



# AGH

**AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W  
KRAKOWIE**

**WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI,  
INFORMATYKI I INŻYNIERII BIOMEDYCZNEJ**

KATEDRA INFORMATYKI STOSOWANEJ

Praca dyplomowa inżynierska

*dHTTP - Rozproszony system wsparcia  
serwerów sieci web*

*dHTTP - Distributed companion  
for central-server based web*

Autor:	<i>Dominik Adamiak</i>
Kierunek studiów:	<i>Informatyka</i>
Opiekun pracy:	<i>dr inż. Piotr Matyasik</i>

Kraków, 2018

*Uprzedzony o odpowiedzialności karnej na podstawie art. 115 ust. 1 i 2 ustawy z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (t.j. Dz.U. z 2006 r. Nr 90, poz. 631 z późn. zm.): „Kto przywłaszcza sobie autorstwo albo wprowadza w błąd co do autorstwa całości lub części cudzego utworu albo artystycznego wykonania, podlega grzywnie, karze ograniczenia wolności albo pozbawienia wolności do lat 3. Tej samej karze podlega, kto rozpowszechnia bez podania nazwiska lub pseudonimu twórcy cudzy utwór w wersji oryginalnej albo w postaci opracowania, artystycznego wykonania albo publicznie zniekształca taki utwór; artystyczne wykonanie, fonogram, wideogram lub nadanie.”, a także uprzedzony o odpowiedzialności dyscyplinarnej na podstawie art. 211 ust. 1 ustawy z dnia 27 lipca 2005 r. Prawo o szkolnictwie wyższym (t.j. Dz. U. z 2012 r. poz. 572, z późn. zm.): „Za naruszenie przepisów obowiązujących w uczelni oraz za czyny uchybiające godności studenta student ponosi odpowiedzialność dyscyplinarną przed komisją dyscyplinarną albo przed sądem koleżeńskim samorządu studenckiego, zwanym dalej «sądem koleżeńskim».”, oświadczam, że niniejszą pracę dyplomową wykonałem(-am) osobiście i samodzielnie i że nie korzystałem(-am) ze źródeł innych niż wymienione w pracy.*

*Serdecznie dziękuję promotorowi za wolność działania, a rodzinie i przyjaciołom za wiarę w to, że się uda.*



# Spis treści

<b>1. Wprowadzenie .....</b>	<b>7</b>
1.1. Zawartość.....	7
1.2. Cele pracy .....	7
<b>2. Droga do rozproszenia .....</b>	<b>9</b>
2.1. Zarys teoretyczny .....	9
2.2. Historia i istniejące narzędzia.....	9
2.3. Podstawa projektu.....	9
<b>3. Projektowanie i implementacja .....</b>	<b>11</b>
3.1. Wymagania funkcjonalne .....	11
3.2. Definicje, architektura i technologie.....	11
3.2.1. Wykorzystane technologie i narzędzia .....	11
3.2.2. Problemy warstwy sieciowej .....	11
3.2.3. Decentralizacja czy rozproszenie? .....	11
3.2.4. Bezpieczeństwo .....	11
3.2.5. Propagacja i przechowywanie danych.....	11
3.3. Aplikacje.....	11
3.3.1. Węzeł .....	11
3.3.2. Klient .....	11
<b>4. Testy i optymalizacja.....</b>	<b>13</b>
4.1. Zarys użyteczności i działania systemu .....	13
4.2. Wydajność systemu .....	13
4.2.1. Wydajność klienta .....	13
4.2.2. Wydajność serwera .....	13
4.3. Podsumowanie rozwiązań optymalizacyjnych .....	13
<b>5. Podsumowanie .....</b>	<b>15</b>
5.1. Potencjalne kierunki rozwoju .....	15



# 1. Wprowadzenie

Dzisiejsza informatyka jak nigdy stoi przed wyzwaniami związanymi z wydajnością, których nie może rozwiązywać tylko poprzez skalowanie wertykalne – próbę zwiększania wydajności pojedynczych węzłów. W natłoku zapytań poradzić sobie można dzięki łatwości skalowania horyzontalnego – możliwości rozszerzania *floty* urządzeń, co wymaga odpowiedniego sposobu projektowania, od sprzętu, przez protokoły, na samej architekturze systemów informatycznych kończąc.

## 1.1. Zawartość

Rozdział 1. stanowi krótkie wprowadzenie i definicję celu pracy.

Rozdział 2. zawiera rozważania na temat teoretycznych podstaw systemów rozproszonych, historię ich implementacji, oraz przykłady najpopularniejszych narzędzi je wykorzystujących. Wprowadza także do teoretycznych i narzędziowych podstaw implementacji systemu dHTTP.

Rozdział 3. jest szczegółowym opisem wymagań, projektu, modułów, a także implementacji samego systemu, w szczególności omawiającym funkcjonalności, biblioteki i narzędzia pozwalające na działanie aplikacji, napotkane w toku projektowania problemy i ich rozwiązania, rys architektoniczny sposobu przechowywania danych czy ich propagacji, w rozróżnieniu podejścia węzłowego i podejścia klienta końcowego.

Rozdział 4. to próba walidacji systemu dHTTP – analiza użyteczności systemu pod kątem wrażeń użytkowych, zbiór badań nad wydajnością systemu (z uwzględnieniem różnych mechanizmów propagacji danych) i propozycji na jej dalsze optymalizację.

Rozdział 5. zawiera krótkie podsumowanie, podkreślające osiągnięte cele i potencjalne kierunki dalszego rozwoju projektu.

## 1.2. Cele pracy

Celem tej pracy jest stworzenie protokołu i oprogramowania pozwalającego na rozproszony i bezpieczny dostęp do współdzielonych zasobów, w warstwie użytkowej odpowiadającego obecnemu Hypertext Transfer Protocol (HTTP). W toku pracy, projekt określany będzie skrótem **dHTTP** – distributed HTTP.

Głównym założeniem projektu jest odciążenie stron, które nagle stają się popularne – często opcją dla właścicieli takich stron jest inwestycja w szybsze łącza czy więcej sprzętu, wiążąca się

ze sporymi kosztami. Kosztami które mogą zresztą nie znaleźć uzasadnienia – dodatkowy sprzęt pomoże w chwilach szczytowego ruchu, będzie jednak przez większość czasu leżeć odłogiem, wciąż jednak generując koszty łącz i prądu.

Częściowym rozwiązaniem tego problemu są chmury i udostępniany przez nie *autoscaling* – użytkownicy, zamiast utrzymywać własną infrastrukturę, mogą nie tylko zdać się na serwery zarządzane przez zewnętrzny podmiot, ale także wykorzystać mechanizmy automatycznego rozszerzania i zmniejszania ilości urządzeń czy wykorzystywanego łącza, w zależności od obciążenia całego klastra. To pozwala na drastyczną minimalizację kosztów i jest rozwiązaniem coraz chętniej stosowanym także przez podmioty o gigantycznym ruchu sieciowym, jak serwis streamingowy Netflix ([1]).

Wciąż jednak są to rozwiązania obciążające właściciela serwera, co może stanowić problem w przypadku projektów hobbystycznych czy krajów rozwijających się – koszt autoscalingu może przecież okazać się nieproporcjonalnie wysoki w stosunku do czasów spokojnego ruchu.

System zaprezentowany w poniższej pracy – dHTTP – ma posłużyć jako samoskalująca się, rozproszona alternatywa do tego podejścia. Musi wziąć on pod uwagę optymalne rozłożenie danych, zabezpieczenie przed zmienianiem zawartości zasobów przez niepowołane podmioty (hashe i podpisy kryptograficzne), szybki i możliwie najmniej scentralizowany sposób współdzielenia informacji o stanie systemu (*kto jest online i może dać mi plik xyz z hosta example.com?*) i aktualizacjach zawartości oraz stronę kliencką, będącą w tym przypadku wtyczką do przeglądarki Google Chrome, nakładającą warstwę dHTTP na polecenia pobierania zasobów. Z punktu widzenia użytkownika system jest *przezroczysty*: zostanie węzłem i klientem sieci dHTTP będzie polegało wyłącznie na instalacji wtyczki, która działać będzie w tle.



## **2. Droga do rozproszenia**

([2], [3])

### **2.1. Zarys teoretyczny**

### **2.2. Historia i istniejące narzędzia**

### **2.3. Podstawa projektu**



## **3. Projektowanie i implementacja**

### **3.1. Wymagania funkcjonalne**

### **3.2. Definicje, architektura i technologie**

#### **3.2.1. Wykorzystane technologie i narzędzia**

#### **3.2.2. Problemy warstwy sieciowej**

#### **3.2.3. Decentralizacja czy rozproszenie?**

#### **3.2.4. Bezpieczeństwo**

##### **3.2.4.1. Czy autentykacja to nasza brożka?**

##### **3.2.4.2. Gdzie leżą granice zdrowych heurystyk?**

#### **3.2.5. Propagacja i przechowywanie danych**

### **3.3. Aplikacje**

#### **3.3.1. Węzeł**

#### **3.3.2. Klient**



## **4. Testy i optymalizacja**

### **4.1. Zarys użytkowości i działania systemu**

### **4.2. Wydajność systemu**

#### **4.2.1. Wydajność klienta**

#### **4.2.2. Wydajność serwera**

##### **4.2.2.1. Wydajność rozwiązań propagacji danych**

##### **4.2.2.2. Wydajność protokołów komunikacji między węzłami**

### **4.3. Podsumowanie rozwiązań optymalizacyjnych**



## **5. Podsumowanie**

### **5.1. Potencjalne kierunki rozwoju**





# Bibliografia

- [1] Amazon Web Services Inc. *Auto Scaling*. 2017. URL: <http://web.archive.org/web/20171127225615/https://aws.amazon.com/autoscaling/> (term. wiz. 2017-12-31).
- [2] A. Diller. *LaTeX wiersz po wierszu*. Gliwice: Wydawnictwo Helion, 2000.
- [3] L. Lamport. *LaTeX system przygotowywania dokumentów*. Kraków: Wydawnictwo Ariel, 1992.