przytacza dokumentacja NetApp albo OpenZFS, z oznaczeniami w nawiasach ostrych tak, jak stoją w dokumentacji; nie dopisuję do nich własnych przełączników. Liczby użytkowników, wydajności i udziałów rynkowych pochodzą przeważnie od producentów i tak są opisane. Opis zabezpieczeń przed ransomware jest opisem mechanizmów obronnych według dokumentacji, a nie instrukcją ich obchodzenia.

Wprowadzenie: toster, lodówka i system plików

W kwietniu 1992 roku powstała firma, której założyciele chcieli zrobić z serwerem plików to, co producenci sprzętu AGD zrobili z kuchnią: urządzenie, które robi jedną rzecz, robi ją dobrze i nie wymaga, żeby ktoś rozumiał, co jest w środku. Nazwali ją Network Appliance — „urządzenie sieciowe”. Pierwszy serwer, według firmowej strony historii, miał w zespole przydomek „Toaster”, toster. Dziewięć lat później prezes firmy, Daniel Warmenhoven, opisywał produkt jeszcze bardziej

domowo: „We built a data refrigerator. Open the door. Take something out. Close the door.” — zbudowaliśmy lodówkę na dane: otwierasz drzwi, wyjmujesz, zamykasz.

Pod tym obrazem kryło się coś, czego klienci według samych założycieli nie chcieli słuchać. Dave Hitz, jeden z nich, pisał po latach o swoim systemie plików: „Absolutely wonderful technology that our non-technical customers don't really want to hear about” — technologia wspaniała, o której nietechniczni klienci nie chcą słyszeć. Ten system plików nazywał się WAFL, Write Anywhere File Layout, i był opisany w artykule na konferencji USENIX w styczniu 1994 roku. Nigdy nie nadpisywał danych w miejscu, zapisywał nowe bloki tam, gdzie było mu wygodnie, i dzięki temu robił migawki — kopie całego systemu plików z danej chwili — prawie bez kosztu. Na dziesięciolecie firmy Hitz nazwał go swoim największym osiągnięciem.

Od tego artykułu prowadzą dwie drogi, które ta strona idzie obok siebie. Pierwsza to droga samego NetAppa: od pojedynczego filera z jednym woluminem przez kupno firmy Spinnaker i jedenaście lat łączenia dwóch linii kodu po klastrowy ONTAP 9, który w 2026 roku wychodzi dwa razy w roku, działa jako usługa w chmurze Amazona i według Fundacji FreeBSD stoi na FreeBSD 14. Druga droga zaczyna się w Sunie. W 2004 roku Sun zapowiedział ZFS — system plików, który też nigdy nie nadpisuje danych w miejscu i też robi tanie migawki. Jeden z czytelników Slashdota już w 2004 roku, w dyskusji pod informacją o zapowiedzi, porównał go do WAFL. W 2007 roku NetApp pozwał Suna o naruszenie siedmiu patentów. Spór trwał trzy lata, przeszedł razem z Sunem do Oracle'a i skończył się obustronnym wycofaniem pozwów, bez wyroku i z poufnymi warunkami.

ZFS tymczasem poszedł własną drogą. Polski programista Paweł Jakub Dawidek przeniósł go w 2006 i 2007 roku do FreeBSD, a na FreeBSD z ZFS-em zbudowano FreeNAS — system do domowego i firmowego serwera plików, który zaczynał jako projekt jednego człowieka na bazie zapory m0n0wall. Firma iXsystems przejęła go w 2009 roku, przepisała od zera, zrobiła z niego komercyjny TrueNAS, a potem — między 2022 a 2025 rokiem — przeniosła z FreeBSD na Debiana. W kwietniu 2025 roku linuksowa odmiana stała się jedynym kodem, z którego powstają i darmowa Community Edition, i płatne urządzenia Enterprise.

Obiegowa wersja tej historii brzmi mniej więcej tak: „Trzech facetów z Auspexu zrobiło w 1992 roku toster do NFS-u na BSD; ich WAFL był systemem log-structured, a Sun zerżnął z niego ZFS, za co NetApp go pozwał i wygrał. FreeNAS to hobbystyczny ZFS na FreeBSD, który iXsystems porzuciło na rzecz Linuksa, a ZFS i tak wymaga pamięci ECC i gigabajta RAM-u na terabajt.” W tym zdaniu prawie każdy człon trzeba rozebrać. Z Auspexu pochodziło dwóch założycieli, o trzecim nie wiem. WAFL nie był log-structured — jego autorzy porównywali go do zupełnie innych mechanizmów. Nikt nie wygrał, bo nie było wyroku. Reguła gigabajta na terabajt jest heurystyką z dokumentacji FreeNAS 8, a ECC dokumentacja OpenZFS „zdecydowanie zaleca”, ale nie wymaga — i z tego samego powodu zaleca ją każdemu innemu systemowi plików. Rozbieram tę opowieść w tabeli pod koniec strony.

W tej księdze ważniejsza od historii jest praktyka: cykle wsparcia, aktualizacje, blokowanie migawek, migracja z TrueNAS CORE i sprzęt pod ZFS. Zbiera ją rozdział o praktyce, oparty na dokumentacji NetApp, iXsystems i OpenZFS.

Network Appliance: serwer, który umie tylko NFS

Trzech założycieli czy dwóch

Kto założył Network Appliance, zależy od tego, które źródło się czyta. Kompilacja historii firm Reference for Business, oparta na raportach spółki i prasie, podaje: firmę założono w kwietniu 1992 roku, a założycielami byli **Michael Malcolm**, **David Hitz** i **James Lau**. Tę samą trójkę wymienia artykuł z konferencji USENIX z 1994 roku, podpisany przez wszystkich trzech jako ludzi „Network Appliance”. Firmowa strona historii NetApp przy roku 1992 pisze natomiast tylko o Hitzu i Lau, którzy budują serwer plików nazywany „Toaster”; Malcolma nie wymienia. Profil Hitza w Computer History Museum też mówi o dwóch[1]. Nie jest to spór o fakty, tylko o pamięć: Malcolm przestał być prezesem w październiku 1994 roku, kiedy zastąpił go Daniel Warmenhoven, a Hitz w swojej książce wspomnieniowej opisuje „bolesne usunięcie” współzałożyciela.

Dwóch z trzech przyszło z jednego miejsca. Według Reference for Business Lau był w firmie Auspex dyrektorem rozwoju oprogramowania, a Hitz — starszym inżynierem od systemów plików i mikrojądra. Auspex pozostał dla Network Appliance punktem odniesienia — w artykule z 1994 roku to między innymi z jego systemem porównywano wydajność nowego serwera. Przed Auspexem Hitz pracował w MIPS Computer nad systemem plików i wejściem-wyjściem dla jądra System V, a studiował informatykę i elektrotechnikę w Princeton. Computer History Museum dodaje szczegół, który dobrze pasuje do tytułu jego późniejszej książki: zanim zajął się komputerami, pracował jako kowboj. Skąd przyszedł Malcolm, w zebranych materiałach nie ustaliłem.

„Toster”: przydomek, nie produkt

Słowo „toster” krąży w opowieściach o NetApp tak często, że łatwo wziąć je za nazwę produktu. Jedynym źródłem pierwotnym, jakie mam, jest firmowa strona historii: „network file server nicknamed the Toaster” — serwer plików nazywany tosterem. W artykule USENIX tego słowa nie ma, a recenzja książki Hitza w The Register też go nie przytacza. Analogia sprzętu domowego była jednak w firmie żywa: sama nazwa „appliance”, lodówka Warmenhovena z 2001 roku i definicja z artykułu z 1994 roku: „An appliance is a device designed to perform a particular function” — urządzenie, które ma spełniać jedną określoną funkcję. W przypadku pierwszego filera tą funkcją był NFS i nic poza nim.

Taki był też cel, który Hitz opisał w pożegnalnym wpisie z lutego 2019 roku: zbudować „the simplest possible NFS system” — najprostszy możliwy serwer NFS — i uprościć pamięć masową tak, jak Cisco uprościło sieci. Świat urządzeń sieciowych, do którego Hitz się odwoływał, opisuje w tej samej księdze strona **Cisco IOS i Junos**. Tutaj ważny jest sam zamiar: zamiast serwera ogólnego przeznaczenia, na którym udostępnianie plików jest jedną z wielu usług, dać urządzenie, które poza

udostępnianiem plików nie robi nic.

Pierwsze filery i pieniądze

Według Reference for Business założyciele zebrali 12,8 mln dolarów kapitału prywatnego i od funduszy venture capital, a pierwsze urządzenia — w firmie nazywane „filerami” — wysłano klientom w czerwcu 1993 roku. Liczby z pierwszych lat, przytaczane za raportami spółki, pokazują firmę, która rosła szybko z bardzo niskiego poziomu: w roku obrotowym 1994 przychody wyniosły 2,2 mln dolarów przy stracie 1,8 mln, w 1995 — 14,7 mln przy stracie 4,7 mln, w 1996 — 46,6 mln, już z zyskiem 6,6 mln. W listopadzie 1995 roku Network Appliance weszło na giełdę i zebrało 36 mln dolarów, w kwietniu 1999 roku druga emisja przyniosła 123 mln. W 1998 roku przychody przekroczyły 166 mln dolarów, w 2003 — 892,1 mln.

Instalacja filera NetApp FAS270 z 2 TB pamięci

Instalacja filera NetApp FAS270 z 2 TB pamięci. Źródło: Flickr, CC BY-NC 2.0, autor: Matteo Castelli.

Firmowa strona historii wylicza kolejne kamienie milowe skrótowo, rok po roku: 2000 — SnapMirror, czyli replikacja migawek na inną macierz; 2002 — pierwsze systemy łączące w jednym urządzeniu dostęp plikowy (NAS) i blokowy (SAN); 2004 — cienkie przydzielanie miejsca (thin provisioning) i FlexClone; 2007 — deduplikacja zarówno dla pamięci pierwotnej, jak i wtórnej; 2008 — zmiana nazwy z Network Appliance na NetApp; 2012 — zunifikowany SAN i NAS w architekturze scale-out; 2015 — George Kurian zastępuje Toma Georgensa na stanowisku prezesa, a firma kupuje SolidFire; 2018 — pierwsza macierz z NVMe od początku do końca, reklamowana jako „over 1.3 million IOPS at 500 microseconds per HA pair”. To ostatnie jest liczbą producenta. Przejęcia Spinnakera z 2004 roku, które ukształtowało dzisiejszy ONTAP bardziej niż którekolwiek z wymienionych zdarzeń, strona historii nie wymienia wcale.

Dla porównania skali: w roku obrotowym 2025, zakończonym 25 IV 2025, NetApp miał według komunikatu złożonego w SEC 6,57 mld dolarów przychodu netto, o 5% więcej niż rok wcześniej. To mniej więcej trzy tysiące razy więcej niż w roku 1994. Roczna stopa przychodów z macierzy all-flash wynosiła 4,1 mld dolarów (rekord, wzrost o 14%), segment chmury publicznej — 665 mln, w tym 416 mln z usług, które dostawcy chmury oferują jako własne (first-party), i ze sprzedaży przez ich sklepy (marketplace). Marża brutto według zasad GAAP wynosiła 70,2%. George Kurian tłumaczył wynik „significant market share gains in all-flash storage”. Lodówka na dane okazała się dobrym biznesem.

Hitz przeszedł na emeryturę w lutym 2019 roku jako „Founder Emeritus” — „I don't get paid, but I get to keep my email address and eat at the company cafeteria”, czyli bez pensji, ale z firmowym adresem e-mail i prawem do stołówki. Poza firmą jego fundacja sfinansowała między innymi badania metodą LIDAR w Gwatemali, które odkryły około 60 000 budowli Majów, a sam Hitz przez sześć lat toczył spór prawny o przyjmowanie kobiet do Deep Springs College. Jego wspomnienia noszą tytuł „How to Castrate a Bull”; recenzent The Register zauważył w 2009 roku, że w książce o historii NetApp nie pada ani nazwa Data ONTAP, ani nazwa Spinnaker.

WAFL od środka: artykuł z USENIX 1994

Artykuł, który przetrwał trzy dekady

19 I 1994 roku na zimowej konferencji USENIX w San Francisco Dave Hitz, James Lau i Michael Malcolm wygłosili referat „File System Design for an NFS File Server Appliance” — projekt systemu plików dla urządzenia będącego serwerem NFS. NetApp przedrukował go później jako raport techniczny TR3002 (wersja C z marca 1995 roku), a dziś tekst można przeczytać w materiałach do wykładu o systemach operacyjnych w Princeton, na stronie USENIX i na stronie publikacji NetApp. To najlepsze źródło do zrozumienia, czym ONTAP był na początku, i duża część tego, co opisuje, działa w macierzach NetApp do dziś. Hitz pisał w 2019 roku, że idee z rdzenia WAFL przetrwały drogę od pojedynczego woluminu o pojemności 6 GB do dzisiejszych systemów i chmury.

Artykuł zaczyna od definicji: „An appliance is a device designed to perform a particular function”. Serwer, o którym mowa, ma jedną funkcję — obsługę protokołu NFS — i całe jego oprogramowanie jest podporządkowane tej funkcji. Z tej definicji wynika reszta projektu: system plików ma przede wszystkim dobrze obsługiwać NFS, po awarii zasilania wracać do pracy szybko i bez długiego sprawdzania dysków, a do tego — i to w 1994 roku wyróżniało projekt najbardziej — robić migawki, czyli kopie całego systemu plików z danej chwili, tworzone często i bez kopiowania danych.

Bloki, i-węzły i metadane, które są plikami

WAFL — Write Anywhere File Layout, „układ plików z zapisem gdziekolwiek” — przechowuje dane w blokach po 4 KB, bez dzielenia ich na mniejsze fragmenty. Pliki opisują i-węzły (inode) w stylu uniksowym; każdy i-węzeł WAFL ma 16 wskaźników bloków. Małe pliki mieszczą się w blokach wskazywanych bezpośrednio, większe korzystają z bloków pośrednich, które same zawierają wskaźniki do kolejnych bloków. Na tym poziomie WAFL nie różni się bardzo od innych uniksowych systemów plików swoich czasów.

Różnica zaczyna się przy metadanych. W klasycznym uniksowym systemie plików tablica i-węzłów i mapa wolnych bloków leżą w ustalonych miejscach dysku, wyznaczonych przy formatowaniu. W WAFL wszystkie metadane są zwykłymi plikami: **plik i-węzłów** zawiera i-węzły wszystkich innych plików, **plik mapy bloków** mówi, które bloki są zajęte, a **plik mapy i-węzłów** — które i-węzły. Autorzy wyjaśniają, po co: „Keeping meta-data in files allows WAFL to write meta-data blocks anywhere on disk” — skoro metadane są plikami, ich bloki też można zapisywać w dowolnym miejscu dysku. A to jest warunek konieczny, żeby system plików nigdy niczego nie nadpisywał w miejscu: gdyby choć jedna struktura musiała leżeć pod stałym adresem, każda jej zmiana oznaczałaby nadpisanie.

Cały system plików jest więc drzewem bloków. Na jego szczycie stoi jeden specjalny i-węzeł — **i-węzeł główny** (root inode), który opisuje plik i-węzłów. Plik i-węzłów wskazuje i-węzły wszystkich innych plików, łącznie z plikami metadanych, a te wskazują bloki danych. W takim układzie w znanym miejscu musi leżeć właściwie tylko i-węzeł główny; wszystko inne można znaleźć, schodząc od niego w dół drzewa.

Mapa bloków: trzydzieści dwa bity na blok

Najprostszy pomysł w artykule jest zarazem najbardziej przemyślny. Mapa bloków WAFL nie przechowuje dla każdego bloku jednego bitu „zajęty–wolny”, jak klasyczne mapy wolnego miejsca, tylko 32-bitowy wpis. Bit 0 mówi, czy blok należy do aktywnego systemu plików — tego, który widzą klienci NFS. Bity od 1 wzwyż mówią, czy blok należy do kolejnych migawek. Blok jest wolny dopiero wtedy, gdy żaden bit nie jest ustawiony: ani aktywny system plików, ani żadna migawka go nie używa.

Dzięki temu usunięcie pliku z aktywnego systemu plików nie zwalnia jego bloków, jeśli są w migawce — zeruje się tylko bit 0, a bity migawek zostają. Usunięcie migawki zeruje jej bit we wszystkich wpisach mapy; bloki, w których nie został żaden inny bit, stają się wolne. Nie trzeba przy tym przeglądać plików ani kopiować danych. Żeby zrozumieć, dlaczego to działa, trzeba jednak najpierw zobaczyć, czym w WAFL jest sama migawka.

Migawka jako kopia jednego i-węzła

Skoro cały system plików jest drzewem zakorzenionym w i-węźle głównym, a żaden blok nie jest nigdy nadpisywany w miejscu, to migawkę można zrobić, kopiując **tylko i-węzeł główny**. Kopia wskazuje to samo drzewo co oryginał. Kiedy klient zmienia plik, WAFL nie nadpisuje starego bloku, tylko zapisuje nowy w wolnym miejscu; żeby nowy blok był osiągalny, trzeba zmienić wskaźnik w bloku pośrednim, który go wskazuje — więc ten blok pośredni też zapisuje się od nowa w innym miejscu, i tak dalej w górę, aż do i-węzła głównego aktywnego systemu plików. Kopia i-węzła głównego zrobiona wcześniej dalej wskazuje stare bloki. Aktywny system plików i migawka zaczynają się różnić dokładnie o te bloki, które od chwili migawki się zmieniły.

To jest mechanizm copy-on-write, „kopiowanie przy zapisie”, w jego najczystszej postaci: nowa migawka w chwili utworzenia nie zajmuje prawie nic, a z czasem rośnie o tyle, ile danych zmieniono w aktywnym systemie plików. Częsty mit mówi, że migawka WAFL „kopiuje dane” — artykuł opisuje coś odwrotnego. Kopiowany jest jeden i-węzeł; dane zostają tam, gdzie były, a koszt migawki ponosi się dopiero przy zmianie danych, i to nie przez kopiowanie, tylko przez pozostawienie starych bloków w spokoju.

W 1994 roku WAFL pozwalał trzymać do 20 migawek jednocześnie. Klient NFS widział je w ukrytym katalogu `.snapshot`, obecnym w każdym katalogu systemu plików i czytelnym zwykłymi narzędziami: żeby odzyskać skasowany wczoraj plik, użytkownik po prostu wchodził do `.snapshot` i kopiował plik z odpowiedniej migawki, bez pomocy administratora i bez taśm. Domyślny

harmonogram z artykułu wyglądał tak: cztery migawki godzinowe w ciągu dnia, trzymane przez dwa dni; migawka nocna o północy, trzymana przez tydzień; migawka tygodniowa w niedzielę o północy, trzymana przez dwa tygodnie. Według autorów tydzień migawek zajmował „10 to 20 percent of disk space”. Ten układ — godzinowe, dzienne, tygodniowe — przetrwał w produktach NetApp jako wzorzec domyślnej polityki migawek.

Punkty spójności i NVRAM

Skoro WAFL nigdy nie nadpisuje bloków w miejscu, to stan systemu plików na dysku zmienia się tylko w jednym momencie: kiedy nowy i-węzeł główny zastępuje stary. Do tej chwili wszystkie nowe bloki — dane, bloki pośrednie, fragmenty plików metadanych — leżą na dysku w miejscach, których stary i-węzeł główny nie wskazuje, więc dla systemu plików ich nie ma. Po zapisaniu nowego i-węzła głównego stary stan znika, a nowy jest kompletny. Autorzy nazywają taki moment **punktem spójności** (consistency point). WAFL tworzy go „every few seconds” — co kilka sekund — i nie rzadziej niż co 10 sekund. Klienci NFS nie widzą punktów spójności w ogóle; z ich perspektywy serwer po prostu przyjmuje i potwierdza żądania.

Pomiędzy punktami spójności na dysku jest więc zawsze stan spójny, ale stary — sprzed kilku sekund. Co z żądaniami, które serwer potwierdził klientom w tym czasie? Serwer, który potwierdził zapis, nie może go zgubić przy awarii. Tu wchodzi drugi element projektu: **NVRAM**, pamięć nieulotna. WAFL zapisuje w niej dziennik żądań NFS, które przyszły od ostatniego punktu spójności. Artykuł opisuje go jako podzielony na dwie połowy, dwa dzienniki, i podaje, że jeden megabajt NVRAM mieści ponad tysiąc operacji. Ile NVRAM-u miały pierwsze filery, artykuł nie mówi.

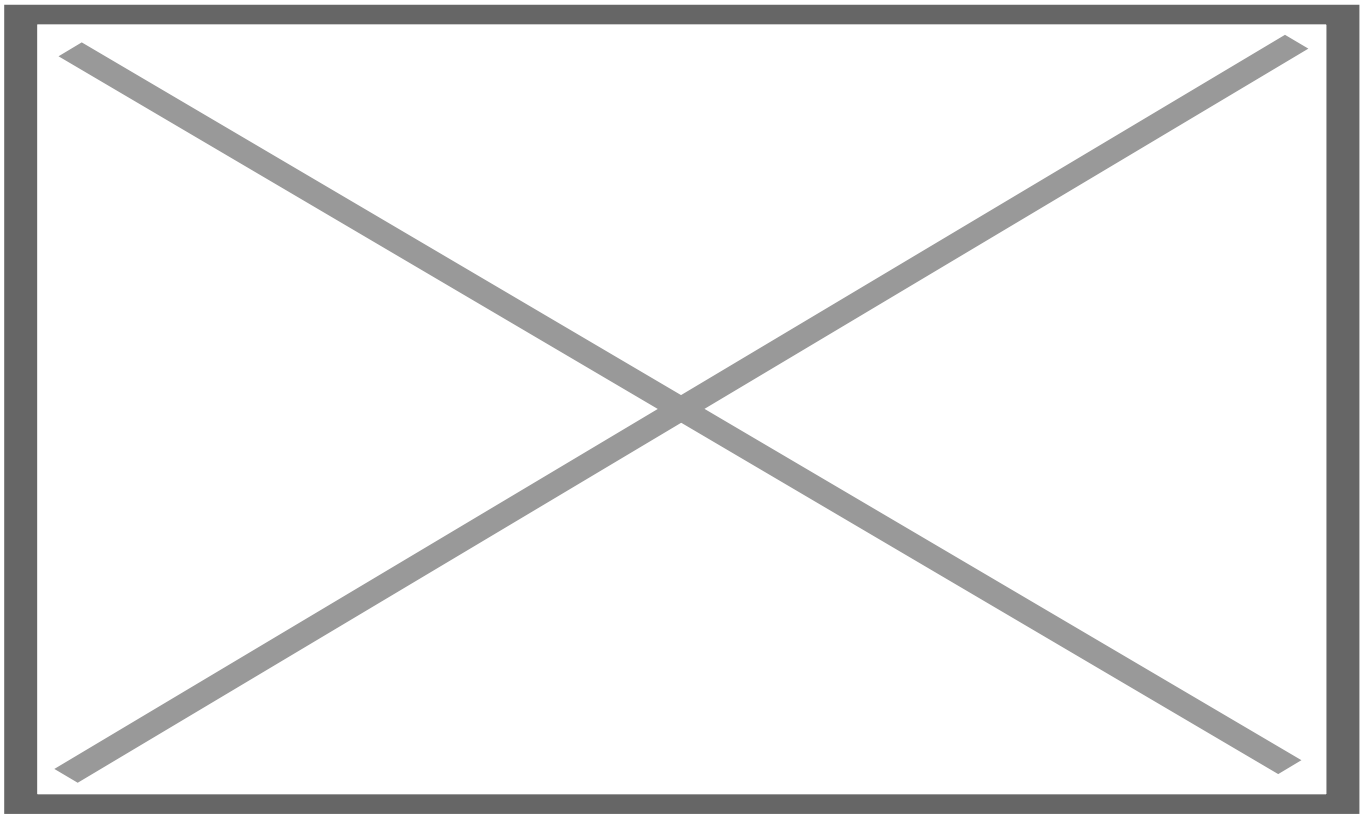
Głowica (kontroler) filera NetApp FAS6030, sfotografowana obiektywem typu rybie oko

Głowica (kontroler) filera NetApp FAS6030, sfotografowana obiektywem typu rybie oko. Źródło: Flickr, CC BY 2.0, autor: ChrisDag (Chris Dagdigian).

W NVRAM nie ma więc bloków dyskowych, tylko dziennik żądań w postaci, w jakiej przyszły od klientów; same bloki — tak wynika z opisu — trafiają na dysk razem, przy najbliższym punkcie spójności.

Restart bez fsck

Z dwóch poprzednich mechanizmów wynika trzeci, który w 1994 roku robił największe wrażenie na administratorach. Po awarii zasilania serwer z klasycznym uniksowym systemem plików musiał uruchomić `fsck` — program, który przegląda wszystkie metadane i naprawia niespójności powstałe, gdy zapis został przerwany w połowie. Przy dużych dyskach trwało to długo. WAFL tego nie potrzebuje: na dysku jest zawsze ostatni punkt spójności, kompletny i spójny, a to, co zdarzyło się później, jest w dzienniku NVRAM. Po awarii serwer wraca do ostatniego punktu spójności i odtwarza żądania z dziennika. Autorzy podają, że filer wstaje w ten sposób „in about a minute even with 20 GB or more of data in its single partition” — w około minutę, nawet z ponad 20 GB danych na jednej partycji. Dwadzieścia gigabajtów było w 1994 roku dużym systemem plików.



Animacja 1. *WAFL w ruchu: od żądania NFS do punktu spójności. Stan na dysku zmienia się tylko w chwili zapisu nowego i-węzła głównego, dlatego po awarii wystarczy ostatni punkt spójności i dziennik NVRAM, a migawka kosztuje dopiero wtedy, gdy zmieniają się dane.*

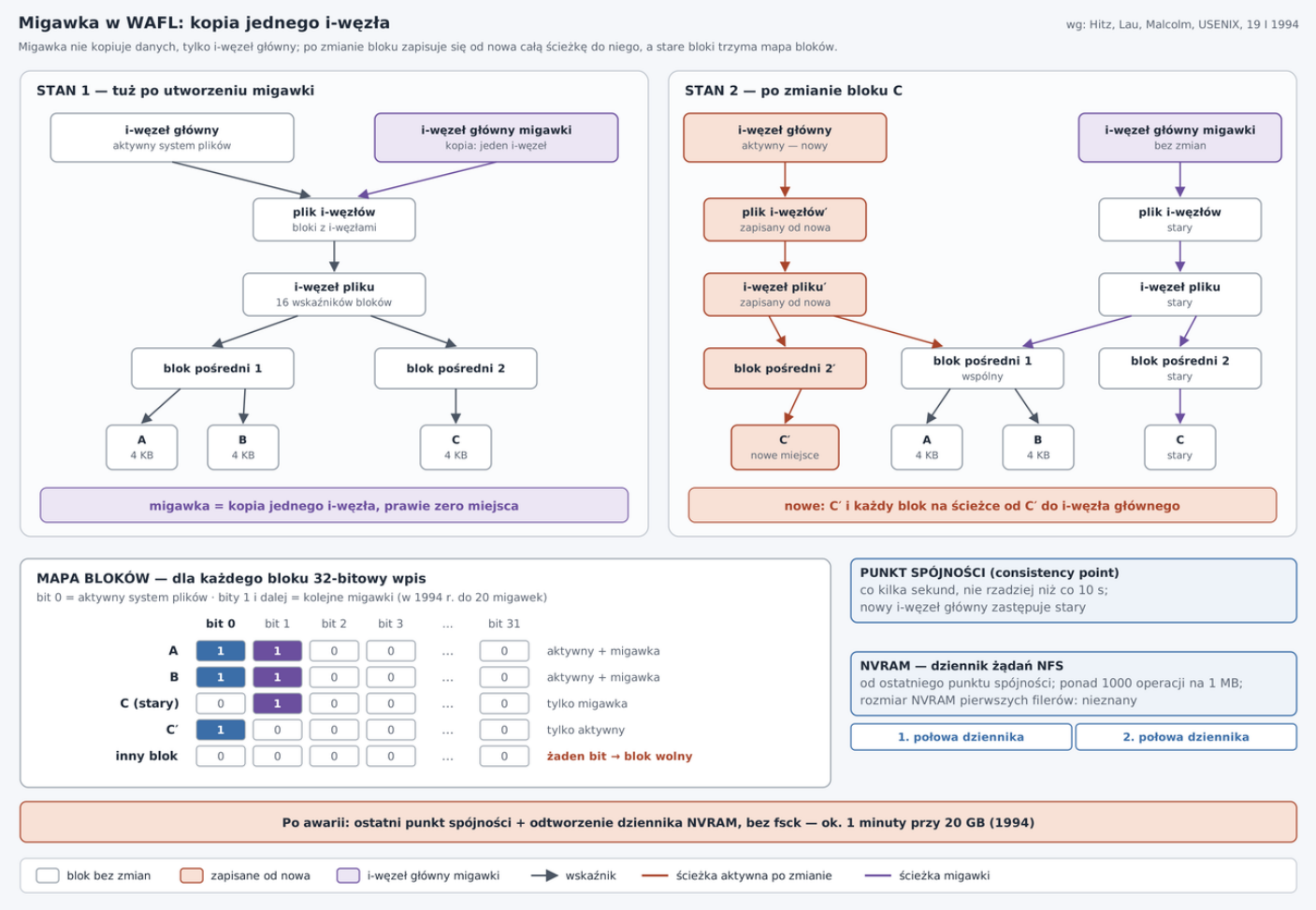
Czym WAFL nie jest: system log-structured

W obiegowych opisach WAFL bywa nazywany systemem plików „log-structured”, czyli takim, w którym cały dysk jest jednym dziennikiem, a dane zapisuje się zawsze na jego końcu. Taki projekt istniał: Log-Structured File System (LFS), opisywany przez Ousterhouta i Douglisa. Artykuł o WAFL cytuje LFS — ale tylko w kontekście obciążeń zdominowanych przez zapisy, jako przykład innego podejścia do tego samego problemu. Sam WAFL nie jest w artykule opisany jako log-structured. Autorzy porównują swoje punkty spójności do dwóch innych mechanizmów: do **shadow paging** (stronicowania cieniowego) z bazy danych System R oraz do **klonów zbiorów plików** (fileset clones) w systemie plików Episode.

Różnica nie jest tylko terminologiczna. WAFL nie dopisuje danych na koniec jednego dziennika, tylko zapisuje nowe bloki „anywhere” — gdziekolwiek jest wolne miejsce, wybierając je tak, jak mu wygodnie — i odzyskuje miejsce przez mapę bloków: blok jest wolny, gdy nie należy ani do aktywnego systemu plików, ani do żadnej migawki. Oba pomysły łączy to, że nie nadpisują danych w miejscu; różni wszystko inne.

Osiem do jednego — z zastrzeżeniem

Artykuł kończy się porównaniem wydajności według testu SPEC SFS (nazywanego też LADDIS) z wynikami w operacjach SPECnfs_A93 na sekundę. Na wykresie stoją obok siebie konfiguracja „FAServer 8X Cluster” Network Appliance, systemy Auspex i serwery Sun SPARCenter. Autorzy formułują wniosek ostro — na jeden system plików WAFL „outperforms the other file systems by almost eight to one”, czyli jest prawie ośmiokrotnie szybszy — ale od razu dodają, że takie porównania mogą wprowadzać w błąd.



Rysunek 1. *Migawka w WAFL nie kopiuje danych — kopiuje jeden i-węzeł. Warto zauważyć, że po zmianie bloku C nowe są także bloki pośrednie na ścieżce do niego, a mapa bloków z 32-bitowym wpisem pozwala zwolnić blok dopiero wtedy, gdy nie należy do żadnej migawki.*

RAID 4, RAID-DP i RAID-TEC

RAID 4 i zapis gdziekolwiek

Macierze NetApp chroniły dane parzystością w układzie RAID 4 — tak opisuje punkt wyjścia praca o RAID-DP z 2004 roku. Dane rozkłada się na kilka dysków, a parzystość — sumę XOR bloków z tego samego „wiersza” — trzyma na jednym osobnym dysku. W podręcznikowym ujęciu RAID 4 ma znaną wadę: każdy zapis zmienia parzystość, więc przy wielu drobnych, rozproszonych zapisach dysk parzystości staje się wąskim gardłem.

Macierz NetApp z półkami dysków; według autora zdjęcia „każda szara szczelina to dysk”

Macierz NetApp z półkami dysków; według autora zdjęcia „każda szara szczelina to dysk”. Źródło: Flickr, CC BY 2.0, autor: zombieite.

WAFL zmienia ten rachunek, bo nie zapisuje drobnych zmian w miejscu. Bloki zmienione od ostatniego punktu spójności trafiają na dysk razem, w miejsca, które system plików sam wybiera. Autorzy pracy o RAID-DP z 2004 roku ujmują to jednym zdaniem: „The WAFL file system uniformly distributes writes across all the data disks” — WAFL rozkłada zapisy równomiernie na wszystkie dyski danych. Przy takim sposobie zapisu oddzielny dysk parzystości nie musi być wąskim gardłem. Tak przynajmniej rozumiem zestawienie tych dwóch faktów; sama praca z 2004 roku nie rozpisuje tego wprost dla pojedynczej parzystości.

Row-Diagonal Parity: dwie awarie naraz

Na początku XXI wieku pojedyncza parzystość przestała wystarczać. Wyjaśnia to praca, którą siedmioro inżynierów Network Appliance — Corbett, English, Goel, Grcanac, Kleiman, Leong i Sankar — przedstawiło na konferencji FAST '04 (3rd USENIX Conference on File and Storage Technologies, San Francisco, 31 III – 2 IV 2004): „Row-Diagonal Parity for Double Disk Failure Correction”. Argument jest statystyczny. Dyski rosły szybciej niż ich przepustowość, więc odbudowa po awarii trwała coraz dłużej; tańsze dyski ATA, coraz częstsze w macierzach, były mniej niezawodne; a producenci dysków podawali, że mniej więcej jeden nienaprawialny błąd odczytu przypada na 10 do 100 TB przeczytanych danych. Autorzy piszą: „a single whole-disk failure with one or more media failures is becoming particularly troublesome” — kłopotem staje się nie tyle awaria dwóch dysków naraz, ile awaria jednego dysku połączona z błędem nośnika na którymś z pozostałych w czasie odbudowy.

Praca pokazuje to na przykładzie. Macierz z dziesięciu dysków po 240 GB, awaryjność dysku jeden na 500 000 godzin, odbudowa z prędkością 100 MB/s. Prawdopodobieństwo awarii drugiego dysku w czasie odbudowy wychodzi około $1{,}2\times10^{-9}$ na godzinę, a prawdopodobieństwo awarii dysku połączonej z błędem nośnika — około $3{,}2\times10^{-6}$ na godzinę, czyli ponad dwa i pół tysiąca razy więcej. Przy pojedynczej parzystości średni czas do utraty danych (MTTDL) wynosi w tym przykładzie około $3{,}1\times10^{5}$ godzin, co autorzy przeliczają na około 0,028 utraty danych na macierz na rok. Innymi słowy: na stu takich macierzach można się spodziewać mniej więcej trzech utrat danych rocznie, i prawie zawsze z powodu błędu nośnika w czasie odbudowy, a nie drugiej awarii całego dysku.

Odpowiedzią był algorytm **Row-Diagonal Parity** (RDP): „a new algorithm for protecting against double disk failures”. Działa wyłącznie na operacjach XOR, a dane nie są w nim w żaden sposób

kodowane — leżą na dyskach danych tak jak w zwykłym RAID 4. Do parzystości wierszowej dochodzi drugi dysk z parzystością przekątną: każdy blok danych należy do dokładnie jednego zestawu parzystości wierszowej i dokładnie jednej przekątnej. Parametrem jest liczba pierwsza p większa od 2; najprostsza macierz ma $p+1$ dysków — $p-1$ dysków danych, jeden dysk parzystości wierszowej i jeden przekątnej. Szczegół, który wygląda na przeoczenie, a jest częścią konstrukcji: parzystość jednej z przekątnych celowo się pomija. Mechanizmu odtwarzania krok po kroku nie streszczam — praca dowodzi, że taki układ wystarcza do odzyskania danych po utracie dowolnych dwóch dysków.

Układ jest w duchu RAID 4: obie parzystości leżą na osobnych dyskach. Według autorów wydajność RDP mieści się „within 6 percent of RAID-4” — w granicach 6% wydajności pojedynczej parzystości. Algorytm wdrożono jako nową funkcję „Data ONTAP version 6.5”; w produktach NetApp nazywa się **RAID-DP** (double parity). Dokładnej daty wydania Data ONTAP 6.5 w zebranych materiałach nie mam; praca z kwietnia 2004 roku opisuje go jako wydanie, w którym algorytm już jest.

RAID-TEC: trzecia parzystość

Dwanaście lat później dyski były jeszcze większe, a odbudowy jeszcze dłuższe. W ONTAP 9, zapowiedzianym w maju 2016 roku, NetApp dodał **RAID-TEC** — ochronę potrójną parzystością. O tej nowości wiem z bloga partnera NetApp, a nie z dokumentacji producenta, więc szczegółów jej budowy nie podaję. Sama sekwencja jest jednak czytelna: RAID 4, RAID-DP w 2004 roku, RAID-TEC w 2016. Przy RAID-DP NetApp uzasadniał krok wprost tym, że dyski rosły szybciej niż prędkość, z jaką da się je przeczytać w całości; uzasadnienia RAID-TEC z pierwszej ręki nie mam.

Administrator, który zna ZFS, zauważy analogię: ZFS też ma grupy RAIDZ z pojedynczą, podwójną i potrójną parzystością, a od OpenZFS 2.3 (2025) do istniejącej grupy RAIDZ każdego z tych poziomów można dokładać po jednym dysku. W obu światach odpowiedzią na coraz większe dyski była kolejna parzystość.

Od filera do klastra: Spinnaker i jedenaście lat dwóch trybów

Data ONTAP 7G: wolumin oderwany od dysków

Do połowy pierwszej dekady XXI wieku wolumin na macierzy NetApp był przywiązany do konkretnych dysków: tworząc go, administrator decydował, z których dysków się składa, a zmiana tej decyzji była pracochłonna. W listopadzie 2004 roku NetApp zapowiedział **Data ONTAP 7G**,

który to zmienił. Dyski łączy się w **agregaty**, a na agregatach tworzy się woluminy **FlexVol**, oderwane od fizycznych dysków — tworzone i zmniejszane jednym poleceniem i dołączone do 7G bez dodatkowej opłaty. Obok nich pojawił się **FlexClone**: zapisywalne klony woluminów FlexVol, tworzone bez kopiowania danych, z licencją sprzedawaną osobno, od około 6000 dolarów. NetApp reklamował, że można zrobić „up to 200 clones … in roughly the time of one”, czyli dwieście klonów w czasie potrzebnym na jeden. Trzecią nowością był gFiler — urządzenie, które wirtualizowało cudze macierze Fibre Channel: HDS, HP XP, IBM DS4000, Sun StorEdge 9900 i IBM ESS.

Filer NetApp 6030

Filer NetApp 6030. Źródło: Flickr, CC BY 2.0, autor: ChrisDag (Chris Dagdigian).

Suresh Vasudevan, wiceprezes NetApp od marketingu produktu, tłumaczył wtedy w Enterprise Storage Forum: „The current approach to provisioning storage requires several hours to create a volume” — przygotowanie woluminu wymagało kilku godzin — i dodawał: „No mainstream storage vendor has anything close to FlexVol”. To drugie zdanie jest oczywiście twierdzeniem marketingowym, ale pierwsze dobrze opisuje, z czym walczył administrator macierzy w 2004 roku. FlexClone przypomina zresztą migawkę z artykułu z 1994 roku, do której można pisać — też bez kopiowania danych. Patent NetApp na „writable snapshots”, zapisywalne migawki, wróci w tej opowieści przy sporze z Sunem.

Spinnaker: 300 milionów w akcjach za klaster

Rok wcześniej NetApp zrobił zakup, który miał zaważyć na całej następnej dekadzie. 4 XI 2003 roku Network Appliance ogłosiło umowę przejęcia **Spinnaker Networks** — prywatnej firmy z Pittsburgha, założonej w 1999 roku, sprzedającej serwery SpinServer. Cena: około 300 mln dolarów w akcjach. Umowę fuzji podpisano 3 XI 2003 roku, aneks — 9 II 2004, a przejęcie zamknięto 18 II 2004 roku. Raport złożony w SEC opisywał Spinnakera jako „a pioneer in scalable system architectures, distributed file systems, next-generation clustering technologies, and virtualization” — pioniera skalowalnych architektur, rozproszonych systemów plików i klastrów.

W komunikacie o przejęciu Warmenhoven mówił ogólnikowo: „Today's business demands fast, reliable, secure access to data”. Konkretniejszy był Bianchini, prezes, dyrektor generalny i współzałożyciel Spinnakera, który obiecywał „flexible data infrastructure that offers high performance, massive and simple scalability”. Chodziło o to, czego filer NetApp nie umiał: o system plików rozłożony na wiele kontrolerów, z jedną przestrzenią nazw, który można powiększać, dokładając kolejne węzły.

W tym samym 2004 roku, według Fundacji FreeBSD, kierownictwo NetApp wybrało FreeBSD jako podstawę ONTAP. Zbieżność dat jest wyraźna, ale źródło nie łączy tych decyzji przyczynowo i ja też tego nie robię; o FreeBSD w ONTAP jest osobny rozdział niżej.

Data ONTAP GX i „Gwiezdne wojny”

Na pierwszy produkt z technologii Spinnakera trzeba było czekać ponad dwa lata. 12 VI 2006 roku NetApp ogłosił **Data ONTAP GX** — wersję systemu z globalną przestrzenią nazw wywodzącą się ze Spinnakera, działającą na macierzach FAS6070 i FAS3050 i certyfikowaną na klastrze z 24 węzłów i 6 PB danych. Adresatem był rynek obliczeń wielkiej skali (HPC): firmy naftowe, bioinformatyka, projektanci układów scalonych, studia efektów filmowych. Computerworld podawał cenę od 212 000 dolarów za dwa węzły FAS3050 z 7 TB pojemności. The Register pisał o „high-performance Linux NAS arrays” — chodziło, jak rozumiem, o macierze dla klientów linuksowych, typowych w HPC, a nie o macierze działające na Linuksie.

Najlepszą reklamą GX było studio efektów specjalnych ILM (Industrial Light & Magic), które według Computerworld używało prototypu przez ponad dwa lata. System renderujący ILM składał się z 3000 procesorów AMD Opteron 64-bit z systemem SUSE Linux, około stu macierzy FAS270C i dwudziestu SpinServerów, które razem tworzyły jeden wirtualny dysk o pojemności 200 TB. Przy otwierającej bitwie z „Gwiezdnych wojen: Części III” studio mogło według tej relacji pracować na ujęciach z 2500 klatek — ponad sto sekund filmu — zamiast około trzydziestu sekund wcześniej. Michael Thompson z ILM mówił: „Without this system, we would have had 10 30TB [volumes]” — bez tego mielibyśmy dziesięć osobnych woluminów po 30 TB — i „It lets end users look at storage like a utility, like electricity or water”, czyli jak na prąd albo wodę z sieci.

Data ONTAP 8: dwa systemy pod jedną nazwą

GX był osobną linią kodu, z osobnymi funkcjami. Klienci, którzy chcieli klastra, musieli zrezygnować z części tego, co miał klasyczny ONTAP 7G, i odwrotnie. Połączyć je miał **Data ONTAP 8**, zapowiedziany we wrześniu 2009 roku: wersja RC wyszła 3 IX 2009, wersja ogólnie dostępna (GA) — 19 III 2010, a 8.0.1 RC1 — 3 IX 2010. Skip Shapiro z NetApp pisał w biuletynie Tech OnTap, że ósemka „merges the capabilities of two previously separate branches of Data ONTAP” — łączy możliwości dwóch dotąd osobnych gałęzi. W praktyce jedna nazwa kryła dwa tryby. **7-Mode** to były funkcje 7.3.x z tym samym interfejsem; **Cluster-Mode** — według Shapiro „supports multicontroller configurations with a global namespace and clustered file system”, czyli następca GX. Agregaty 32-bitowe miały limit 16 TB, nowe 64-bitowe — „as large as 30TB to 100TB”, zależnie od modelu. I zapowiedź na przyszłość: „All future Data ONTAP feature development will be on Data ONTAP 8.”

Macierz NetApp FAS6030, opisana przez autora zdjęcia jako klaster

Macierz NetApp FAS6030, opisana przez autora zdjęcia jako klaster. Źródło: Flickr, CC BY 2.0, autor: ChrisDag (Chris Dagdigian).

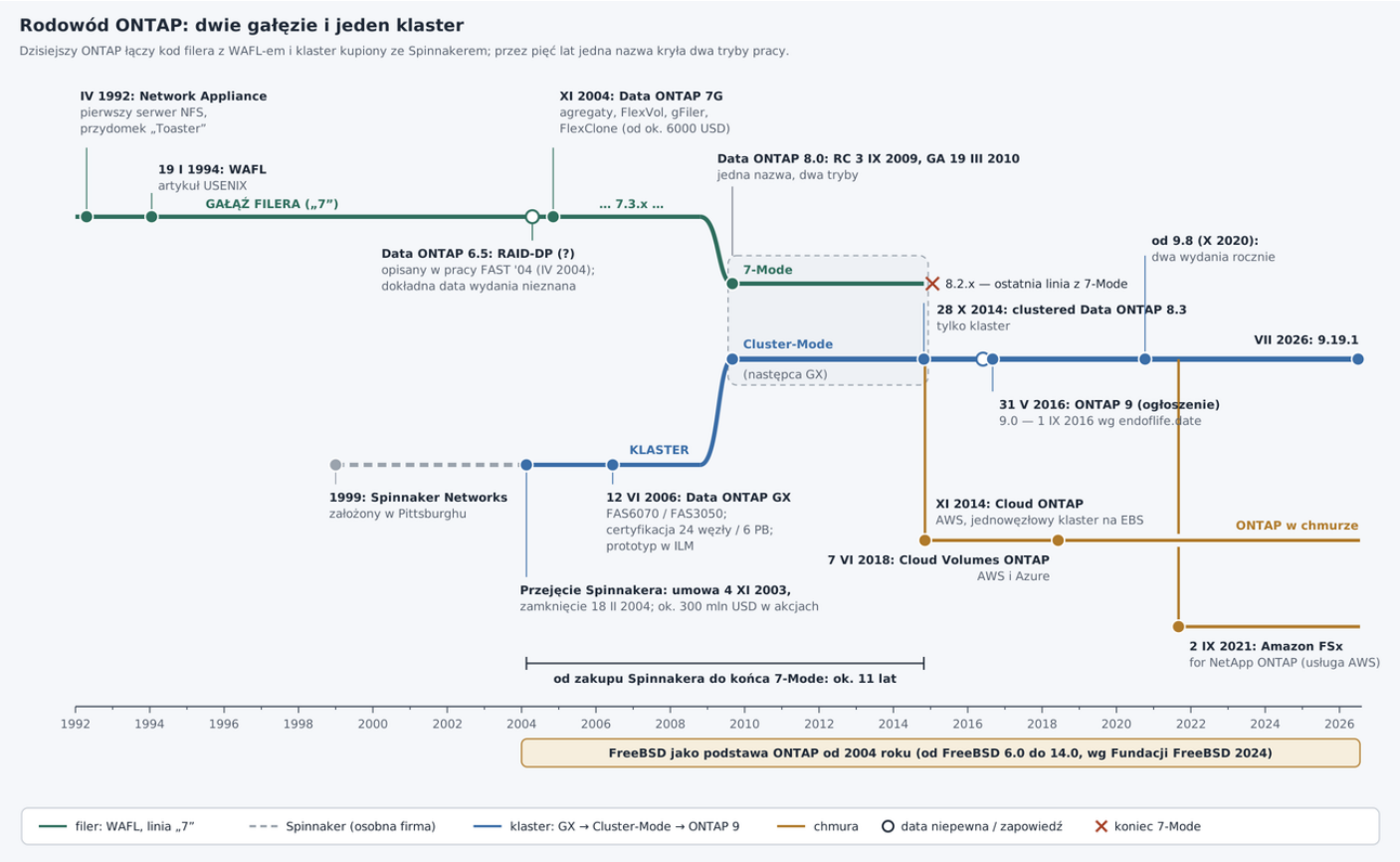
Na forum NetApp z tamtego czasu widać, jak to wyglądało od strony klienta. Jeden z uczestników dyskusji tłumaczył, że tryby 7 i Cluster to wybór „albo–albo”: w 8.0 nie dało się połączyć klastra wywodzącego się z GX z funkcjami 7G, na przykład z deduplikacją A-SIS. W tej samej dyskusji pada zdanie: „Cluster-Mode is the successor to Data ONTAP GX”. Dwa systemy pod jedną nazwą miały wspólną numerację, ale administrator musiał wybrać, którym z nich jest jego macierz.

8.3: koniec 7-Mode

Rozstrzygnięcie przyszło 28 X 2014 roku na konferencji NetApp Insight. Clustered Data ONTAP 8.3 był według Tech OnTap „the first Data ONTAP release to support clustered operation only” — pierwszym wydaniem obsługującym wyłącznie tryb klastrowy — i „does not support 7-Mode”. Klienci z 7-Mode mieli zostać na linii 8.2.x. NetApp podawał przy tej okazji, że w trybie klastrowym pracuje już ponad 1,9 EB danych i prawie 24 000 kontrolerów, a ponad 100 000 systemów korzysta z 64-bitowych agregatów — to liczby producenta z listopada 2014 roku. Od zakupu Spinnakera do porzucenia 7-Mode minęło więc około jedenastu lat.

ONTAP 9: bez „Data”

31 V 2016 roku NetApp ogłosił kolejną wersję i przy okazji skrócił nazwę: „Today, NetApp announced the new version on Data ONTAP, now called ONTAP 9” — tak relacjonował to blog jednego z partnerów, z którego mam tę informację. Pierwsza wersja kandydująca (RC1) miała się pojawić pod koniec czerwca 2016 roku; serwis endoflife.date podaje jako datę wydania 9.0 dzień 1 IX 2016[2]. Wśród nowości były kompaktowanie danych (data compaction), funkcja HeadRoom i wspomniany już RAID-TEC. Od tej pory numeracja szła 9.0, 9.1, 9.2 i dalej, a późniejsze wydania noszą numery z końcówką .1 — w zebranych tabelach od 9.9.1 do 9.19.1 z lipca 2026 roku. O cyklu wydań i wsparciu jest osobny rozdział w części praktycznej.



Rysunek 2. *Dzisiejszy ONTAP jest potomkiem dwóch linii kodu: filera z WAFL-em i klastra kupionego razem ze Spinnakerem. Warto zauważyć, jak długo trwało ich łączenie — przez pięć lat (2009–2014) jedna nazwa kryła dwa różne tryby pracy.*

Co jest pod spodem: FreeBSD

Decyzja z 2004 roku

Pytanie „na czym działa ONTAP” ma w sieci kilka sprzecznych odpowiedzi: że na własnym jądrze, że na Linuksie, że na FreeBSD, ale tylko w niektórych liniach. Najpewniejsze źródła, jakie znalazłem, to dwa: Fundacja FreeBSD i biuletyny bezpieczeństwa samego NetApp. Oba mówią to samo.

Fundacja FreeBSD opublikowała 18 III 2024 roku studium przypadku „FreeBSD provides Flexibility and Performance for NetApp's ONTAP Data Management System”. Według niego w 2004 roku kierownictwo NetApp wybrało FreeBSD jako podstawę ONTAP — „the underlying data management software across their entire product range”, czyli oprogramowania zarządzania danymi w całej gamie produktów. Powodem była licencja BSD, która pozwala dołożyć do systemu własnościową technologię bez obowiązku publikowania jej kodu. Rozważano inne systemy z rodziny BSD; FreeBSD wybrano za równowagę wydajności i bezpieczeństwa. Od tego czasu FreeBSD jest w produktach NetApp „from version 6.0 in 2004 to version 14.0 today” — od wersji 6.0 w 2004 roku

do 14.0 w chwili publikacji.

Studium opisuje też, jak zmieniła się ta współpraca. Dawniej NetApp pracował na starszym wydaniu FreeBSD; dziś „tracking head” — śledzi bieżącą gałąź rozwojową. Inżynierowie NetApp oddali projektowi FreeBSD prawie 300 zmian, codziennie testują kod FreeBSD we własnym systemie ciągłej integracji, a wsparcie komercyjne i pomoc w oddawaniu zmian zapewnia im firma Klara Systems. Z NetApp pochodzi też — według tego samego studium — hipernadzorca **bhyve**: „NetApp engineers upstreamed bhyve”. NetApp sponsoruje Fundację, a jego ludzie bywają na konferencji BSDCan i na spotkaniach producentów (Vendor Summit). Więcej o bhyve i o FreeBSD w ogóle mówi strona **FreeBSD**.

Druga wersja studium, z 1 XI 2024 roku, powtarza te fakty i dodaje cytat. Ranjith Balachandran, inżynier oprogramowania w NetApp: „FreeBSD gives us a variety of implementation options. It allows us to define our own strategy.” — FreeBSD daje nam wiele możliwości wdrożenia i pozwala ustalić własną strategię. Rady NetApp dla innych firm budujących produkty na FreeBSD brzmią jak podręcznik dobrych praktyk: aktualizować regularnie, trzymać się blisko najnowszego wydania, prywatne modyfikacje ograniczać do kluczowej własności intelektualnej, a resztę oddawać projektowi.

Co mówią biuletyny bezpieczeństwa

Fundacja FreeBSD jest stroną zainteresowaną — studium przypadku jest też reklamą systemu. Dlatego cenne jest drugie, niezależne potwierdzenie: biuletyny NetApp Product Security. W biuletynach dotyczących luk we FreeBSD NetApp pisze wprost: „Multiple NetApp products incorporate FreeBSD” — wiele produktów NetApp zawiera FreeBSD. Biuletyn NTAP-20200413-0001 z 13 IV 2020 roku (aktualizowany 15 I 2021) wymienia jako dotknięte luką FreeBSD dwa produkty: „Clustered Data ONTAP” oraz „Data ONTAP operating in 7-Mode”. W innych biuletynach status bywa odwrotny: przy CVE-2021-29628 Clustered Data ONTAP jest oznaczony jako niedotknięty. Tak, jak rozumiem, wygląda produkt zbudowany na cudzym systemie: każdą lukę w nim trzeba ocenić osobno, bo nie każdy fragment FreeBSD jest w ONTAP używany.

Obecność 7-Mode w biuletynie z 2020 roku jest ciekawa. Co dokładnie z FreeBSD było w 7-Mode — biuletyn nie precyzuje, a ja nie ustaliłem. Wiem tyle, ile mówi Fundacja: decyzja z 2004 roku dotyczyła całej gamy produktów.

Co jest FreeBSD, a co NetApp

Z tych źródeł wynika obraz, który warto zapamiętać. Współczesny ONTAP 9 działa na FreeBSD jako podstawie — w 2024 roku na FreeBSD 14. Tym, co sprawia, że ONTAP jest ONTAP-em — WAFL, RAID-DP i RAID-TEC, migawki, SnapMirror, SnapLock, ochrona przed ransomware, klaster z jedną przestrzenią nazw — jest kod własny NetApp, dołożony do FreeBSD właśnie dlatego, że licencja BSD na to pozwala. Administrator macierzy pracuje w wierszu poleceń ONTAP, z jego własnymi poleceniami, takimi jak `volume`, `cluster image` czy `security anti-ransomware`, opisanymi niżej. Jak

wyglądało jądro ONTAP przed 2004 rokiem i co wziął z wcześniejszego kodu BSD, opisuje strona **BSD w praktyce**; w materiałach, z których powstała ta strona, tej części nie sprawdzałem.

ONTAP w chmurze

Gdy system przestaje być przywiązany do sprzętu, można go uruchomić gdziekolwiek. W listopadzie 2014 roku NetApp opisał w Tech OnTap **Cloud ONTAP** dla Amazon Web Services: „Cloud ONTAP is a virtual storage appliance, or VSA, that runs as an Amazon Elastic Compute Cloud (EC2) instance” — wirtualne urządzenie pamięci masowej działające jako instancja EC2. Każda instancja była jednowęzłowym klastrem clustered Data ONTAP, a jej „dyskami” były woluminy EBS typu GP2. Płaciło się na dwa sposoby: godzinowo, przez AWS Marketplace, albo przez sześciomiesięczną subskrypcję kupowaną od NetApp, z własną licencją (BYOL). 7 VI 2018 roku produkt przemianowano na **Cloud Volumes ONTAP** (CVO), dostępny w AWS i Azure; wpis ogłaszający zmianę nazwy nie wymienia Google Cloud.

Trzy lata później ONTAP zrobił krok, którego trudno było się spodziewać po firmie sprzedającej sprzęt. 2 IX 2021 roku Amazon ogłosił ogólną dostępność **Amazon FSx for NetApp ONTAP** — „complete, fully managed ONTAP file systems in the cloud”, czyli pełnych, w całości zarządzanych systemów plików ONTAP w chmurze. Usługa udostępnia NFS, SMB i iSCSI, ma migawki, SnapMirror i FlexClone, replikuje dane między strefami dostępności, a zarządza się nią przez konsolę AWS, wiersz poleceń, API usługi FSx, NetApp Cloud Manager albo bezpośrednio przez REST API i wiersz poleceń ONTAP. Na starcie była dostępna między innymi we Frankfurcie, Irlandii, Londynie, Mediolanie, Paryżu i Sztokholmie. Wayne Duso z AWS nazwał ją „the only complete cloud service that is built on ONTAP” — to zdanie marketingowe — a Anthony Lye z NetApp podkreślał, że usługę obsługuje, uruchamia i sprzedaje AWS. FSx for ONTAP obsługuje też woluminy SnapLock (WORM); od kiedy, nie ustaliłem.

Dla administratora różnica jest istotna: CVO to ONTAP uruchamiany jako maszyna wirtualna w chmurze AWS albo Azure, FSx — usługa, którą obsługuje, uruchamia i sprzedaje Amazon. W obu działa system, który w 1992 roku miał być „najprostszym możliwym NFS-em”.

ZFS i spór NetApp z Sunem

„Ostatnie słowo w systemach plików”

We wrześniu 2004 roku Sun zapowiedział nowy system plików dla Solarisa 10. Slashdot opisał to 16 IX 2004 pod tytułem „ZFS: the last word in file systems” i odesłał do artykułu na stronach Suna; adres tamtego tekstu już nie działa, a jego kopii nie sprawdzałem. Cytaty z materiałów Suna przytoczone w dyskusji brzmiały jak z broszury: „As the world's first 128-bit file system, ZFS offers 16 billion billion times the capacity of 32- or 64-bit systems” oraz zapowiedź, że ZFS „will make you

forget everything you thought you knew about file systems”. Przypisywane Jeffowi Bonwickowi zdanie „You couldn't fill a 128-bit storage pool without boiling the oceans” — nie da się zapełnić 128-bitowej puli bez zagotowania oceanów — znam tylko z drugiej ręki, z komentarza na Slashdocie. W tej samej dyskusji jeden z czytelników zwrócił uwagę, że mechanizm copy-on-write w ZFS przypomina WAFL NetAppa.

Kod ZFS trafił do głównej gałęzi Solarisa w październiku 2005 roku. Nazwa pierwotnie rozwijała się jako „Zettabyte File System”, ale rozwinięcia później zaniechano. Historię samego ZFS-a — kto go pisał i jak po przejęciu Suna przez Oracle'a powstał OpenZFS — opowiada strona **Solaris i SunOS**. Dla tej strony ważne jest jedno: ZFS robił to, co WAFL robił od 1994 roku — nie nadpisywał danych w miejscu, robił tanie migawki, sam zarządzał redundancją dysków — i był rozpowszechniany jako otwarte oprogramowanie.

Sun Fire X4500 „Thumper”, serwer pamięci Suna. Autor zdjęcia, pracownik Suna, podawał w opisie 2

Sun Fire X4500 „Thumper”, serwer pamięci Suna. Autor zdjęcia, pracownik Suna, podawał w opisie 24 TB pojemności i cenę 2 USD za gigabajt. Źródło: Flickr, CC BY-SA 2.0, autor: Richard Masoner / Cyclelicious.

NetApp nie był pierwszym, który zauważył, że patenty WAFL mogą dotyczyć innych systemów plików. Już w 2000 roku Daniel Phillips, autor linuksowego systemu plików tux2, pisał o nich w tekście, który LWN opatrzył tytułem „Tux2 — evil patents sighted”. Wśród patentów WAFL wymienianych w późniejszych dyskusjach były: 5,819,292 („copy on write”), 5,963,962 („Write anywhere file-system layout”), 6,038,570 (przydział plików w systemie plików zintegrowanym z RAID), 6,857,001 („writable snapshots”, zapisywalne migawki) i 7,174,352 (migawki).

Pozew we Wschodnim Teksasie

5 IX 2007 roku The Register poinformował, że NetApp pozwał Suna przed sądem federalnym w Teksasie. Sun mówił o „East Texas”, Wschodnim Teksasie; według Computerworld, cytowanego w serwisie endsoftwarepatents, był to sąd w Lufkin. Pozew miał dwie części. W pierwszej NetApp żądał stwierdzenia, że nie narusza patentów Suna. W drugiej zarzucał, że ZFS narusza siedem patentów WAFL — dotyczących migawek, korekcji błędów, programowego RAID i przesyłania obrazów systemów plików — i żądał wstrzymania sprzedaży i rozwoju ZFS oraz odszkodowania.

Hitz wyjaśniał na swoim blogu w sposób, który do dziś jest najczęściej cytowanym fragmentem sporu: „We filed suit against Sun because after we pointed out the WAFL patents, their lawyers stopped getting back to us” — pozwaliśmy Suna, bo kiedy wskazaliśmy patenty WAFL, ich prawnicy przestali się odzywać. Według jego relacji to Sun zaczął: mniej więcej półtora roku wcześniej prawnicy Suna wskazali NetAppowi patenty Suna i zaproponowali licencję, za którą trzeba by „pay them lots of money”. Hitz zapewniał też, że pozew „isn't about downloads for personal or non-commercial use; it is about what Sun is doing” — nie chodzi o pobieranie ZFS-a do użytku prywatnego i niekomercyjnego, tylko o to, co robi Sun — i że NetApp nie wierzy, by mógł „erase every copy of ZFS that has been downloaded”, wymazać każdą pobraną kopię.

Sun przedstawiał to inaczej. Według jego wersji to NetApp najpierw chciał kupić patenty Suna, pozew był atakiem na otwarte oprogramowanie, a wybór Wschodniego Teksasu — sądu, który Sun nazywał ulubionym przez trolle patentowe — mówił sam za siebie. Sun zapowiedział, że zabezpieczy (zrekompensuje ewentualne roszczenia) swoich klientów korzystających z ZFS-a.

Kontrpozew, USPTO i dwa sądy

W październiku 2007 roku Sun odpowiedział kontrpozwem w Kalifornii, a w kwietniu 2008 roku dorzucił drugi, o produkty Onaro SANscreen i NAS Insight. Sprawy połączono w sądzie federalnym Północnego Dystryktu Kalifornii: Network Appliance Inc. v. Sun Microsystems Inc., sygnatura C-07-06053 EDL, prowadzona przez sędzię (magistrate judge) Elizabeth D. LaPorte. Liczba patentów Suna w sporze różni się w zależności od źródła[3].

Sun Fire X4540 „Thor", następca Thumpera i serwer pamięci z ZFS-em — ujęcie z albumu autora „Tl

Sun Fire X4540 „Thor", następca Thumpera i serwer pamięci z ZFS-em — ujęcie z albumu autora „Thor/Thumper". Źródło: Flickr, CC BY-NC 2.0, autor: Edmund White.

Równolegle patenty NetApp trafiły do ponownego badania (reexamination) w amerykańskim urzędzie patentowym USPTO. Postanowienia sądu pokazują, jak to wpływało na proces. 23 V 2008 roku sąd zawiesił sprawę w zakresie patentu 6,857,001, ale odmówił zawieszenia co do 5,819,292 i 6,892,211. 28 VIII 2008 roku USPTO wydał pierwszą decyzję w ponownym badaniu patentu '211, wskazując 12 publikacji ze stanu techniki i 16 podstaw odrzucenia. 10 IX 2008 roku odbyła się wykładnia zastrzeżeń patentowych (claim construction). 3 XI 2008 roku sąd ponownie odmówił zawieszenia co do '211, szacując, że ponowne badanie potrwa 31–33 miesiące, i wyznaczył terminy wniosków o rozstrzygnięcie bez rozprawy (summary judgment): 18 VI 2009 dla patentów NetApp i 25 VI 2009 dla patentów Suna.

7 X 2008 roku Chris Mellor w The Register opisał wersję Suna, przedstawioną przez jego radcę generalnego Mike'a Dillona. Według niej USPTO odrzucił wszystkie 63 zastrzeżenia patentu 6,857,001, wszystkie 24 zastrzeżenia 6,892,211 i wszystkie powołane w sporze zastrzeżenia 5,819,292 — trzech „rdzeniowych" patentów WAFL, których współwynalazcą był Hitz. Sąd uznał ponadto zastrzeżenia patentu 7,200,715, dotyczącego RAID, za nieokreślone („indefinite"). Sun skomentował: „Two down and four to go" — dwa z głowy, zostały cztery. NetApp odpowiedział, że ma „policy not to comment".

Decyzje USPTO z 2008 roku były jednak wstępne (non-final office actions), a nie prawomocnym unieważnieniem — sam sąd szacował, że ponowne badanie potrwa prawie trzy lata.

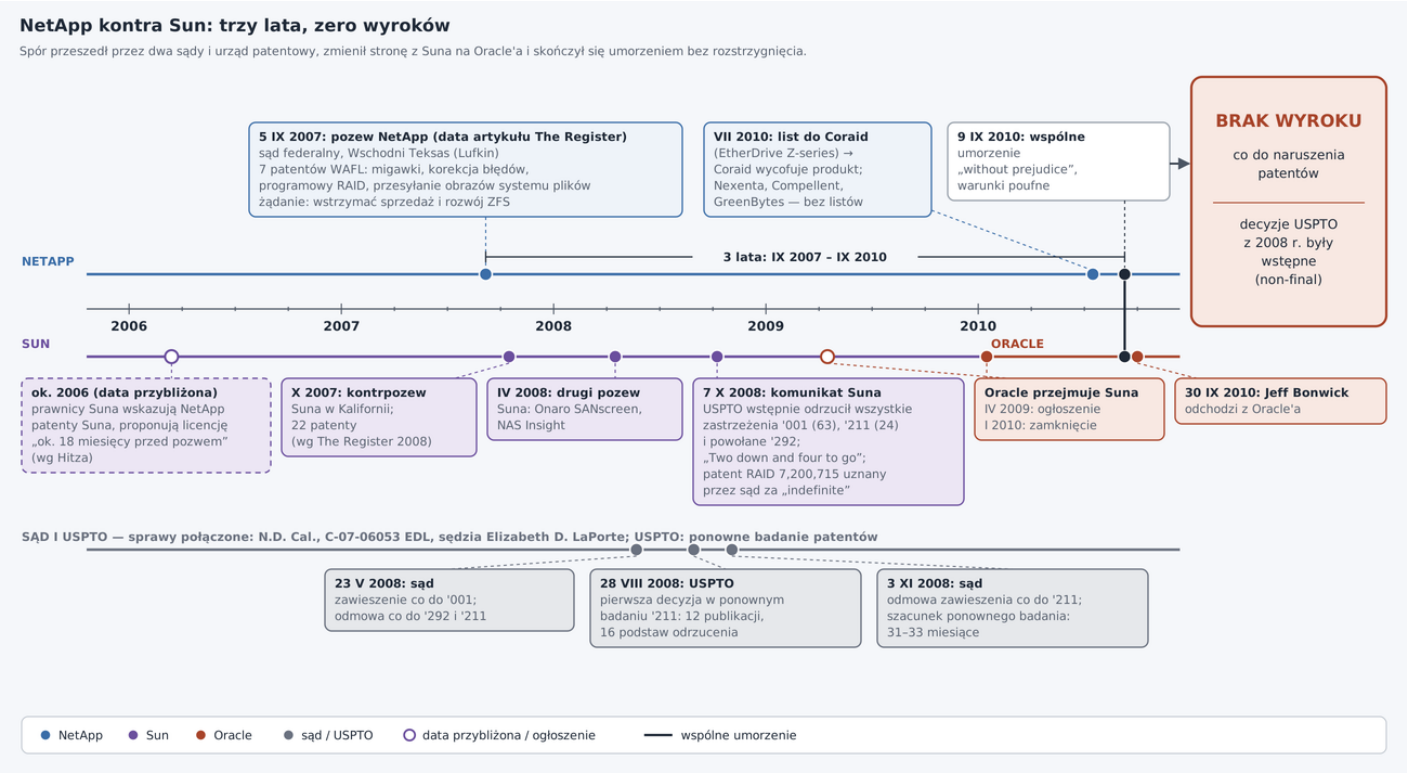
Oracle, Coraid i umorzenie bez przesądzenia

W kwietniu 2009 roku Oracle ogłosił przejęcie Suna, zamknięte w styczniu 2010 roku. Razem z Sunem przejął procesy. W lipcu 2010 roku NetApp wysłał list firmie Coraid, sprzedającej urządzenia NAS EtherDrive z serii Z działające na ZFS-ie, z żądaniem zaprzestania sprzedaży. Coraid się podporządkował. Do innych firm budujących produkty na ZFS-ie — Nexenty, Compellenta, GreenBytes — według The Register podobnych listów nie wysłano.

9 IX 2010 roku NetApp opublikował komunikat „NETAPP AND ORACLE AGREE TO DISMISS LAWSUITS”: „Oracle and NetApp seek to have the lawsuits dismissed without prejudice. The terms of the agreement are confidential.” Strony wnosiły o umorzenie spraw bez przesądzenia (without prejudice), a warunki porozumienia pozostały poufne. Tom Georgens, wtedy prezes NetApp, dodał: „Moving forward, we will continue to collaborate with Oracle…” — nadal będziemy współpracować z Oracle'em.

Z tego wynika rzecz najważniejsza dla obiegowej opowieści: **żaden sąd nie orzekł, czy ZFS narusza patenty WAFL**. Nie wygrał NetApp, nie wygrał Sun. Nie wiadomo też, czy przy porozumieniu płynęły pieniądze albo licencje. Twierdzenie, że „Sun skopiował WAFL”, jest więc tezą jednej strony sporu, której nikt nie rozstrzygnął; twierdzenie, że „NetApp przegrał”, jest tezą drugiej. Ostatecznych wyników ponownych badań patentów WAFL po 2010 roku nie szukałem.

Trzy tygodnie po ugodzie z Oracle'a odszedł twórca ZFS-a. 27 IX 2010 roku Jeff Bonwick ogłosił we wpisie, że 30 IX odchodzi z Oracle'a do startupu działającego w ukryciu („always wanted to try doing a startup”). eWeek przytoczył jego zdanie: „After a decade in the making, ZFS is now an adult” — po dekadzie pracy ZFS jest dorosły. Drugiego współtwórcę, Billa Moore'a, Nexenta przedstawiła 9 IV 2010 roku jako członka swojej rady doradczej. Zbieżność dat z ugodą jest tylko zbieżnością; żadne źródło ich nie łączy.



Rysunek 3. *Spór o ZFS trwał trzy lata, przeszedł przez dwa sądy i urząd patentowy, zmienił jedną ze stron z Suna na Oracle'a — i skończył się bez wyroku. Warto zauważyć, że jedynym namacalnym skutkiem dla rynku było wycofanie produktu przez Coraid.*

FreeNAS: od zapory m0n0wall do TrueNAS-a

Projekt jednego człowieka

Druga połowa tej historii zaczyna się nie od systemu plików, tylko od zapory sieciowej. Dokumentacja FreeNAS 8.2 podaje: „The FreeNAS™ Project was originally founded by Olivier Cochard-Labbé in 2005.” Kod pierwszego FreeNAS-a wywodził się z zapory **m0n0wall**, a gałąź oznaczana jako .7 była napisana w PHP.

FreeNAS działał na FreeBSD, a od FreeBSD 7.0 (2007) system ten miał eksperymentalny ZFS — przeniesiony przez Pawła Jakuba Dawidka, o którym jest mowa w wątku polskim. W FreeNAS 8, opartym na FreeBSD 8.2, ZFS był już podstawą systemu. Źródło, które mówiłoby wprost „FreeNAS korzysta z portu Dawidka”, nie trafiło mi do ręki; wynika to jednak z dat i z tego, na czym FreeNAS był zbudowany.

Rozłam z 2009 roku

W grudniu 2009 roku gałąź .7 przeszła w tryb wyłącznie utrzymania. Dokumentacja FreeNAS opisuje, co stało się dalej, z perspektywy jednej strony. Volker Theile założył **OpenMediaVault** — „a rewrite of FreeNAS™ based on Debian Linux”, przepisany FreeNAS na Debianie, na licencji GPLv3. Wielu użytkownikom się to nie spodobało: zmieniała się licencja, a na Linuksie nie było ZFS-a w jądrze. Firma **iXsystems** „took the initiative to continue the development of a BSD licensed FreeNAS™ solution based on FreeBSD” — podjęła się dalszego rozwoju FreeNAS-a na licencji BSD i na FreeBSD. Następna wersja miała być „rewritten from scratch using a modular design”, napisana od zera, modułowo; gałąź 8.x założono w 2010 roku, a interfejs WWW zbudowano na frameworku Django.

Stary kod też nie umarł. Według wpisu na forum TrueNAS z 2019 roku kod FreeNAS 0.7 wydzielono do projektu **NAS4Free**, „which is now known as XigmaNAS”. Na tym samym forum przytacza się zdanie, że gałąź 0.7 zakończyła życie pod koniec 2011 roku, po tym jak iXsystems prawnie nabyło nazwę FreeNAS — ale to cytat za Wikipedią i nie traktuję go jako potwierdzonego. Kiedy dokładnie powstał NAS4Free, nie ustaliłem.

XigmaNAS: gałąź, która została przy FreeBSD

DistroWatch notuje: „2018-08-06 NAS4Free becomes XigmaNAS” — 6 VIII 2018 roku NAS4Free zmienił nazwę na **XigmaNAS**. Projekt jest aktywny, oparty na FreeBSD, udostępnia pliki przez SMB, NFS, FTP i iSCSI i ma interfejs WWW. Najnowsze wydanie w zebranych źródłach to 14.3.0.5 z października 2025 roku[4]. XigmaNAS jest dziś czymś, czym FreeNAS był przed 2009 rokiem: małym, niezależnym systemem NAS na FreeBSD, dla tych, którzy nie chcą przechodzić na Linuksa razem z TrueNAS-em.

iXsystems: firma od sprzętu

Kim było iXsystems? Podręcznik FreeNAS 8.0.1 przedstawia firmę jako „a provider of FreeBSD-based hardware solutions and professional support” — dostawcę sprzętu z FreeBSD i profesjonalnego wsparcia — i właściciela znaku FreeNAS. Najbarwniejszy portret dał blog Storage Soup serwisu TechTarget (Dave Raffo, tekst niedatowany, z kontekstu z drugiej połowy poprzedniej dekady): prywatna, rentowna firma bez zewnętrznych inwestorów, około 130 pracowników, ponad 4000 klientów, wśród nich Sony, NBC, Uniwersytet Duke'a i NASA, mniej więcej 70% urządzeń budowanych na zamówienie. Projekt FreeNAS przejęła „in late 2009” i przez dwa lata przepisywała system, zanim zaczęła na nim zarabiać. Brett Davis, wiceprezes wykonawczy, mówił: „We're a hardware company in our heart and soul”, „We're private, profitable and self-funded” i „We've been doing open source since way before it was cool” — jesteśmy firmą sprzętową, prywatną, rentowną, finansowaną z własnych pieniędzy, i robiliśmy open source, zanim stało się to modne.

Zdjęcie grupowe uczestników MeetBSD California 2014; autor dziękuje w opisie sponsorom — Weste

Zdjęcie grupowe uczestników MeetBSD California 2014; autor dziękuje w opisie sponsorom — Western Digital i iXsystems. Źródło: Flickr, CC BY 2.0, autor: Andrew Fresh.

FreeNAS 8.0, przeprojektowany od podstaw, wyszedł 2 V 2011 roku. Wersja 8.0.1 z 30 IX 2011 roku działała na FreeBSD 8.2, z ZFS-em w wersji puli 15. Od 2012 roku obok darmowego FreeNAS-a istniał komercyjny **TrueNAS**. Pierwotnie nazywał się „FreeNAS Pro”; nazwę zmieniono, bo — jak pisało iXsystems w 2020 roku — firmy nie traktowały poważnie produktu ze słowem „Free” w nazwie.

TrueNAS CORE: jedna nazwa, jeden obraz

W marcu 2020 roku iXsystems zapowiedziało połączenie obu produktów (prasa pisała o tym 7 III 2020). Od wersji 12.0 miał być jeden obraz systemu i jedna nazwa: **TrueNAS CORE** — darmowy i otwarty — oraz **TrueNAS Enterprise**, z rozszerzeniami odblokowywanymi kluczem licencyjnym na wspieranym sprzęcie. Blog przypominał, że już w wersji 11.3 oba produkty miały wspólne ponad

95% kodu i że „FreeNAS and TrueNAS have been separate-but-related members of the #1 Open Source storage software family since 2012”. Obiecywał też „a free, Open Source edition (this will never change!) and an enterprise edition” — wydanie darmowe i otwarte („to się nigdy nie zmieni!”) obok wydania korporacyjnego. Wydanie CORE 12.0 zaplanowano na trzeci kwartał 2020 roku. Nazwa FreeNAS przeszła do historii.

TrueNAS SCALE: Linux, najpierw „do domu”

22 II 2022 roku iXsystems wydało **TrueNAS SCALE 22.02.0 „Angelfish”** — pierwsze oficjalne wydanie TrueNAS na Linuksie, z „familiar Debian Linux system environment”, znajomym środowiskiem Debiana. Między wersją alfa a wydaniem minęło około 18 miesięcy, od OpenZFS 2.0 — prawie trzy lata pracy, a system testowało ponad 10 000 osób. Najważniejsza zmiana dotyczyła aplikacji: zamiast jaili i wtyczek FreeBSD SCALE uruchamiał je jako kontenery w Kubernetesie, z pakietami Helm i obrazami Dockera. Do tego dochodziło łączenie OpenZFS z GlusterFS w „scale-out ZFS” — rozproszony magazyn na wielu serwerach, rozwijany w kolejnym wydaniu, Bluefin. Kontenery i Kubernetesa opisuje strona **Kontenery: od chroot do Kubernetesa**.

Ważne było też to, do kogo SCALE był skierowany. Blog z lutego 2022 roku pisał wprost: „TrueNAS SCALE is primarily for home usage, smaller projects, and tech labs” — przede wszystkim do domu, mniejszych projektów i laboratoriów — a TrueNAS ma być dostępny „on both FreeBSD (TrueNAS CORE & TrueNAS Enterprise) and Linux with TrueNAS SCALE”. Trzy lata później ta sama linuksowa linia była jedynym kodem TrueNAS-a i podstawą wersji Enterprise.

Droga do tego prowadziła przez kolejne wydania nazywane od ryb. **24.04 „Dragonfish”** stał się najczęściej wdrażaną wersją i wyprzedził CORE 13.0. **24.10 „Electric Eel”**, wydany 29 X 2024 roku, „migrates the previous Kubernetes-based Application back-end to the simpler Docker Compose solution” — zastąpił Kubernetesa prostszym Docker Compose — i wprowadził rozszerzanie grup RAIDZ online o pojedynczy dysk, przy pojedynczej, podwójnej i potrójnej parzystości; klasyczne dokładanie całej grupy dysków (vdev) iXsystems nadal uważało za lepszą metodę.

Fangtooth i Community Edition: koniec dwóch linii

7 I 2025 roku blog TrueNAS ogłosił: „Fangtooth is the new, unified Community Edition.” **TrueNAS 25.04 „Fangtooth”** miał być ścieżką aktualizacji zarówno dla SCALE 24.10, jak i dla CORE 13.x. O CORE blog pisał, że „has been in sustaining mode for the last 18 months” — od półtora roku jest w trybie podtrzymania — a ostatnią dużą aktualizacją było 13.0-U6 z listopada 2024 roku. CORE 13.0 można było przenieść na SCALE 24.10 z zachowaniem danych, udziałów SMB, NFS i iSCSI oraz maszyn wirtualnych; wyjątkiem były jaile, które trzeba było ręcznie przenieść do „Linux

Sandboxes” z narzędziami Dockge albo Portainer, a docelowo zastąpić kontenerami LXC. Fangtooth działał na jądrze Linux 6.12 LTS. iXsystems podawało, że w październiku 2024 roku SCALE i CORE miały mniej więcej tyle samo użytkowników, a liczba systemów SCALE podwoiła się w ciągu roku.

Wydanie 25.04.0 ukazało się w połowie kwietnia 2025 roku[5]. Przyniosło kontenery LXC i maszyny wirtualne QEMU/KVM zarządzane przez Incusa (w interfejsie jako „Instances”), z Secure Boot i TPM — oznaczone jako eksperymentalne — oraz Fast Dedup i, według producenta, „5X faster RAID-Z expansion”. 15 V 2025 roku blog ogłosił zmianę nazwy: SCALE stał się **TrueNAS Community Edition** (CE) — jedną darmową platformą łączącą gałęzie CORE i SCALE dla zastosowań nie-korporacyjnych. CORE „being transitioned to its maintenance phase”, przechodził w fazę utrzymania, bez podanej daty końca. **TrueNAS Enterprise** to od tej pory ten sam kod co CE, uzupełniony o wysoką dostępność (HA), funkcje bezpieczeństwa, integracje i pełne wsparcie, sprzedawany zwykle jako gotowe urządzenia. Kris Moore, starszy wiceprezes ds. inżynierii, zapewniał: „TrueNAS Community Edition isn't a reduction, it's an evolution” i „Everything users valued in CORE or SCALE is here”.

Eksperyment z Incusem dla maszyn wirtualnych nie przetrwał lata. 24 VII 2025 roku, przy wydaniu 25.04.2, iXsystems przywróciło stary mechanizm: „Electric Eel Virtualization is more robust and flexible and is now the approach for adding VMs”. Nowe maszyny wirtualne tworzy się znowu w klasycznym interfejsie „Virtualization”, a „Instances” zostały — nadal eksperymentalnie — dla kontenerów LXC. Producent podawał przy tej okazji, że Fangtootha używa ponad 100 000 użytkowników i że ponad 60% z nich łączy Dockera, ZFS i udostępnianie plików, a CORE 13.0 dostał jeszcze aktualizację 13.0-U8.

Goldeye i TrueNAS 26

TrueNAS 25.10 „Goldeye” ukazał się pod koniec października 2025 roku. Wprowadził NVMe over TCP w Community Edition (NVMe over RDMA — na sprzęcie Enterprise), „Data Hypervisor” łączący KVM, ZFS i wysoką dostępność, instalator obsługiwany przez przeglądarkę, profile aktualizacji (General, Early Adopter, Developer), obsługę sieci 400GbE i DirectIO w ARC i ZIL. 19 II 2026 roku, przy wydaniu 25.10.2, iXsystems uznało Goldeye za wersję „Stable”, zastępującą Fangtootha, Electric Eela, Dragonfisha i TrueNAS 13.0, i podało ponad 100 000 wdrożeń od premiery.

4 II 2026 roku blog „TrueNAS Plans for 2026” ogłosił koniec dotychczasowego rytmu. Od TrueNAS 26 wydania mają się ukazywać raz w roku, bez rybich nazw, z numeracją w rodzaju „26.1” zamiast „25.04.2.6”. Zapowiedziano udostępnianie plików przez przeglądarkę z wyszukiwaniem (WebShare), „wykrywanie i ochronę przed ransomware”, OpenZFS 2.4, w pełni wspierane kontenery LXC jako ścieżkę dla dawnych jaili CORE, jądro Linux 6.18 LTS i warstwowanie danych między NVMe a dyskami twardymi. W tym samym wpisie iXsystems podawało, że w 2025 roku „nearly half a million systems worldwide ran on TrueNAS” — prawie pół miliona systemów; to liczba producenta, podobnie jak „More than 60% of Fortune 500 companies”.

7 IV 2026 roku ukazała się wersja **26-BETA1** — pierwsza z nowej serii rocznej, rozwijanej pod roboczą nazwą „Halfmoon”, z zapowiedzią kolejnych wydań 26.1 i 26.2. Ma jądro Linux 6.18 LTS i OpenZFS 2.4, hybrydowe pule łączące flash i dyski twarde (z przydziałem wybranych zbiorów

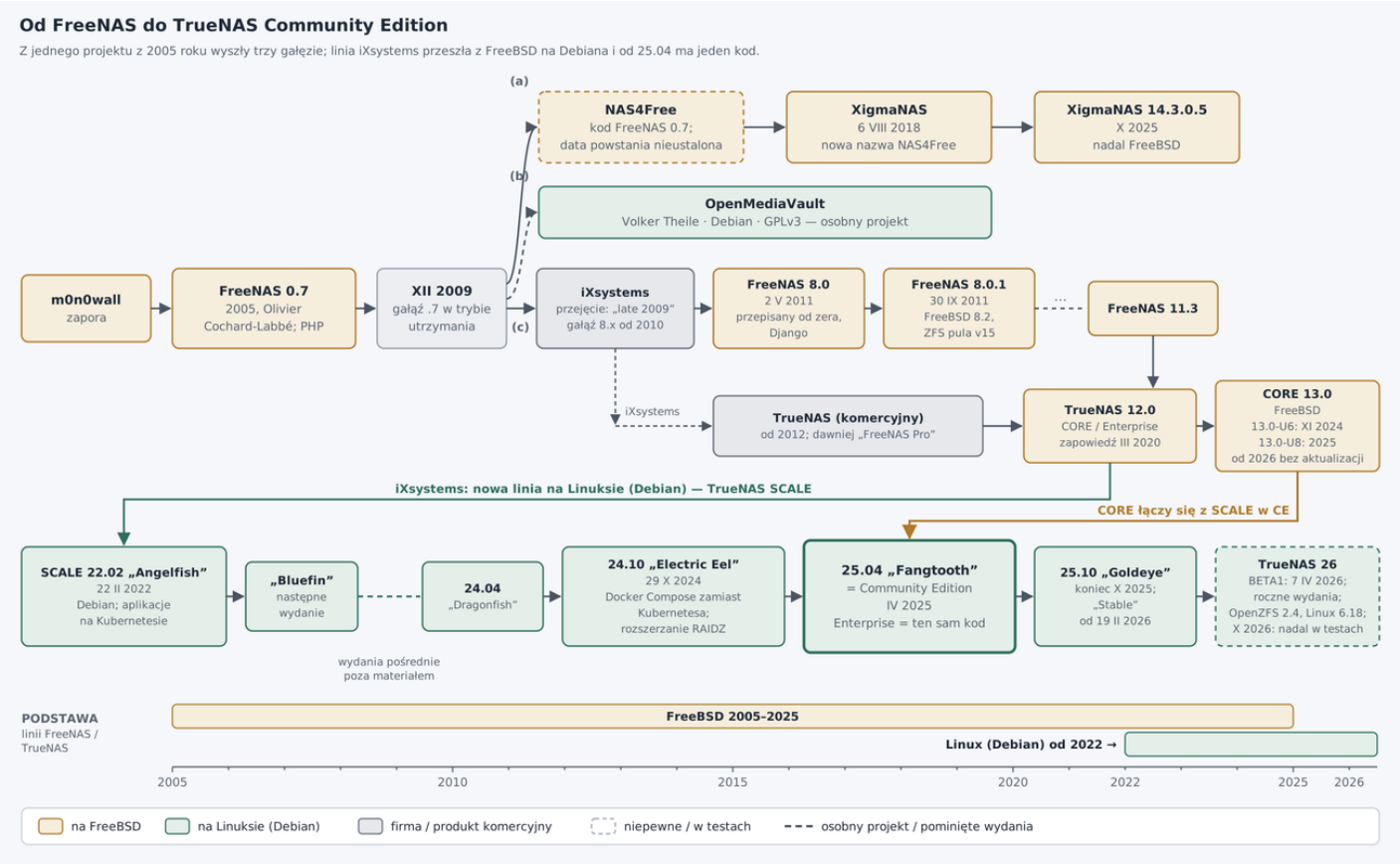
danych i LUN-ów do flash oraz metadanymi i małymi plikami na flash), harmonogram scrubów omijający godziny szczytu, szybsze Fast Dedup, kontenery LXC oraz WebShare i TrueSearch, które wymagają konta w usłudze TrueNAS Connect. Producent liczy „over 700 updates and fixes” i tłumaczy zmianę rytmu jako „a direct response to Community preference for annual releases”. Ścieżka aktualizacji z 25.10 w interfejsie ma się pojawić dopiero od 25.10.3. Istniała też wersja BETA.2; jej opisu nie udało mi się odczytać.

Stan na październik 2026 roku jest więc taki: stabilną linią jest **25.10 Goldeye**, a **TrueNAS 26 jest w fazie testów i wczesnej dokumentacji** — jego dokumentacja nadal nosi oznaczenie „Early Release Documentation - TrueNAS 26”. Komunikatu o wydaniu ostatecznym nie znalazłem. Dokumentacja 26 zapowiada dla migrujących z CORE: „Users migrating from TrueNAS CORE who previously relied on custom Jails can use containers as a supported migration path”, a dla Enterprise — przełączanie kontenerów między kontrolerami HA i stanowy failover SMB.

OpenZFS 2.3 i 2.4: co dołożył TrueNAS

TrueNAS przestał być tylko użytkownikiem ZFS-a. **OpenZFS 2.3.0**, wydany 13 I 2025 roku, przyniósł według Phoronix szybkie deduplikowanie (Fast Dedup), rozszerzanie grup RAIDZ (dokładanie dysków do istniejącego vdev), Direct I/O z pominięciem pamięci podręcznej ARC, długie nazwy plików do 1023 znaków i wyjście w formacie JSON; obsługiwał jądra Linux od 4.18 do 6.12 i FreeBSD 13.3 oraz 14.0–14.2. Blog TrueNAS z 20 I 2025 roku przypisuje sobie udział w dwóch z tych funkcji: zespół TrueNAS „helped sponsor and complete RAIDZ expansion in OpenZFS”, a poprawka jego inżynierów przyspiesza przebudowę danych przy rozszerzaniu (reflow) mniej więcej pięciokrotnie, w niektórych przypadkach do dziesięciu razy; Fast Dedup współtworzyli z firmą Klara Systems. Ten sam blog uczciwie podaje koszt: zapisy z Fast Dedup są o około 60% wolniejsze niż bez deduplikacji, odczyty unikalnych danych — bez zmian.

Klara Systems to ta sama firma, która według Fundacji FreeBSD jest partnerem NetApp przy FreeBSD. W latach 2024–2025 jedna firma pracowała więc i nad systemem, na którym stoi ONTAP, i nad ZFS-em, o który NetApp wiódł spór z Sunem. OpenZFS 2.4 trafił do TrueNAS 26.



Rysunek 4. *Od jednego projektu z 2005 roku odchodzą trzy gałęzie: XigmaNAS został przy FreeBSD, OpenMediaVault od początku był na Debianie, a linia iXsystems przeszła z FreeBSD na Debiana w latach 2022–2025. Warto zauważyć, że produkt reklamowany w 2022 roku jako „głównie do domu" trzy lata później był jedynym kodem TrueNAS-a.*

Ransomware: migawki, które trudno skasować

SnapLock: WORM na dyskach od 2003 roku

Migawka chroni przed skasowaniem pliku przez użytkownika, ale nie przed kimś, kto zdobył uprawnienia administratora i skasuje migawki razem z danymi. NetApp zajął się tym problemem dwadzieścia lat przed obecną falą ransomware, tyle że z innego powodu: przepisów o archiwizacji. W październiku 2003 roku eWeek opisał **SnapLock** — pakiet do tworzenia na zwykłych dyskach woluminów WORM (write once, read many: zapisz raz, czytaj wiele razy), alternatywy dla nośników optycznych, dostępnych przez NFS i CIFS, ale nie przez SAN. Wersja SnapLock Compliance była w sprzedaży od maja 2003 roku, najpierw tylko na systemach NearStore z dyskami ATA.

Zapowiadana na październik wersja Enterprise pozwalała administratorowi usunąć cały wolumin, ale nie pojedyncze pliki. Licencja na 12 TB kosztowała 30 000 dolarów, na 24 TB — 60 000. eWeek pisał: „Once snaplock is set to protect data, there is no way to delete or alter data, short of smashing the hard drives” — gdy SnapLock chroni dane, nie da się ich usunąć ani zmienić, chyba że rozbije się dyski.

Dzisiejszy SnapLock według dokumentacji NetApp ma dwa tryby. **Compliance** chroni dane na poziomie dysku: plików nie da się usunąć przed końcem okresu retencji; tryb jest przeznaczony dla wymogów zewnętrznych, takich jak amerykańska reguła SEC 17a-4(f). **Enterprise** służy politykom wewnętrznym: wyznaczony „compliance administrator” może usunąć pliki WORM operacją „privileged delete”, która jest rejestrowana. Plik zatwierdzony jako WORM nie może być zmieniony nawet po końcu retencji; można go skopiować, ale kopia traci ochronę. Czas mierzą zegary zgodności — wolumenowy (VCC) i systemowy (SCC). Licencja SnapLock wchodzi w skład pakietu ONTAP One, a funkcja działa na macierzach AFF, FAS i w ONTAP Select.

Zablokowane migawki od 9.12.1

Woluminy WORM są dobre do archiwów, ale nie do zwykłych udziałów, na których pliki się zmieniają. Dla nich NetApp wprowadził w ONTAP 9.12.1 **blokowanie migawek** (tamperproof snapshot locking) na zwykłych woluminach, nie-SnapLockowych. System Manager obsługuje je od 9.13.1, a długą retencję na celach replikacji SnapMirror — od 9.14.1. Mechanizm korzysta z zegara zgodności SnapLock, który musi być zainicjowany, i wymaga licencji SnapLock (z pakietu ONTAP One). Retencja może sięgać stu lat i ma pierwszeństwo przed liczbą migawek ustawioną w polityce. Jest też konsekwencja, o której dokumentacja ostrzega: po włączeniu blokady nie można cofnąć wersji ONTAP, dopóki zablokowane migawki nie wygasną. Polecenia z dokumentacji są w części praktycznej.

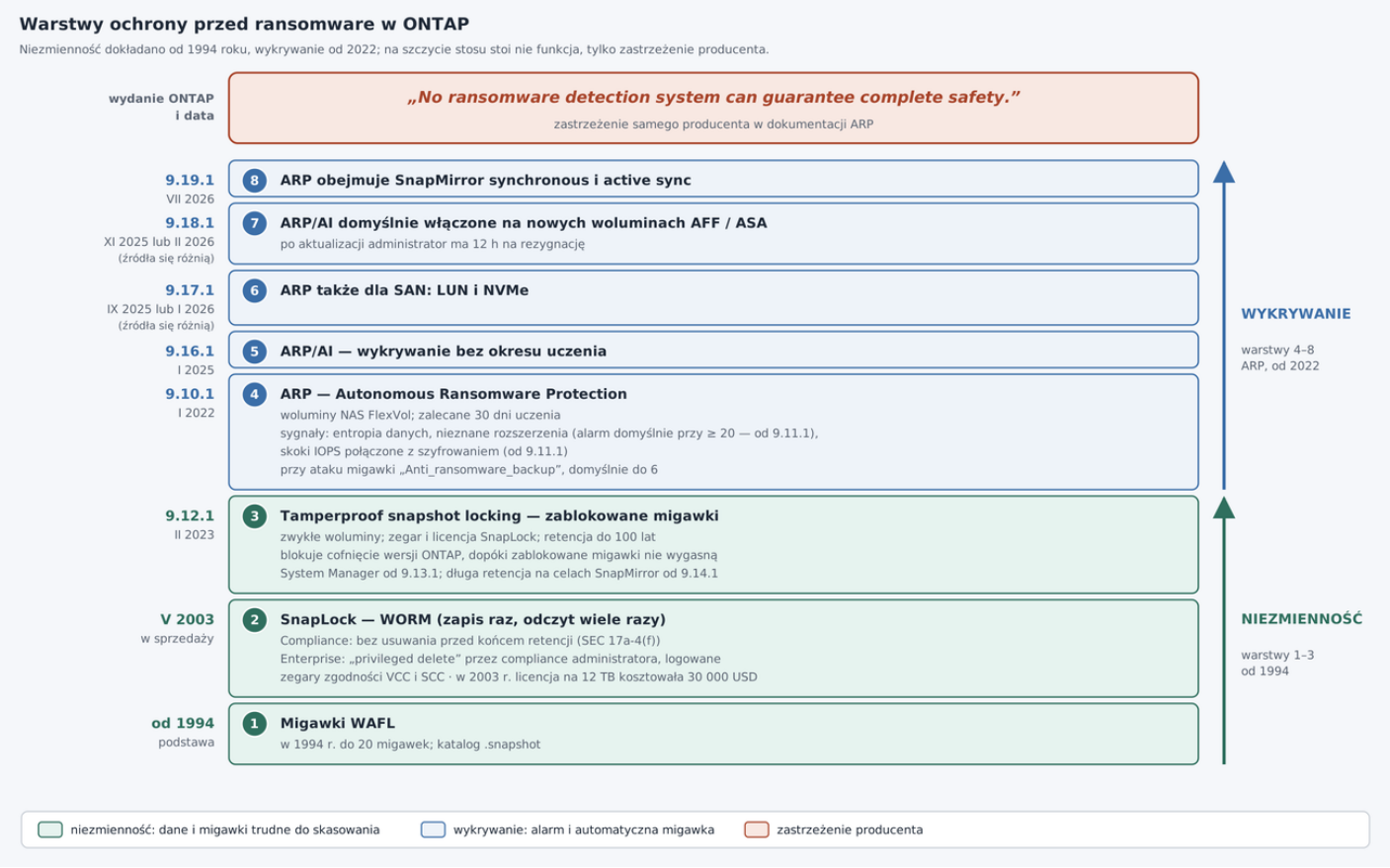
Autonomous Ransomware Protection

Trzecią warstwą jest wykrywanie. **Autonomous Ransomware Protection** (ARP) pojawiło się w ONTAP 9.10.1 dla woluminów NAS typu FlexVol, udostępnianych przez NFS i SMB: najpierw w trybie uczenia, potem aktywnym. Według dokumentacji ARP obserwuje entropię zapisywanych danych (zaszyfrowane dane wyglądają jak losowe), nietypowe rozszerzenia plików, a od 9.11.1 także skoki liczby operacji wejścia-wyjścia połączone z szyfrowaniem. Od 9.11.1 alarm podnosi się domyślnie, gdy w krótkim czasie pojawi się co najmniej 20 nieznanych wcześniej rozszerzeń. Przy pierwszych oznakach ataku ONTAP robi migawkę z przedrostkiem „Anti_ransomware_backup”, zablokowaną przed zwykłym usunięciem — domyślnie do sześciu takich migawek — i trzyma ją, dopóki administrator nie oznaczy zdarzenia jako ataku albo fałszywego alarmu; w tym drugim przypadku ONTAP migawkę usuwa. Dokumentacja zalecała 30 dni uczenia.

Kolejne wydania rozszerzały zasięg. 9.12.1 objęło relacje SnapMirror i SVM DR, 9.13.1 — woluminy FlexGroup, automatyczne przejście z uczenia do trybu aktywnego i zatwierdzanie operacji przez kilku administratorów (multi-admin verification), 9.14.1 — konfigurowalne alerty. W 9.16.1 stary model zastąpiono modelem **ARP/AI**: bez okresu uczenia, bez limitu 4000 unikalnych rozszerzeń, z

aktualizacjami „security bundle” poza cyklem wydań. 9.17.1 rozciągnęło ochronę na SAN — LUN-y i przestrzenie nazw NVMe — z wykrywaniem anomalii szyfrowania i okresem oceny od dwóch do czterech tygodni, a także na hipernadzorców (Hyper-V, KVM, a od 9.17.1P5 OpenStack). W 9.18.1 ARP/AI jest domyślnie włączone na nowych woluminach macierzy AFF A i C, ASA i ASA r2 — po aktualizacji administrator ma 12 godzin na rezygnację. 9.19.1 dodało synchroniczny SnapMirror i active sync.

Najważniejsze zdanie w dokumentacji ARP pisze sam producent: „No ransomware detection system can guarantee complete safety.” ARP ma być „an extra layer of defense if anti-virus software fails to detect an intrusion” — dodatkową warstwą obrony, gdy oprogramowanie antywirusowe nie wykryje włamania. To nie jest zastrzeżenie prawnika dopisane drobnym drukiem, tylko opis tego, czym ten mechanizm jest: alarmem i automatyczną migawką, a nie gwarancją. TrueNAS w tej dziedzinie jest na etapie zapowiedzi: plan na 2026 rok obiecuje „wykrywanie i ochronę przed ransomware”, a płatna usługa TrueNAS Connect Plus — dodatkową ochronę. Jak to będzie działać, w materiałach jeszcze nie ma.



Rysunek 5. *Ochrona przed ransomware w ONTAP to kilka warstw dokładanych przez dwadzieścia lat: od migawek z 1994 roku przez WORM z 2003 po wykrywanie z 2022. Warto zauważyć, że najwyższą warstwą nie jest żadna funkcja, tylko zastrzeżenie samego producenta.*

Praktyka administratora 2026

Ten rozdział idzie w kolejności pytań administratora: co wybrać, jak długo wersja będzie wspierana, jak zaktualizować, jak zabezpieczyć migawki, jak przejść z TrueNAS CORE i ile sprzętu potrzebuje ZFS.

Wybór: macierz z ONTAP czy TrueNAS

Nie rozstrzygam tego za czytelnika; podaję kryteria, które wynikają z materiału. Za ONTAP przemawia przewidywalny cykl wsparcia (trzy lata pełnego, dwa ograniczonego, trzy samoobsługowego), dwa wydania w roku, aktualizacja klastra bez przestoju, replikacja synchroniczna między lokalizacjami (MetroCluster), woluminy SnapLock spełniające wymogi takie jak SEC 17a-4(f), ochrona ARP/AI włączana domyślnie od 9.18.1 i dostępność tego samego systemu w chmurze — jako Cloud Volumes ONTAP w AWS i Azure albo jako usługa Amazon FSx for NetApp ONTAP. Za TrueNAS-em przemawia otwarty kod, darmowa Community Edition, ten sam kod w płatnym Enterprise, sprzedawanym jako gotowe urządzenia z wysoką dostępnością i wsparciem i OpenZFS jako system plików. Gdy wymogi regulacyjne każą trzymać dane w formie niezmiennej — decyduje SnapLock lub jego odpowiednik. Gdy potrzebna jest para kontrolerów z przełączaniem awaryjnym — w grę wchodzi ONTAP albo TrueNAS Enterprise, a nie samodzielnie złożony serwer z CE. Gdy chodzi o laboratorium albo mały budżet — CE.

Cykl wsparcia ONTAP

NetApp opisuje wsparcie wydań ONTAP na trzech poziomach. **Full** (lata 1–3): wsparcie techniczne, analiza przyczyn awarii, pobieranie oprogramowania, wydania poprawkowe (P-releases) i alerty o lukach. **Limited** (lata 4–5): wsparcie techniczne, analiza przyczyn i pobieranie, ale bez nowych wydań poprawkowych. **Self-service** (lata 6–8): tylko dokumentacja online. Od 9.8 „NetApp delivers ONTAP releases twice per calendar year”, z zamiarem wydawania w drugim i czwartym kwartale. Dokumentacja dodaje wprost: „If you are running an ONTAP version prior to 9.10.1, it is likely on Limited Support or Self-Service Support.” Wczesne wydania dziewiątki — 9.0, 9.2, 9.4 — miały według tabeli endoflife.date krótsze wsparcie, około roku pełnego; pełny cykl trzech plus dwóch plus trzech lat dotyczy wydań od 9.8.

Wydanie	Data wydania (NetApp)	Data wydania (endoflife.date)	Koniec Full (endoflife.date)
9.19.1	VII 2026	1 VII 2026	31 VII 2029
9.18.1	XI 2025	4 II 2026	31 I 2029
9.17.1	IX 2025	15 I 2026	30 IX 2028
9.16.1	I 2025	1 I 2025	26 II 2028
9.15.1	VII 2024	1 V 2024	31 VII 2027
9.14.1	I 2024	I 2024	31 I 2027
9.13.1	VI 2023	—	30 VI 2026

Wydanie	Data wydania (NetApp)	Data wydania (endoflife.date)	Koniec Full (endoflife.date)
9.12.1	II 2023	—	28 II 2026
9.11.1 / 9.10.1 / 9.9.1	VII 2022 / I 2022 / VI 2021	—	poza Full (wyliczenie: ponad 3 lata od wydania)

Daty wydań według NetApp i według serwisu endoflife.date w kilku miejscach się rozchodzą: źródła podają, że 9.17.1 wyszło we wrześniu 2025 albo w styczniu 2026, 9.18.1 — w listopadzie 2025 albo w lutym 2026, a 9.15.1 — w lipcu albo w maju 2024 roku[6]. Może to być różnica między wersją kandydującą a ogólnie dostępną, ale tego nie potwierdziłem. Dla praktyki ważniejsze jest co innego: w październiku 2026 roku pełne wsparcie mają wydania od 9.14.1 do 9.19.1, przy czym 9.14.1 traci je w styczniu 2027, a 9.15.1 w lipcu 2027 roku. Kto ma 9.13.1 lub starsze, jest już poza pełnym wsparciem i nie dostanie nowych wydań poprawkowych.

Kiedy aktualizować

NetApp pisze w dokumentacji „When to upgrade”: „You should upgrade your ONTAP software on a regular cadence.” Zalecenie jest konkretne: „NetApp recommends that you run the newest release for 1 year after general availability (GA)” — przez rok od ogólnej dostępności korzystać z najnowszego wydania, a w pozostałej części okresu pełnego wsparcia planować przejście na następne. Poprawki niekrytyczne instalować co trzy do sześciu miesięcy, krytyczne — natychmiast. „New ONTAP releases are entitled to full support for 3 years.”

Aktualizacja bez przestoju

Automatyczna aktualizacja bez przestoju (ANDU, automated nondisruptive upgrade) przechodzi przez węzły klastra po kolei: każdy przekazuje obsługę partnerowi (takeover), aktualizuje się i odbiera ją z powrotem (giveback). Dokumentacja NetApp podaje polecenia w tej kolejności — usunięcie starego pakietu z repozytorium, pobranie nowego, sprawdzenie repozytorium, walidacja, szacunek czasu, właściwa aktualizacja, podgląd postępu i powiadomienie AutoSupport po zakończeniu:

```
cluster image package delete -version <previous_ONTAP_Version>
cluster image package get -url location
cluster image package show-repository
cluster image validate -version <package_version_number>
cluster image update -version <package_version_number> -estimate-only
cluster image update -version <package_version_number>
cluster image show-update-progress
autosupport invoke -node * -type all -message "Finishing_NDU"
```

Uwagi z tej samej strony: klastry z ośmiu i więcej węzłów domyślnie aktualizują się metodą wsadową (batch), a przełącznik `-force-rolling` wymusza aktualizację węzeł po węźle. Po każdym przejęciu i oddaniu obsługi ONTAP domyślnie czeka 8 minut na stabilizację (`-stabilize-minutes`). Przy aktualizacji wieloskokowej (multi-hop) w poleceniu `update` podaje się wersję docelową. Wstrzymaną aktualizację wznawia `cluster image resume-update`, a przerywa `cluster image cancel-update`. Klaster jednowęzłowy nie ma partnera, więc jego aktualizacja wiąże się z przestojem. W konfiguracji MetroCluster najpierw aktualizuje się klaster A, potem B.

Blokowanie migawek i podgląd ARP

Blokadę migawek włącza się na woluminie przy tworzeniu albo później; dokumentacja podaje polecenia z oznaczeniami do uzupełnienia:

```
volume create -vserver <vserver_name> -volume <volume_name> -snapshot-locking-enabled true
volume modify -vserver <vserver_name> -volume <volume_name> -snapshot-locking-enabled true
volume snapshot policy create -policy <policy_name> -enabled true -schedule1 <schedule1_name>
-count1 <maximum snapshots> -retention-period1 <retention_period>
volume modify -volume <volume_name> -vserver <vserver_name> -snapshot-policy <policy_name>
volume snapshot create -volume <volume_name> -snapshot <snapshot name> -snaplock-expiry-time
<expiration_date_time>
```

Przed włączeniem trzeba zainicjować zegar zgodności SnapLock i mieć licencję SnapLock. Warto pamiętać o skutku ubocznym, który łatwo przeoczyć przy planowaniu: dopóki zablokowane migawki nie wygasną, nie da się cofnąć wersji ONTAP. Stan ochrony ARP i zachowanie obciążenia woluminu pokazują polecenia:

```
security anti-ransomware volume show
security anti-ransomware volume workload-behavior show
```

Cykl TrueNAS

Według planu iXsystems z 4 II 2026 roku **TrueNAS 13.0 (CORE) i 24.x** nadal działają, ale nie dostają aktualizacji; kto potrzebuje łatek bezpieczeństwa, ma przejść na 25.10 lub nowszy. To w praktyce koniec CORE. **25.04 Fangtooth** jest najszerzej wdrożony, **25.10 Goldeye** — od 19 II 2026 roku „Stable” i zalecany do nowych wdrożeń. **TrueNAS 26** jest od 7 IV 2026 roku w testach; ścieżka z 25.10 w interfejsie ma być dostępna od 25.10.3. Profile aktualizacji z Goldeye — General, Early Adopter, Developer — pozwalają zdecydować, jak wcześnie system ma dostawać nowe wydania. Od TrueNAS 26 wydania mają być roczne.

Migracja z CORE

Dokumentacja TrueNAS 24.10 („Preparing to Migrate”) jest w tej sprawie stanowcza: „Migrating TrueNAS from CORE to SCALE is a one-way operation” — to operacja jednokierunkowa, a powrót do środowiska rozruchowego CORE może zepsuć system. Przed migracją trzeba zaktualizować CORE do 13.0-U6.2 (albo 13.3 w wersji społecznościowej) i — „Download your system configuration file and a debug file” — pobrać plik konfiguracji i plik diagnostyczny. Nie przechodzą:

- poświadczenia i zadania Cloud Sync dla Microsoft OneDrive;
- szyfrowanie FreeBSD GELI — „TrueNAS SCALE is Linux based, so it does not support FreeBSD GELI encryption”, więc dane z pul GELI trzeba wcześniej przenieść na pule bez GELI;
- dane NIS, wielościeżkowe połączenia SAS (nieobsługiwane od 24.04), parametry systemowe (tunables), środowiska rozruchowe ZFS i dyski samoszyfrujące (SED) skonfigurowane w CORE;
- wtyczki i jaile — trzeba zapisać ich ustawienia i zrobić kopię danych, a potem odtworzyć je jako kontenery (Docker, LXC; w TrueNAS 26 kontenery są wspieraną ścieżką migracji);
- nazwy użytkowników zaczynające się cyfrą — trzeba je zmienić przed migracją.

Udziały AFP zamieniają się w udziały SMB z włączoną zgodnością z AFP. Magazyn i podstawowa konfiguracja maszyn wirtualnych przechodzą, ale przed uruchomieniem trzeba sprawdzić maszyny i ich interfejsy sieciowe. Skrypty startowe i zamykające należy skopiować albo przynajmniej zrobić zrzut ekranu ich konfiguracji. Powłoka `netcli` jest zamieniana na `nologin`. Sama migracja odbywa się przez interfejs albo instalator — obrazem ISO na pamięci USB z opcją „Upgrade Install” albo wyborem gałęzi wydań w interfejsie najnowszego CORE; dokumentacja nie podaje do niej poleceń wiersza poleceń i ja ich nie wymyślam.

Dwie pułapki dotyczą nie CORE, tylko starszych wersji SCALE. Po aktualizacji katalogu 1 VI 2025 roku skończył się okres automatycznej migracji aplikacji z Kubernetesa do Dockera; aplikacje z 24.04 i starszych trzeba instalować ręcznie, a iXsystems zalecało przejście najpierw na 24.10.2.2 albo 25.04. Druga pułapka to maszyny wirtualne utworzone w pierwszych wydaniach 25.04 przez „Instances”: od 25.04.2 zalecanym mechanizmem jest znów klasyczna wirtualizacja, więc przed kolejną aktualizacją warto sprawdzić, którym mechanizmem tworzono każdą maszynę. Z CORE 13 albo 24.04 nie należy więc skakać na najnowsze wydanie „w ciemno”.

ZFS: kontroler, pamięć ECC i ile RAM-u

Kontroler. Dokumentacja OpenZFS jest tu jednoznaczna: „Hardware RAID controllers should not be used with ZFS.” Sprzętowy RAID „will limit opportunities for ZFS to perform self healing on checksum failures” — ogranicza możliwość samonaprawy przy błędach sum kontrolnych, bo ZFS nie widzi poszczególnych dysków. „It is best to use a HBA instead of a RAID controller, for both performance and reliability” — lepiej użyć zwykłego kontrolera HBA. Dokumentacja ostrzega też przed udawaniem HBA przez skonfigurowanie każdego dysku jako osobnej macierzy RAID 0.

Pamięć ECC. Ta sama strona: „Bit flips can have fairly dramatic consequences for all computer filesystems and ZFS is no exception” oraz „No technique used in ZFS (or any other filesystem) is capable of protecting against bit flips. Consequently, ECC Memory is highly recommended.” Pamięć

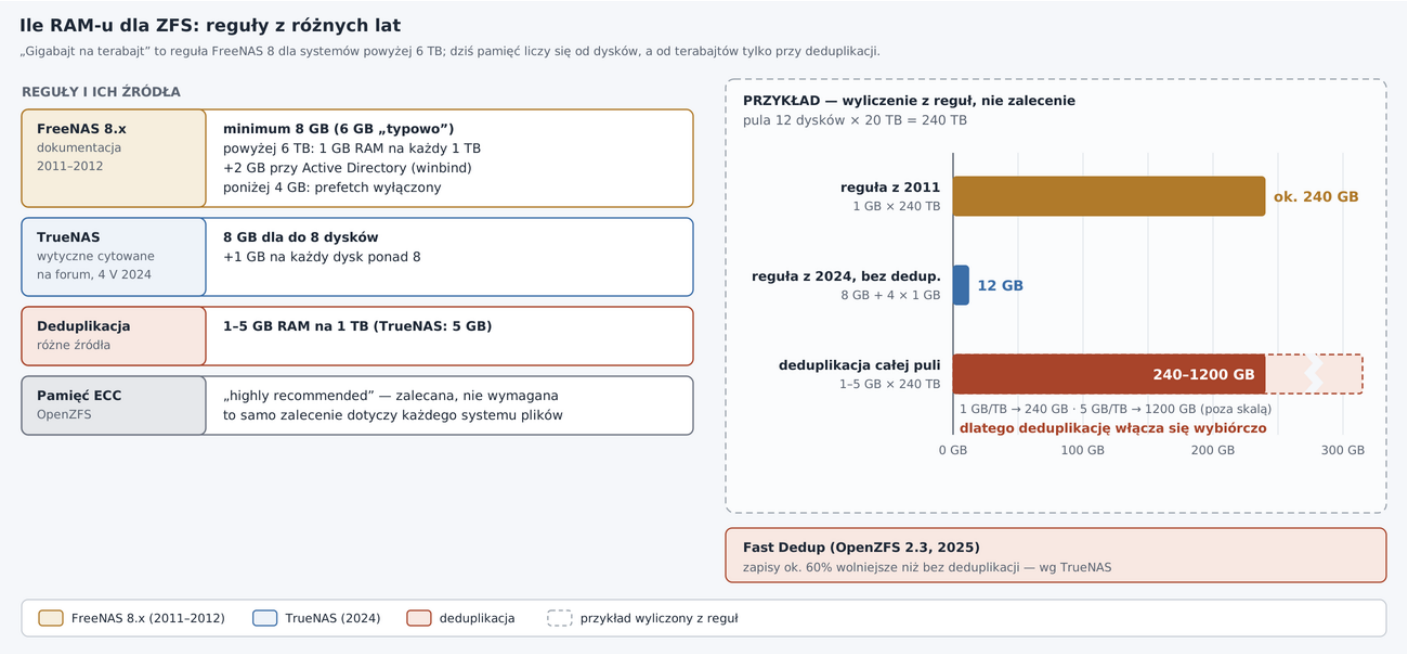
z korekcją błędów jest więc „zdecydowanie zalecana”, ale nie wymagana — i zalecana z tego samego powodu dla każdego systemu plików. W najgorszym przypadku przekłamany bit może sprawić, że puli nie da się zaimportować; przekłamanie, które nastąpi przed policzeniem sumy kontrolnej, przejdzie niewykryte, bo suma zostanie policzona z błędnych danych. Podobnie brzmi zdanie przypisywane Mattowi Ahrensowi, jednemu z twórców ZFS-a: „There's nothing special about ZFS that requires/encourages the use of ECC RAM more so than any other filesystem” — znam je jednak tylko z przytoczenia na innym forum, bo pierwotny wpis był dla mnie niedostępny.

Osobna sprawa to tak zwany „scrub of death”: przekonanie, że scrub — przegląd całej puli — na maszynie z uszkodzoną pamięcią bez ECC zniszczy dane na dyskach, bo ZFS „naprawi” dobre dane złymi. W wątku na forum FreeNAS z 2013 roku uczestnik o nicku jgreco rozdzielał to precyzyjnie: twierdzenie, że scrub ze złą pamięcią „*will* kill your on-disk data”, jest fałszywe, ale że „*could*” — może, na przykład przez uszkodzone metadane — prawdziwe. Richard Yao, deweloper ZFS on Linux, cytowany w tym samym wątku, pisał: „In no way does ZFS ever make a bad situation (checksum failure) worse by replacing good data with bad data.” To spór, nie rozstrzygnięcie; wniosek praktyczny jest wspólny dla obu stron — zła pamięć jest groźna dla każdego systemu plików, a ECC zmniejsza to ryzyko.

Ile RAM-u. Reguła „gigabajt pamięci na terabajt danych” ma konkretne źródło: dokumentację FreeNAS 8.2. Czytamy tam, że „ZFS typically requires a minimum of 6 GB of RAM in order to provide good performance”, w praktyce „the minimum is really 8 GB”, a dalej: „For systems with large disk capacity (greater than 6 TB), a general rule of thumb is 1GB of RAM for every 1TB of storage.” Była to więc reguła kciuka dla systemów większych niż 6 TB, w dokumentacji jednego produktu z 2011–2012 roku, obok uwag, że ZFS wyłącza wczytywanie z wyprzedzeniem (prefetch) przy mniej niż 4 GB pamięci, a integracja z Active Directory potrzebuje dodatkowych 2 GB. Ta sama dokumentacja, co ciekawe, nie zalecała ECC. Na forum FreeNAS reguła żyła potem jako „de facto standard”; w 2020 roku jeden z użytkowników zauważał, że nowa oficjalna strona wymagań podaje już około 8 GB dla systemów do ośmiu dysków.

Wytyczne TrueNAS przytaczane na forum w maju 2024 roku brzmią tak: „You should have 8 GB of RAM for basic TrueNAS operations with up to eight drives”, „Add 1 GB for each drive added after eight…” i „Add the suggested 5 GB per TB of storage for deduplication that depends on an in-RAM deduplication table”. Reguła „na terabajt” dotyczy więc dziś tylko deduplikacji, której tablica musi mieścić się w pamięci; ile na terabajt, źródła podają różnie — od 1 do 5 GB[7]. W tym samym wątku użytkownicy opisywali problemy z wydajnością po przejściu na Dragonfish (24.04), gdy pamięć podręczna ARC zajmowała ponad połowę RAM-u, i pisali, że „Limiting the ARC maximum size to 50% of RAM resolved the problem” — to hipoteza użytkowników, nie zalecenie producenta.

Wniosek dla praktyki: deduplikację włączać tylko świadomie. Nawet Fast Dedup, według samego TrueNAS-a, spowalnia zapisy o około 60%. Bez deduplikacji pamięć dobiera się do liczby dysków i obciążenia, a nie do terabajtów.



Rysunek 6. *Reguła „gigabajt na terabajt" pochodzi z dokumentacji FreeNAS 8 dla systemów większych niż 6 TB; dzisiejsze wytyczne liczą pamięć od liczby dysków, a „na terabajt" — tylko przy deduplikacji. Warto zauważyć, jak różne wychodzą liczby dla tej samej puli 240 TB.*

Scrub: regularny przegląd puli

Scrub czyta wszystkie dane w puli i sprawdza sumę kontrolną każdego bloku — w odróżnieniu od resilveringu, który nie czyta całej puli. Strona podręcznika `zpool-scrub` z bieżącej gałęzi OpenZFS: „For replicated (mirror, raidz, or draid) devices, ZFS automatically repairs any damage discovered during the scrub" — na urządzeniach z redundancją ZFS sam naprawia znalezione uszkodzenia. Pełna składnia według tej strony to `zpool scrub [-w] [-t] [-C | [-S date] [-E date]] -a|pool…`; tryb `-t` dodatkowo odszyfrowuje i dekompresuje dane. Opcje `-C`, `-S`, `-E` i `-t` pochodzą z gałęzi rozwojowej i mogą nie występować w starszych wersjach — przed użyciem trzeba sprawdzić stronę podręcznika własnego systemu. Podstawowe polecenia:

```
zpool scrub <pool>
zpool scrub -p <pool>
zpool scrub -s <pool>
zpool status
```

Pierwsze uruchamia scrub, drugie go wstrzymuje, trzecie zatrzymuje, a `zpool status` pokazuje postęp. Na systemach z systemd ta sama strona podaje gotowe timery:

```
systemctl enable zfs-scrub-weekly@rpool.timer --now
systemctl enable zfs-scrub-monthly@otherpool.timer --now
```

W TrueNAS scruby planuje się w interfejsie; TrueNAS 26 dodaje harmonogram, który omija godziny szczytu.

Mity i sprostowania

Człon opowieści	Werdykt	Dlaczego
„Trzech założycieli, 1992”	broni się, z niuansem	Reference for Business i artykuł USENIX: Malcolm, Hitz, Lau (IV 1992); strona historii NetApp i CHM wymieniają tylko Hitza i Lau; Malcolm stracił stanowisko prezesa w X 1994.
„…z Auspexu”	częściowo	Hitz i Lau pracowali w Auspexie; o Malcolmie nie ustaliłem.
„Toster”	broni się jako przydomek	Strona historii NetApp: „nicknamed the Toaster”; w artykule z 1994 roku słowa brak.
„WAFL to system log-structured”	przewraca się	Artykuł opisuje zapis gdziekolwiek z punktami spójności, porównany do shadow paging z System R i klonów z Episode; LFS przywołany tylko przy obciążeniach zapisem.
„Migawka WAFL kopiuje dane”	przewraca się	Kopiowany jest i-węzeł główny; tydzień migawek zajmował w 1994 roku 10–20% miejsca.
„ONTAP 9 nie ma nic wspólnego z FreeBSD” / „ONTAP to Linux”	przewraca się	Fundacja FreeBSD: FreeBSD bazą ONTAP od 2004 roku (od 6.0 do 14.0); NetApp: „Multiple NetApp products incorporate FreeBSD”. Dla Linuksa brak źródła.
„Sun skopiował WAFL”	nie da się ustalić	Żaden sąd tego nie orzekł; to teza jednej strony sporu.
„NetApp wygrał z Sunem”	przewraca się	9 IX 2010 obustronne umorzenie „without prejudice”, warunki poufne.
„Sun wygrał, bo USPTO unieważnił patenty”	przewraca się	Odrzucenia z 2008 roku były wstępne (non-final); sąd szacował ponowne badanie na 31–33 miesiące; ugoda zapadła wcześniej.
„FreeNAS to projekt hobbystyczny”	częściowo	Start w 2005 roku jako projekt jednej osoby; od 2009/2010 rozwijany przez iXsystems, przepisany (8.0, 2 V 2011), od 2012 obok komercyjnego TrueNAS.

Człon opowieści	Werdykt	Dlaczego
„iXsystems porzuciło FreeBSD”	broni się (dla produktu)	Od 25.04 jedynym kodem jest linuksowa Community Edition, także w Enterprise; CORE 13.0 bez aktualizacji od 2026 roku. FreeBSD został przy XigmaNAS.
„ZFS wymaga pamięci ECC”	przewraca się	OpenZFS: „highly recommended”; przekłamania bitów szkodzą każdemu systemowi plików.
„ZFS potrzebuje 1 GB RAM na 1 TB”	przewraca się jako wymóg	Reguła kciuka FreeNAS 8.x dla systemów powyżej 6 TB; dziś 8 GB do ośmiu dysków plus około 1 GB na dysk, 1–5 GB/TB tylko przy deduplikacji.
„Scrub bez ECC zniszczy pulę”	sporne	jgreco: „could”, nie „will”; Richard Yao: ZFS nie zastępuje dobrych danych złymi.
„ZFS na sprzętowym RAID jest bezpieczniejszy”	przewraca się	OpenZFS: „Hardware RAID controllers should not be used with ZFS”.
„Do RAIDZ nie da się dołożyć jednego dysku”	nieaktualne	Rozszerzanie RAIDZ od OpenZFS 2.3 (13 I 2025) i TrueNAS 24.10.
„Zablokowana migawka to to samo co SnapLock”	częściowo	Blokada (od 9.12.1) używa zegara i licencji SnapLock, ale działa na zwykłych woluminach; SnapLock Compliance i Enterprise to woluminy WORM z innymi regułami.
„ARP chroni przed ransomware”	częściowo	Producent: „No ransomware detection system can guarantee complete safety” — to dodatkowa warstwa, nie gwarancja.

Epilog

W 1994 roku trzej założyciele Network Appliance opisali system plików, który nigdy nie nadpisuje danych w miejscu, i zbudowali na nim urządzenie robiące jedną rzecz. Trzy dekady później ten sam pomysł działa w klastrach ONTAP, w chmurze Amazona i — w innej implementacji — w każdej puli ZFS, od serwera w piwnicy po urządzenia TrueNAS Enterprise. Spór o to, czy drugi pomysł wziął się z pierwszego, nie dostał wyroku i już go nie dostanie. Ironią tej historii jest to, że obie strony skończyły bliżej siebie, niż zaczynały: NetApp na FreeBSD, z którego wyrósł FreeNAS, a TrueNAS na Linuksie, o którym w 2022 roku pisano, że jest „do domu”. Administrator w 2026 roku nie musi tego rozstrzygać. Musi wiedzieć, ile zostało pełnego wsparcia jego wydaniu, czy jego migawki da się skasować i czy dane z CORE przejdą na nowy system — i to akurat da się sprawdzić w dokumentacji.

Ciekawostki zebrane

- Pierwszy wolumin WAFL miał według Hitza 6 GB; w 1994 roku filer z ponad 20 GB danych wstawał po awarii w około minutę, bez `fsck`.
- Hitz przed informatyką pracował jako kowboj, a jego wspomnienia noszą tytuł „How to Castrate a Bull”.
- Prototyp klastra GX pracował w ILM przy „Gwiezdnych wojnach: Części III” — 3000 Opteronów z SUSE Linux i wirtualny dysk 200 TB.
- W 2010 roku NetApp skłonił Coraid do wycofania NAS-a na ZFS-ie, ale do Nexenty listu nie wysłał.
- Gałąź, w której Dawidek przenosił ZFS do FreeBSD, miała opis „This is not a ZFS port!”.
- OpenMediaVault, „a rewrite of FreeNAS based on Debian Linux”, powstał dwanaście lat przed linuksowym TrueNAS SCALE.
- Według przykładu NetApp z 2004 roku macierz z pojedynczą parzystością traciła dane średnio 0,028 raza na rok — głównie przez błąd nośnika w czasie odbudowy.

„Jak nie wysyłać klastra filerów NetApp za 200 000 euro” — tak autor zatytułował zdjęcie szafy zapa

„Jak nie wysyłać klastra filerów NetApp za 200 000 euro” — tak autor zatytułował zdjęcie szafy zapakowanej do skrzyni bez amortyzacji. Źródło: Flickr, CC BY 2.0, autor: Robert (Jemimus).

Oś czasu

Data	Wydarzenie
IV 1992	Założenie Network Appliance
19 I 1994	Artykuł o WAFL na USENIX Winter w San Francisco
XI 1995	Debiut giełdowy (36 mln USD)
V 2003	SnapLock Compliance w sprzedaży
4 XI 2003	Ogłoszenie przejęcia Spinnakera (ok. 300 mln USD w akcjach)
18 II 2004	Zamknięcie przejęcia Spinnakera
31 III – 2 IV 2004	FAST '04: RAID-DP, wdrożony w Data ONTAP 6.5
2004	FreeBSD (6.0) podstawą ONTAP
IX 2004	Sun zapowiada ZFS
XI 2004	Data ONTAP 7G: FlexVol i FlexClone
2005	Cochard-Labbé zakłada FreeNAS

Data	Wydarzenie
12 VI 2006	Data ONTAP GX
12–19 VIII 2006	Dawidek zaczyna port ZFS do FreeBSD; pierwszy mount
6 IV 2007	ZFS w gałęzi HEAD FreeBSD, eksperymentalnie w 7.0
IX 2007	NetApp pozywa Suna w Teksasie
X 2007	Kontrpozew Suna w Kalifornii
2008	Network Appliance zmienia nazwę na NetApp
3 IX 2009 / 19 III 2010	Data ONTAP 8.0 RC / GA (7-Mode i Cluster-Mode)
XII 2009	FreeNAS 0.7 w trybie utrzymania; projekt w rękach iXsystems
9 IX 2010	NetApp i Oracle umarzają spory
30 IX 2010	Bonwick odchodzi z Oracle'a
2 V 2011	FreeNAS 8.0, przepisany od zera
2012	Komercyjny TrueNAS obok FreeNAS
28 X 2014	Clustered Data ONTAP 8.3 — tylko klaster
31 V 2016	Zapowiedź ONTAP 9 z RAID-TEC
7 VI 2018	Cloud Volumes ONTAP
6 VIII 2018	NAS4Free staje się XigmaNAS
III 2020	Zapowiedź TrueNAS 12.0 CORE i Enterprise
X 2020	ONTAP 9.8; odtąd dwa wydania rocznie
2 IX 2021	Amazon FSx for NetApp ONTAP
I 2022	ONTAP 9.10.1 z pierwszym ARP
22 II 2022	TrueNAS SCALE 22.02 „Angelfish”
II 2023	ONTAP 9.12.1: blokowanie migawek
29 X 2024	TrueNAS 24.10 „Electric Eel”: Docker zamiast Kubernetesa
13 I 2025	OpenZFS 2.3.0
IV 2025	TrueNAS 25.04 „Fangtooth”; od 15 V nazwa Community Edition
24 VII 2025	TrueNAS 25.04.2: powrót klasycznej wirtualizacji
X 2025	TrueNAS 25.10 „Goldeye”
4 II 2026	Plan TrueNAS 26; 13.0 i 24.x bez aktualizacji
7 IV 2026	TrueNAS 26-BETA1
VII 2026	ONTAP 9.19.1

Wątek polski

Paweł Jakub Dawidek i ZFS we FreeBSD

Najważniejszy polski ślad w tej historii prowadzi przez ZFS. W FreeBSD Journal z maja–czerwca 2023 roku Paweł Jakub Dawidek opisał w tekście „How ZFS Made Its Way into FreeBSD”, jak to się zaczęło. Latem 2005 roku, na urlopie na Mazurach, dostał od znajomego z polskiej firmy telekomunikacyjnej wydrukowane ogłoszenie Suna o ZFS. Był wtedy od 2003 roku commiterem kodu źródłowego FreeBSD, zajmującym się podsystemem GEOM. Port zaczął 12 VIII 2006 roku, a 19 VIII po raz pierwszy zamontował ZFS w jądrze FreeBSD. Samo załadowanie modułu `zfs.ko` zajęło mu pięć dni — przerobił w tym celu linker tak, żeby wypisywał wszystkie brakujące symbole naraz. Prototyp z odczytem i zapisem miał po około dziesięciu dniach, choć zakładał, że na wersję tylko do odczytu będzie potrzebował pół roku. Gałąź w repozytorium Perforce opisał przekornie: „This is not a ZFS port!”. Pisze też o sobie, że dorastał w małym polskim miasteczku, pierwszym komputerem był Commodore 64, a programować w BASIC-u zaczął w wieku dwunastu lat dzięki kuzynowi. Dziś jest współzałożycielem i dyrektorem technicznym Fudo Security. Tekst kończy: „Long live OpenZFS! Long live FreeBSD!” (FreeBSD Journal, 2023).

Ogłoszenie na liście freebsd-current z 6 IV 2007 roku (archiwum listy) brzmiało: „ZFS is available in the HEAD branch and will be available in FreeBSD 7.0-RELEASE as an experimental feature”. Na razie tylko dla i386, jako moduł, z obietnicą amd64 wkrótce; bez celu iSCSI, list ACL, atrybutów rozszerzonych i rozruchu z ZFS-a. Jako wspierających wymieniono trzy podmioty: Wheel LTD, Fundację FreeBSD i Sentex. W FreeBSD 7.0 ZFS był eksperymentalny, od 8.0 (2009) — produkcyjny. Opis techniczny portu jest w pracy Dawidka z 2007 roku (papers.freebsd.org). Na FreeBSD z ZFS-em zbudowano potem FreeNAS 8 i całą linię, która doprowadziła do TrueNAS CORE.

ZTM Poznań: PEKA na MetroClusterze

Polskie studium przypadku NetApp (stopka z 2018 roku) opisuje Zarząd Transportu Miejskiego w Poznaniu i system biletowy **PEKA** (netapp.com). ZTM korzysta z ONTAP 9 i MetroClustera — aktywnego centrum danych z synchroniczną replikacją między dwiema lokalizacjami — na macierzach FAS8200, w środowisku Microsoft i VMware; jako partnera wdrożeniowego dokument wymienia firmę zapisaną jako „Hardosft – Telekom”. Skalę oddają liczby z dokumentu: ponad 500 000 pasażerów w aglomeracji, 6000 urządzeń, ponad 4000 czytników kart w pojazdach, komputery pokładowe w ponad 700 pojazdach, 50 serwerów wirtualnych i 10 zintegrowanych systemów. Twierdzenie o poprawie jakości usług o 70% jest tekstem marketingowym. Daniel Wawrzyniak, dyrektor techniczny ds. publicznego transportu zbiorowego ZTM, mówi w nim: „Dzięki uniwersalnym interfejsom NetApp możemy z łatwością rozszerzyć centralny system PEKA” — na kolejne usługi, systemy i chmurę bez dodatkowych licencji.

Źródła i dalsze lektury

Przypisy

[1] Trzech założycieli (Malcolm, Hitz, Lau; IV 1992): Reference for Business, autorzy artykułu: USENIX 1994; dwóch (Hitz i Lau): strona historii NetApp, Computer History Museum.

[2] Zapowiedź 31 V 2016, RC1 pod koniec VI 2016: greatwhitetec; wydanie 9.0 — 1 IX 2016: endoflife.date.

[3] 22 patenty w kontrpozwie Suna: The Register, 7 X 2008; groźba kontrpozwu z ok. 45–56 patentami związanymi z ZFS (źródło wiki, niepotwierdzone): endsoftwarepatents.

[4] 14.3.0.5 — 19 X 2025: DistroWatch (kopia strony); 14.3.0.5.10566 — 28 X 2025: MajorGeeks.

[5] 14 IV 2025: blog wydania Fangtooth; 15 IV 2025 („officially released on April 15, 2025”): blog Community Edition, 15 V 2025.

[6] 9.15.1 — VII 2024, 9.17.1 — IX 2025, 9.18.1 — XI 2025: NetApp, ONTAP release support; 9.15.1 — 1 V 2024, 9.17.1 — 15 I 2026, 9.18.1 — 4 II 2026: endoflife.date (serwis społecznościowy, stan 4 X 2026).

[7] 5 GB RAM na 1 TB przy deduplikacji: wytyczne TrueNAS przytoczone na forum TrueNAS (4 V 2024); 1–5 GB na 1 TB jako reguła kciuka: strona **Solaris i SunOS**.

Lektury

- WAFL i RAID: Hitz, Lau, Malcolm (1994, kopia Princeton), NetApp ATG, Row-Diagonal Parity (FAST '04)
- Rodowód ONTAP: Spinnaker (2003), 8-K (2004), 7G (ESF), 7G (The Register), GX (The Register), GX (Computerworld), Data ONTAP 8, forum o 8.0, 8.3
- FreeBSD w ONTAP: Fundacja FreeBSD (III 2024), (XI 2024), NTAP-20200413-0001, NTAP-20210713-0002
- Spór o ZFS: Slashdot (2004), pozew (2007), postanowienie z 3 XI 2008, podsumowanie (2010), komunikat o umorzeniu, odejście Bonwicka

- FreeNAS i TrueNAS: podręcznik 8.0.1, podręcznik 8.2, TechTarget o iXsystems, połączenie (2020), SCALE (2022), Electric Eel, Fangtooth, 25.04.2, Goldeye, 25.10.2, plany 2026, 26-BETA1, dokumentacja 26
- Praktyka: kiedy aktualizować, ANDU, SnapLock, blokowanie migawek, ARP, historia ARP, migracja z CORE, termin migracji aplikacji
- ZFS: OpenZFS — sprzęt, zpool-scrub, OpenZFS 2.3, TrueNAS o OpenZFS 2.3, „scrub of death”, wymagania RAM (2024), SnapLock w 2003 roku

Ilustracje

Diagramy i animacje są moje. Zdjęcia pochodzą ze zbiorów cyfrowych muzeów, kolekcji i serwisu Flickr, które udostępniają je jako domenę publiczną albo na licencjach Creative Commons; część jest na licencjach niekomercyjnych (NC) lub bez utworów zależnych (ND) — ta wiki jest prywatna i niekomercyjna, obrazy pokazujemy bez zmian. Obrazy są wyświetlane bezpośrednio z serwerów instytucji i autorów; licencja i autor stoją w każdym podpisie. Materiałów objętych zastrzeżeniem praw nie osadzamy — odsyłamy do nich w źródłach.

Nota metodyczna: czego nie wiemy

- **Wydanie ostateczne TrueNAS 26** — do 19 X 2026 roku nie znalazłem komunikatu; dokumentacja ma status „Early Release”, a opisu wersji BETA.2 nie udało się odczytać.
- **Dokładne daty ogólnej dostępności ONTAP 9.15.1, 9.17.1 i 9.18.1** — NetApp i endoflife.date się różnią; przyczyny nie ustaliłem.
- **Malcolm przed 1992 rokiem** i to, czy „Toaster” był nazwą prototypu, czy tylko przydomkiem — mam jedno zdanie ze strony NetApp.
- **Co z FreeBSD było w 7-Mode** i jak wyglądało jądro ONTAP przed 2004 rokiem — biuletyn z 2020 roku nie precyzuje.
- **Rozmiar NVRAM i cena** pierwszych filerów, data wydania Data ONTAP 6.5.
- **Ostateczne wyniki ponownych badań patentów WAFL** po 2010 roku; startup, do którego przeszedł Bonwick.
- **Data powstania NAS4Free** i data nabycia znaku FreeNAS przez iXsystems.
- **Cytaty z drugiej ręki**: Ahrens o ECC (pierwotny wpis niedostępny) i Bonwick o „gotowaniu oceanów” (komentarz na Slashdocie).

Strona opiera się na artykułach naukowych NetApp z konferencji USENIX, dokumentach SEC, dokumentacji NetApp, iXsystems i OpenZFS, blogach producentów, postanowieniach sądu, materiałach Fundacji FreeBSD, forach NetApp i TrueNAS oraz prasie branżowej. Liczby użytkowników, wdrożeń i wydajności pochodzą przeważnie od producentów i tak są opisane; dane z

forów traktuję jako relacje użytkowników.

Revision #7
Created 2026-10-19 15:27:54 UTC by Tomasz Konieczny
Updated 2026-11-01 17:34:54 UTC by Tomasz Konieczny