

# Capacity Planning dla baz danych Oracle

*Beata Deptuła, Marcin Przepiórowski,  
Marcin Kwaśniński, Paweł Chomicz*

Altkom Akademia

Proces **Capacity Planning** (CP) to przewidywanie obciążeń środowiska zachodzących pod wpływem zmian wymagań biznesowych. Zmiany biznesowe mogą być związane ze zjawiskami takimi jak:

- <sup>1</sup> zmiany w implementacji aplikacji, np. dodanie nowej funkcjonalności,
- <sup>1</sup> duże zmiany rozmiarów bazy danych, np. połączenie dwóch firm,
- <sup>1</sup> zmiana charakteru obciążenia bazy danych.

Celem procesu CP jest udzielenie odpowiedzi na pytanie: czy po zmianach biznesowych nasze środowisko dalej będzie spełniało założone wymagania pod kątem czasu odpowiedzi lub innych parametrów. Jeżeli wymagania nie będą spełnione, należy dokonać analizy pozwalającej stwierdzić, jakich zmian należy dokonać w środowisku aby wymagania zostały spełnione.

W zależności od wymaganej dokładności prognoz i analiz, proces CP jest bardzo zmienny. Większa dokładność wiąże się z dłuższym czasem analizy, a więc i z większymi kosztami. Z kolei analizy o niskiej dokładności (z dużym błędem) nie zawsze dadzą poprawną odpowiedź na zadawane pytania.

Proces modelowania wzrostu systemów komputerowych powiązany jest z opisaniem zachowania się systemu informatycznego od strony użytkownika końcowego. Zmiana warunków pracy widziana od strony użytkownika końcowego, spowodowana jest zmianą warunków pracy systemu informatycznego. Powiązanie zaś tych dwóch punktów widzenia daje możliwość przewidywania wzrostu systemu informatycznego oraz określenie, czy dany system informatyczny będzie dalej spełniał wymagania klienta.

## 1. Capacity Planning – wprowadzenie

Proces Capacity Planning (CP) jest to przewidywanie zmian obciążenia środowiska zachodzących pod wpływem zmian wymagań biznesowych. Zmiany biznesowe mogą być związane:

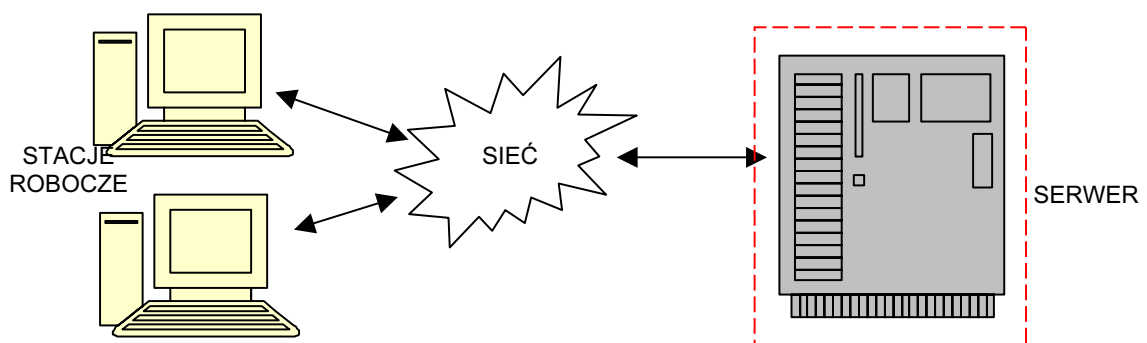
- z dużymi zmianami rozmiarów bazy danych – np. połączeniem się dwóch firm,
- ze zmianami charakteru obciążenia bazy danych,
- z dodaniem użytkowników.

Celem procesu CP jest udzielenie odpowiedzi czy po zmianach biznesowych nasze środowisko dalej będzie spełniało nasze wymagania pod kątem np. czasu odpowiedzi lub innych parametrów. Jeżeli wymagania nie będą spełnione należy dokonać analizy, pozwalającej stwierdzić jakich zmian należy dokonać w środowisku aby spełnienie ich nastąpiło.

W zależności od wymaganej dokładności prognoz i analiz proces CP jest bardzo zmienny. Większa dokładność wiąże się z dłuższym czasem analizy a więc i z większymi kosztami, a z kolei analizy o niskiej dokładności (z dużym błędem) nie zawsze dadzą poprawną odpowiedź na zadawane pytania.

Przed przystąpieniem do planowania wzrostu należy dokładnie określić zakres elementów środowiska, które będą podlegały monitorowaniu i planowaniu.

Przykładowe środowisko z podziałem na elementy z planowaniem wzrostu i bez planowania wzrostu.



Rys. 1. Przykładowe środowisko

W tym środowisku procesowi monitorowania i planowania wzrostu podlega tylko serwer. Reszta środowiska nie podlega planowaniu oraz monitorowaniu. Aby zapewnić pełną wydajność środowiska, należy zadbać o odpowiednią wydajność elementów nie monitorowanych lub też włączyć je do polityki planowania i monitorowania. Ma to szczególne znaczenie w przypadku planowania i monitorowania czasów odpowiedzi aplikacji. Czas odpowiedzi w danym środowisku wynosi:

$$T_r = T_s + T_n + T_w$$

Gdzie:

$T_r$  – czas odpowiedzi całego systemu

$T_s$  – czas odpowiedzi serwera, aktualny i planowany

$T_n$  – czas przesyłania danych przez sieć

$T_w$  – czas wyświetlenia danych na stacji klienckiej.

W powyższym środowisku monitorowany i planowany jest tylko czas odpowiedzi serwera, czas odpowiedzi sieci i stacji klienckiej nie jest monitorowany. Należy więc przyjąć, że przy odpowiedniej wydajności sieci i stacji klienckiej jest on stały i nie zależy od liczby transakcji biznesowych w systemie. W związku z tym czas odpowiedzi aplikacji zależy tylko od czasu odpowiedzi serwera.

Jeżeli sieć i stacja kliencka nie są dostatecznie wydajne, to pomiary i przewidywanie czasu odpowiedzi na serwerze oraz czasu odpowiedzi całej aplikacji będą obciążone dużym błędem i nie spełnią wymogów związanych z planowaniem wzrostu środowiska i określeniem czasu odpowiedzi czy określonym obciążeniu.

## 2. Metody stosowane w procesie CP

### 2.1. Zasada mnożnika

Czas wykonanie zwiększa się wprost proporcjonalnie do ilości danych.

*Przykład:*

Cztery zadania wsadowe wykonują w nocy ładowanie danych do bazy. Czas potrzebny do zakończenia działania do 4 godziny. Liczba danych zwiększyła się o 50%. O ile wydłuży się czas ładowania danych do bazy?

Za pomocą prostego mnożenia można by oczekiwać że czas również zwiększy się o 50%, a zatem z 4 do 6 godzin. Jednak ta metoda nie przewiduje możliwości kolejkowania się procesów, braku wydajności I/O oraz wielu innych rzeczy. Oczywiście, może się okazać że czas zwiększy się o 50% ale jest to raczej przypadek niż reguła. Metoda ta nie powinna być stosowana przy rzeczywistym i profesjonalnym planowaniu przyrostu.

### 2.2. Analiza trendów

Analiza trendów może być wykorzystywana w procesie CP, jednak nie do wszystkich parametrów. Jedynie parametry, których wzrost jest bliski liniowemu mogą być określane w ten sposób z dość dobrym prawdopodobieństwem. Źródłem danych do analizy trendów powinien być dość szeroki (np. roczny) okres zbierania danych. Na podstawie tych danych, odpowiednio obrobionych można przewidywać wartości parametrów w przyszłości.

### 2.3. Analiza za pomocą regresji

Analiza wykonywana za pomocą regresji może być stosowana dla tego samego zestawu parametrów co analiza trendów. Parametry zmieniające się w sposób nieliniowy, analizowane za pomocą metody regresji mogą dawać błędne wyniki. Metoda regresji działa na zasadzie określania związków pomiędzy parami wartości, a następnie na wykreślaniu linii według której zmieniają się parametry. np.

$$y = m * x + c$$

gdzie:

- x – liczba transakcji
- m – współczynnik określający zależność pomiędzy transakcją a utylizacją CPU
- c – współczynnik pomocniczy
- y – utylizacja CPU w zależności od ilości transakcji.

## 2.4. Symulacja

Obserwowanie wzrostu obciążenia środowiska za podstawie symulacji może dostarczyć dość dokładne wyniki. Dużym minusem tej metody jest konieczność zbudowania modelu całego środowiska, co jest skomplikowane i czasochłonne, a następnie wykonanie symulacji na zadanych zestawach danych. Symulowanie całego środowiska wymaga dużej mocy obliczeniowej dla uzyskania dobrych i wiarygodnych wyników.

## 2.5. Modelowanie

Modelowanie polega na stworzeniu uproszczonego modelu środowiska bazodanowego, na następnie szacowania na jego podstawie zmian zachodzących w środowisku pod wpływem zmian biznesowych. Metoda modelowania daje dość dobre wyniki, nie posiadając wad związanych z symulacją środowiska. Istnieje kilka metod modelowania systemów:

- model kolejkowy M/M/n
- systemy kolejkowe
- modelowanie za pomocą współczynników – 'Ratio Base Modelling' – opracowane przez specjalistów firmy Oracle.

Tabela 1. Porównanie metod przewidywania

Przewidywana wielkość	Mnożnik	Trendy/Regresja	Symulacja	Modelowanie
<b>Utylizacja CPU</b>	Możliwe	Tak	Tak	Tak
Czas odpowiedzi	Nie	Nie	Tak	Tak
Zmiana obciążenia	Nie	Nie	Tak	Tak
Zmiana środowiska	Nie	Nie	Możliwe	Tak

## 3. Przygotowanie do CP

Pierwszym krokiem przed przystąpieniem do procesu Capacity Planningu jest określenie wymagań klienta co do spodziewanych wyników analizy. Na tej podstawie należy wybrać metodę przeprowadzania procesu planowania rozwoju środowiska.

Kolejnym etapem jest przeprowadzenie procesu optymalizacji działania istniejącego środowiska bazodanowego, polegające na strojeniu parametrów pracy serwera jak również na określeniu poprawności projektu bazy danych pod kątem kodu SQL-a oraz implementacji struktur pomocniczych (indeksy). Pominięcie tego etapu może wprowadzać duże błędy do procesu planowania. Np. brak indeksu na tabeli która ma kilka wierszy nie wpłynie na wydajność pracy bazy, jeżeli zwiększymy liczbę sesji, jeżeli natomiast tabela zwiększy się z kilku do kilku tysięcy wierszy, brak indeksu wpłynie znacząco na wydajność bazy. Tak więc etap ten może być pominięty tylko w sytuacji

gdy klient posiada zoptymalizowaną bazę danych. W przeciwnym przypadku należy ograniczyć się do znalezienia sesji oraz poleceń najbardziej obciążających bazę.

Po zakończeniu procesu strojenia bazy danych należy określić zestaw danych, które będą zbierane na potrzeby procesu planowania. Należy również określić okres zbierania danych oraz okres próbkowania systemu. Okres zbierania danych powinien być dostatecznie długi aby wartości średnie parametrów mogły być dobrze oszacowane. Próbkowanie systemu nie powinno być wykonywane za często z dwóch powodów:

- próbkowanie dodatkowo obciąża serwer
- zbyt duża liczba danych z krótkiego okresu czasu, może dawać złe wartości średnie.

### 3.1. Charakterystyka obciążenia

Jednym z najważniejszych elementów w procesie planowania rozwoju środowiska jest poprawne i jasne zdefiniowanie obciążenia w danym środowisku. Definiowanie obciążenia zależy od wymagań jakie powinny być osiągnięte w metodzie planowania. W przypadku planowania środowisk bazodanowych można analizować obciążenie z dwóch punktów widzenia:

- systemu operacyjnego – analizując ilość i zajętość zasobów przez procesy związane z działaniem bazy danych
- bazy danych – analizując ilość i statystyki sesji, które wykonują dane zadania w bazie danych.

Najlepsze wyniki uzyskuje się łącząc obie metody, co daje pełny obraz działania środowiska. Metoda mieszana jest nieczuła na błędne określanie czasu użycia procesora przez niektóre wersje baz danych Oracle oraz pozwala uwzględnić obciążenie, którego serwer Oracle nie jest w stanie wychwycić.

Zebrane dane, dotyczące obciążenia systemu, należy obrobić za pomocą metod matematycznych, np. odrzucając nie powtarzające się bardzo wysokie lub bardzo niskie odczyty, obliczając średnią wartość w poszczególnych przedziałach godzinowych na podstawie danych z dłuższego okresu, określić trendy w charakterystyce obciążenia.

Oprócz definiowania klas obciążenia należy również określić rozłożenie obciążenia w czasie, pozwalające stwierdzić jaki jest jego charakter. Na tej podstawie należy określić okres oraz częstotliwość zbierania danych ze środowiska. Jeżeli obciążenie jest równomiernie rozłożone, okres zbierania danych powinien być długi a pomiary stosunkowo rzadkie (np. okres 24 godzin pomiar co ½ godziny). Przy nierównomiernym rozłożeniu obciążenia należy próbować system w okresie największego obciążenia (np. okres 2 godzin – pomiary co 5 minut). Uśrednianie danych przy dużych zmianach dobowego obciążenia może powodować, że w chwilach maksymalnego obciążenia system obliczonych dla danych uśrednionych byłby niewydajny.

## 4. Modelowanie

### 4.1. Metoda współczynników

Metoda ta została opracowana przez specjalistów firmy Orapub w roku 1996. Dokładny opis tej metody można znaleźć na stronach [www.orapub.com](http://www.orapub.com). Za jej pomocą możemy określić zależności pomiędzy wykonywanymi klasami zadań a danym zasobem systemowym, np. zależność pomiędzy zadaniami wsadowymi a obciążeniem procesora.

Podstawowy model zastosowany w tej technice jest prosty i wygląda w następujący sposób

$$S = C_1 / R_1 + \dots + C_n / R_n \ +/- K$$

Zmienna S jest szacowaną (obliczaną) wartością użycia danego zasobu systemowego, C jest liczbą definiującą obciążenie (workload) w danej kategorii, R jest współczynnikiem proporcjonalności a K jest błędem metody.

Metoda współczynników jest używana przez jej twórców do następujących celów:

- planowania zakupów sprzętu
- określania alternatywnych architektur dla środowiska
- size'owania środowiska
- planowania rozwoju systemów produkcyjnych.

Przykład zastosowania metody współczynników

Szczytowa ilość sesji OLTP	575 - C1
Szczytowa ilość sesji wsadowych	6 - C2
Zakładany błąd	0
Wartość szukana	liczba procesorów ?

Przykładowy model:

$$S = C_1 / R_1 + \dots + C_n / R_n \ +/- K$$

Zbierając dane z istniejącego środowiska, szukana wartość S będzie równa liczbie procesorów N pomnożonej przez średnią użycie procesorów U.

W naszym przykładzie, model wygląda następująco:

$$N * U = 575 / R_1 + 6 / R_2$$

Występują w nim 4 niewiadome:

- R1 – współczynnik dla transakcji OLTP - oltp\_to\_cpu
- R2 – współczynnik dla zadań wsadowych – batch\_to\_cpu
- N – liczba procesorów
- U – użycie procesorów

Współczynniki będą obliczane na podstawie danych zbieranych z już istniejącego środowiska, dla odpowiednio zdefiniowanego obciążenia. Aby uniknąć błędów przy zbieraniu danych należy dobrze poznać aplikację oraz charakterystykę obciążenia w środowisku produkcyjnym. W przykładzie występują dwa rodzaje obciążenia: OLTP oraz wsadowe. Jeżeli zadania wsadowe są wykonywane w okresie, kiedy występuje znikoma liczba zadań OLTP, to można uprościć model, do następującej postaci:

$$N * U = 6 / R_2$$

Zbierając dane w tym okresie, możemy w prosty sposób wyliczyć współczynnik R2. Przykładowo:

Liczba zadań wsadowych	6 - C2
Liczba procesorów	12
Średnie obciążenie (użycie)	45%

$$R_2 = 6 / (12 * 0.45) = 1.11$$

Po podstawieniu do modelu wykorzystywanego w przykładzie, uzyskujemy:

$$N * U = C_1 / R_1 + C_2 / 1.11$$

Następnym krokiem jest zebranie danych przy pewnej liczbie sesji OLTP oraz sesji batchowych. Na podstawie tych danych obliczony zostanie współczynnik  $R_1$ .

Przykładowo:

Liczba sesji OLTP                    200 –  $C_1$

Liczba zadań wsadowych            5 –  $C_2$

Liczba procesorów                    12

Średnie obciążenie (użytkowanie) 60%

$$12 * 65\% = 200 / R_1 + 5 / 1.11$$

Na podstawie tego modelu, wyliczamy współczynnik  $R_1$

$$R_1 = 200 / (12 * 0.65 - 5/1.11) = 60.60$$

Współczynnik ten oznacza, że procesor wykorzystany w 100 % jest w stanie obsłużyć 60.60 sesji OLTP.

Wykorzystując obliczone współczynniki i wstawiając je do szacowanego modelu, otrzymujemy następujące wyrażenie:

$$S = 575 / 60.61 + 6 / 1.11 = 14.90$$

Wyliczona wartość  $S$  oznacza, że 14.90 procesorów wykorzystywanych w 100% jest w stanie obsłużyć zakładaną liczbę sesji OLTP oraz zadań wsadowych.

Proces obliczania współczynników nie powinien zakończyć się w tym miejscu, proces ten należy ponawiać dla różnych parametrów wejściowych, tak aby uniknąć błędnej interpretacji współczynnika.

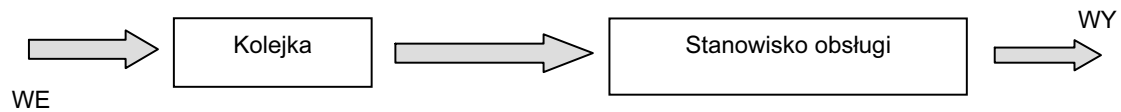
Poniższa lista przedstawia przykładową obróbkę danych, z której korzystają autorzy metody współczynników:

- usunięcie danych dla zadań wsadowych, gdzie liczba zadań było mniejsza niż 3
- usunięcie danych oddalonych od klastra lub linii trendu. Jeżeli występuje duża liczba danych oddalonych od linii trendu, należy je dołączyć do analizy oraz określić dlaczego one wstępują
- usunięcie danych zbieranych podczas nienormalnego obciążenia środowiska, np. podczas backupu.

#### 4.2. Model kolejkowy M/M/n

Początki analizy za pomocą modeli kolejek wywodzą się z roku 1909. Jedne z pierwszych badań przeprowadził A.K. Erlang, badając ruch w centralach telefonicznych. Już na samym początku rozwoju systemów informatycznych okazało się, że można je dobrze opisywać za pomocą pro-

stych modeli kolejkowych. Za pomocą tej metody można modelować systemy komputerowe, sieci komputerowe o dowolnej topologii, jak również inne sieci telekomunikacyjne.



Rys. 2. Przykładowe stanowisko M/M/1

WE	proces wejścia: opis strumienia klientów przybywających w czasie. Np. liczba użytkowników systemu, liczba sesji w systemie
WY	proces wyjścia: opis strumienia wychodzących w czasie. Np. zakończone procesy użytkowników, średni czas odpowiedzi systemu
Kolejka	rozkład czasu czekania w kolejce do rozpoczęcia obsługi,
Stanowisko obsługi	jest to proces wykonujący pewne czynności na danych. Każdy klient wchodzący do systemu, musi być obsłużony w stanowisku obsługi. Jego parametrem jest średni czas obsługi klienta na stanowisku.

Jednokanałowy system obsługi nosi nazwę systemu kolejkowego M/M/1. Jeżeli w modelu występuje więcej stanowisk obsługi (n), to model ten nosi nazwę M/M/n.

Przykładem systemu kolejkowego może być bar szybkiej obsługi.

Klienci – osoby wchodzące do baru zjeść obiad

Kolejka – osoby, które weszły do baru przed nami i nie zostały jeszcze obsłużone

Stanowisko obsługi – lada, przy której przyjmowane są zamówienia i wydawane są posiłki

Czas przebywania w kolejce można zdefiniować jako czas liczony od wejścia do baru do momentu podejścia do lady. Czas obsługi klienta jest to czas od podejścia klienta do lady do wydania mu posiłku i jego odejścia od lady. Przepustowość – liczba klientów obsłużonych w jednostce czasu. Czas odpowiedzi jest to suma czasu przebywania w kolejce oraz czasu obsługi, czyli czas od wejścia do baru do momentu zakończenia obsługi przy ladzie.

W modelowaniu środowiska komputerowego, jako stanowisko obsługi można podstawić dowolny zasób, np. procesor czy dysk. Strumieniem wejściowym może być np. liczba użytkowników w systemie.

Aby jednoznacznie opisać model kolejkowy, potrzebne są następujące parametry:

- liczba transakcji na wejściu w jednostce czasu, np. 5 trx/s
- Czas obsługi jednej transakcji, np. czas procesora potrzebny do wykonania pewnego zadania.

Najciekawszą możliwością modelowania za pomocą kolejek jest możliwość oszacowania zmian czasu odpowiedzi środowiska. Oczywiście czas odpowiedzi uzyskany z prostego modelu nie jest czasem odpowiedzi widzianym przez użytkownika (np. skutek opóźnień związanych z siecią), ale jest on proporcjonalny do rzeczywistego czasu odpowiedzi systemu. Generalnie, czas odpowiedzi widziany przez użytkownika powinien być podzielony na dwie składowe:

- czas odpowiedzi badanego systemu –  $T_s$
- czas przesłania informacji, oraz jej wyświetlenia na ekranie –  $T_r$

Czas  $T_r$  nie zależy od wydajności systemu komputerowego, a jedynie od wydajności warstw pośrednich, sieci oraz stacji klienckiej. Bardziej interesującym jest czas  $T_s$ , którego wartość zależy od wydajności systemu. Czas ten może być powiązany proporcjonalnie z czasem odpowiedzi wygenerowanym z modelu. W ten sposób można szacować do jakiej utylizacji danego zasobu, czas odpowiedzi systemu jest jeszcze akceptowalny dla środowiska bazodanowego.

W rzeczywistości najczęściej spotyka się dwie architektury:

- dwuwarstwowa (klient-serwer),
- trójwarstwowa (baza – serwer aplikacji – klient).

W obu przypadkach interesować nas będzie czas odpowiedzi bazy danych, natomiast całkowity czas odpowiedzi będzie składał się z różnych składników.

W architekturze dwuwarstwowej całkowity czas odpowiedzi można podzielić na dwie składowe – czas odpowiedzi bazy danych i czas potrzebny do przesłania i wyświetlenia informacji na stacji klienckiej. W architekturze trójwarstwowej całkowity czas odpowiedzi składa się z trzech składników: czas odpowiedzi bazy, czas odpowiedzi serwera aplikacji oraz czas przesłania danych do klienta. W przypadku analizy czasu odpowiedzi bazy danych, czas odpowiedzi serwera aplikacji oraz czas przesyłania danych do klienta można zsumować. W takim przypadku traktujemy środowisko trójwarstwowe jako dwuwarstwowe, gdzie serwerem jest baza danych a klientem serwer aplikacji i reszta infrastruktury.

Przykład:

Czas obserwacji	1 min
Utylizacja 1 procesora w systemie wynosi	80%
Liczba sesji wynosi	50

Maksymalny akceptowalny czas odpowiedzi serwera wynosi 0,15 min

Jaki jest czas odpowiedzi systemu w chwili obecnej a jaki będzie po dodaniu jeszcze jednego procesora ?

Średni czas obsługi na stanowisku dla jednej sesji wynosi:

$$T_o = 1 \text{ min} * 80 \% / 50 = 0,8/50 = 0,016 \text{ min}$$

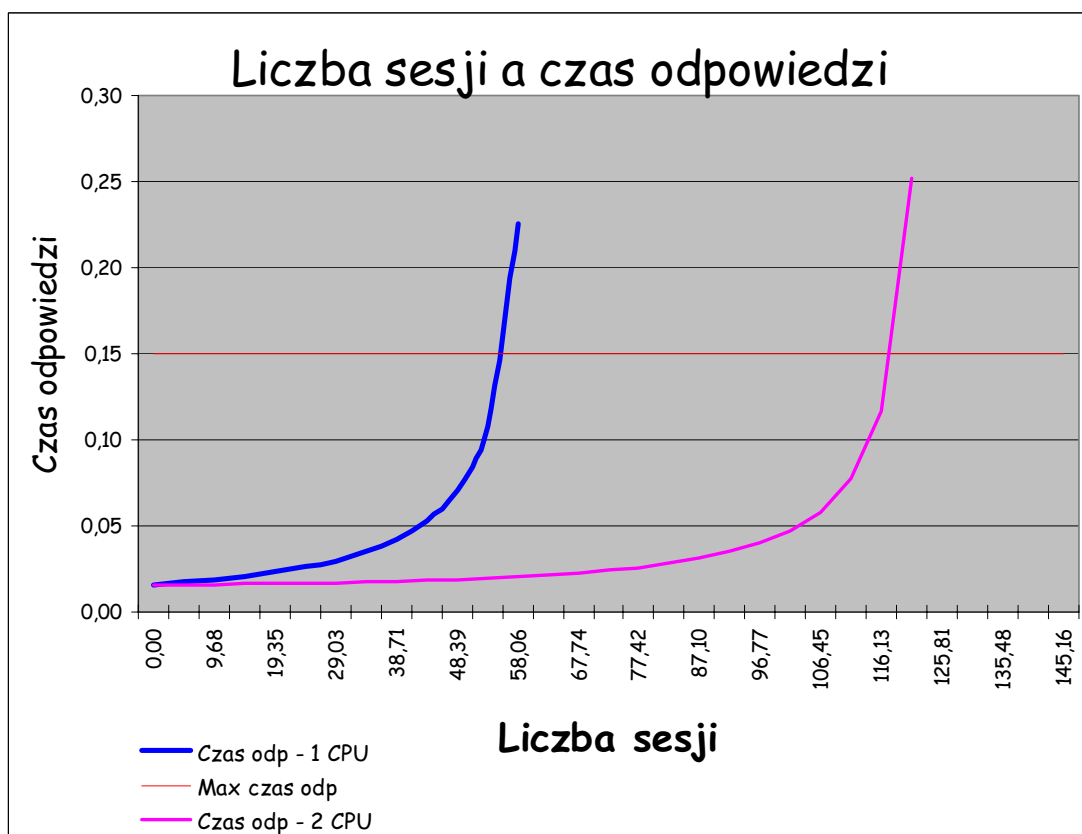
Zakładamy że czas obsługi na stanowisku nie zależy od liczby sesji.

Czas obsługi jest to czas zajęty przez 1 sesję w czasie obserwacji. Można więc założyć dla uproszczenia obliczeń, że w tym czasie wykonała jedną transakcję biznesową. Jeżeli w rzeczywistości wykonałaby więcej transakcji biznesowych, to należałoby podzielić czas obsługi przez liczbę wykonanych transakcji biznesowych, a następnie przy parametrze wejściowym opisującym liczbę transakcji na minutę (tu równym liczbie sesji) wykonać mnożenie.

Liczba transakcji na wejściu – 50 tr/min

Do obliczeń został wykorzystany arkusz Excel-a napisany przez Craiga A. Shallahamera, prezesa OraPub, Inc.

Description	Case 1 Values	Case 2 Values	Unit	Variable
Queues in system	1,000	1,000	#	QpS
Servers per queue	1,000	2,000	#	M
Service time	0,016	0,016	min/trx	Ts
System Arrival rate	50,000	50,000	trx/min	sys_lambda
Response time tolerance	0,150		min	Rt
Server Arrival rate	50,000	50,000	trx/min	Lambda
Traffic intensity	0,800	0,800	#	TI
Utilization	0,800	0,400	#	Util
Queue time	0,064	0,003	min	Tw
Queue length	3,200	0,152	#	Lw
Response time expected	0,080	0,019	min	Tq
Resp time + 1 stddev of Q time	0,158	0,028	min	$Tq + 1 * StdDev Tw$
Resp time + 2 stddev of Q time	0,237	0,036	min	$Tq + 2 * StdDev Tw$
Number in system	4,000	0,952	#	Lq



Rys. nr 3. Wykres czasów odpowiedzi w zależności od liczby sesji

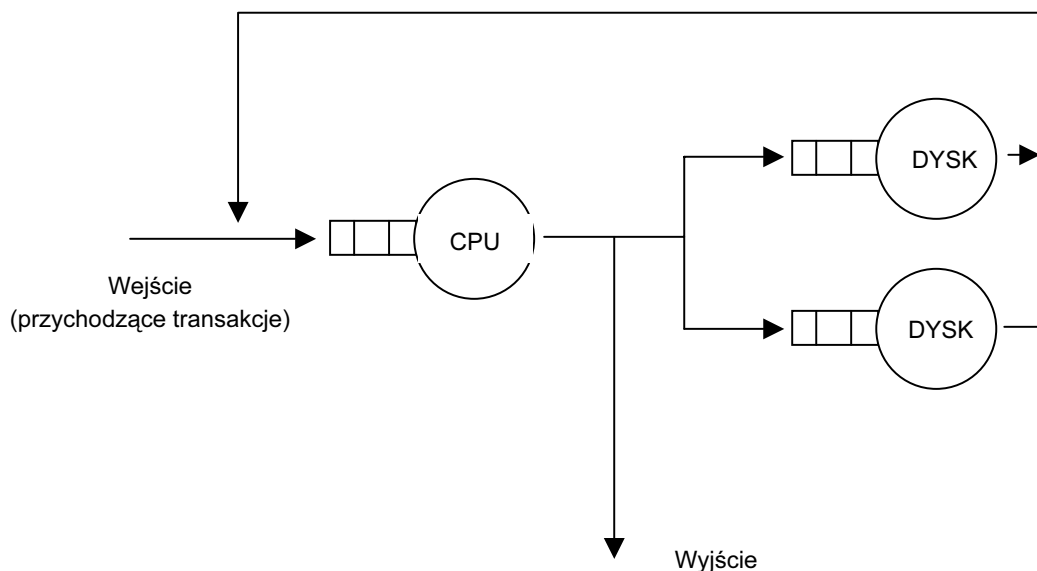
W chwili obecnej czas odpowiedzi systemu można oszacować na poziomie 0,08 min.

Założony maksymalny czas odpowiedzi wynosi 0,15 min i zostanie osiągnięty przy użyciu pomiędzy 86 a 92%. Użytkownik taki odpowiada ilości sesji pomiędzy 54 a 57. Oznacza to, że sys-

tem pracuje na granicy swojej wydajności, ponieważ w chwili obecnej pracuje tam 50 sesji, a więc po dodaniu kolejnych 5 sesji system przestanie spełniać założenia. Po dodaniu drugiego procesora czas odpowiedzi systemu ulegnie skróceniu i będzie wynosił około 0,019 min przy aktualnym obciążeniu (50 sesji). Maksymalny czas odpowiedzi w przypadku środowiska dwuprocesorowego zostanie osiągnięty przy ilości sesji pomiędzy 112 a 120, co oznacza że w chwili obecnej można podwoić liczbę sesji w systemie.

### 4.3. System kolejkowy

W metodzie systemów kolejkowych całe środowisko (system komputerowy) przedstawione jest jako sieć kolejek, które można rozwiązywać metodami analitycznymi. Sieć kolejek jest to zbiór stanowisk obsługi (omówionych w poprzednim punkcie) reprezentujących różne zasoby (np. CPU, dyski, sieć) oraz zbiór klientów reprezentujących użytkowników systemu lub wykonywane przez nich transakcje. Sieci kolejkowe mogą być rozwiązywane różnymi metodami analitycznymi, m.in. metodą asymptot oraz metodą MVA (Mean Value Analyzist).



Systemy kolejkowe mogą być wykorzystywane do modelowania dowolnego rodzaju obciążenia. Występują dwa rodzaje systemów kolejkowych:

- otwarte – gdzie określana jest liczba transakcji wejściowych, a każda obsłużona transakcja opuszcza system
- zamknięte – gdzie określana jest liczba klientów oraz ich czas opóźnienia ('think time'). Każda obsłużona transakcja, po czasie opóźnienia wraca do kolejki wejściowej.

Aby dobrze rozumieć metody rozwiązywania modeli kolejkowych należy zapoznać się z występującymi w nich wielkościami:

czas obserwacji	T
liczba transakcji	A
liczba zakończonych transakcji	C
przepustowość	$X = C/T$ – liczba zakończonych transakcji w czasie obserwacji
czas zajętości zasobu	B

użytkownicy  $U = B/T$

czas obsługi  $S = B/C$

Na podstawie praw opisujących zachowania systemu kolejkowego, można rozwiązać analitycznie całe środowisko. Jako wyniki otrzymuje się użycie poszczególnych centrów obsługi, czas odpowiedzi systemu, jak również średnie czas przebywania zadań w kolejce.

Przykład:

Model składa się z dysku oraz procesora, wykonywane są w nim zadania należące do dwóch klas obciążenia

Dane wejściowe:

Liczba zadań/s klasy 1 = zmienna w zakresie od 0 do 1,5

Liczba zadań/s klasy 2 = zmienna w zakresie od 0 do 2 z krokiem 0,5

Średni czas obsługi procesora dla klasy 1 = 0,01 s

Średni czas obsługi procesora dla klasy 2 = 0,1 s

Średni czas obsługi dysku dla klasy 1 = 0,02 s

Średni czas obsługi dysku dla klasy 2 = 0,02 s

Liczba wizyt na procesorze dla klasy 1 – 40

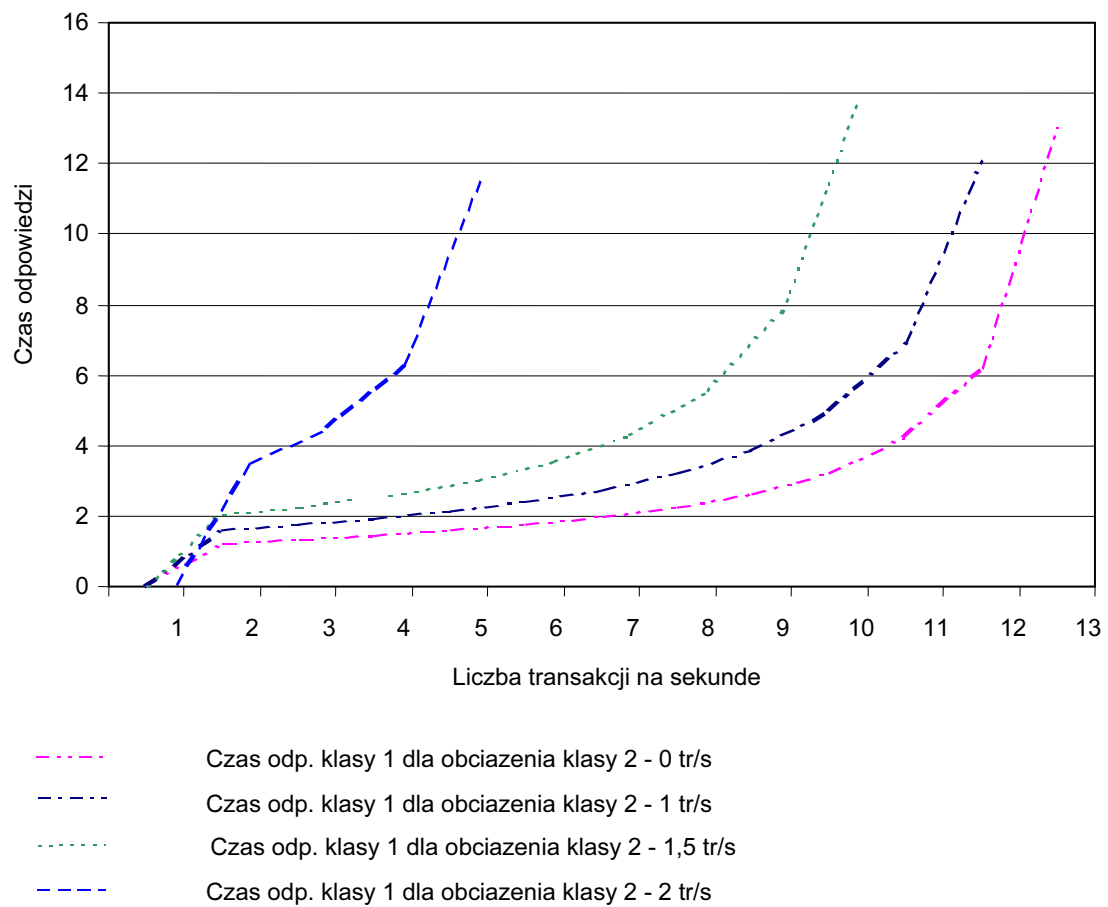
Liczba wizyt na procesorze dla klasy 2 – 4

Liczba wizyt na dysku (i/o request) dla klasy 1 – 39

Liczba wizyt na dysku (i/o request) dla klasy 2 – 3

Na podstawie danych wejściowych obliczono czas odpowiedzi systemu oraz jego zmiany.

Jak można zaobserwować na Rys. 4, czas odpowiedzi zadań należących do grupy 1 jest silnie zależy od liczby zadań grupy 2. Na tej podstawie można szacować zależności pomiędzy poszczególnymi grupami zadań oraz określać, jak zmieni się czas odpowiedzi w zależności od zmiany parametrów zewnętrznych. Oprócz tego można również szacować obciążenie poszczególnych centrów obsługi – czyli poszczególnych zasobów systemu.



Rys. 4. Wykres czasów odpowiedzi systemu