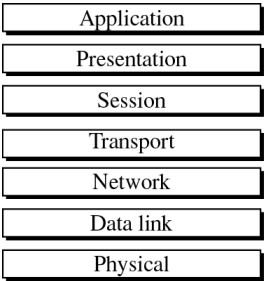


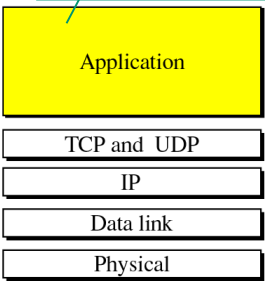
Горни слоеве:  
*сесиен,  
представителен  
и приложен*

OSI срещу TCP/IP

В TCP/IP, сесийният и представителният слоеве са абсорбирани от приложния слой.



OSI model

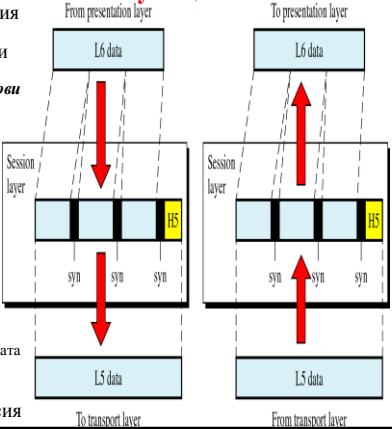


TCP/IP protocol suite

*Сесиен слой*

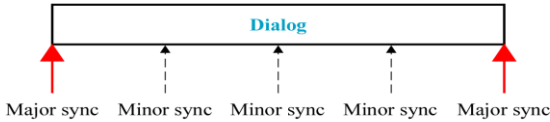
Сесиен слой: Функции

- Установяване на сесия между двете комуникаращи страни
- Обмен на данни:
  - С помощта на *диалогови единици (dialog units)*
  - Координиране на сесията (кой изпраща и кога)
  - Установяване на *синхронизационни точки* по време на диалога
    - В случай на комуникационни грешки над допустимия праг, *прекъсване* (преустановяване) на диалога с *подновяване* по-късно от последната договорена синхронизационна точка
- Прекратяване на сесия

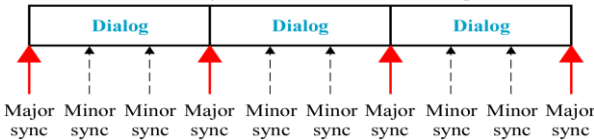


Сесиен слой: Синхронизационни точки

An activity made of only one dialog



An activity made of more than one dialog



- Разделяне обмена на данни на серии от диалогови единици
- Всяка **главна** синхронизационна точка трябва да бъде **потвърдена**
- **Второстепенните** синхронизационни точки **не трябва** да се **потвърждават**

*Сесийни  
протоколи:  
SIP и SDP*

## Session Initiation Protocol (SIP)

- Сесиен протокол, реализиран в приложния слой на TCP/IP модела.
- Създаване, управление и прекратяване (в реално време) на IP-базирани сесии
- Поддръжка на мултимедийни сесии
  - VoIP, телеконференции, видеоконференции.
- Използване на транспортните услуги на:
  - RTP/UDP (за ефективност)
  - TCP или SCTP (ако са необходими по-надеждни услуги)
  - TLS (ако са необходими услуги със сигурност)
- Текстово базиран протокол, по модела на HTTP.
  - Клиентската заявка извиква определен **метод** на сървъра и получава най-малко 1 **отговор** от него
- Включено използване на SDP за дефиниране съдържанието на сесия (RFC 2327)
  - SDP-кодираното тяло на SIP съобщенията съдържа информация за това, какво медийно кодиране е използвано.
  - Използва се напор от типове, подобни на MIME типовете.
- RFC 3261, 5411

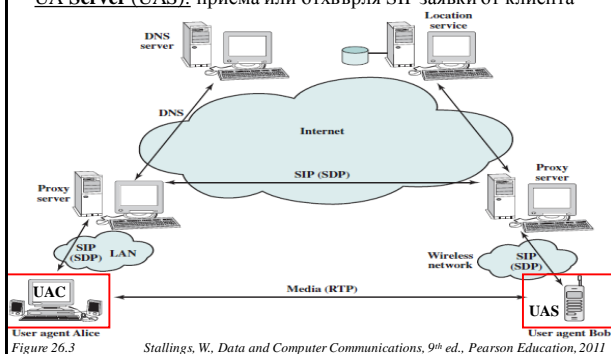
## SIP: Функции

- **Местоположение на потребителя** (т.е. текущ IP адрес)
  - Виканият потребител може да се мести от едно място на друго (т.е. да сменя мрежата и адреса си)
  - Виканият потребител може да има няколко устройства, всяко с различен IP адрес.
- **Наличност на потребителя**
  - Определяне готовността на викания за участие в комуникацията
- **Възможности на потребителя**
  - Поддържани медийни типове и формати
- **Настройка на сесия (setup)**
  - С договорени параметри
    - Между 2 страни (*point-to-point*)
    - Между няколко страни (конферендна сесия)
    - *Multicast* (един източник – много получатели)
- **Управление на сесия**
  - Поддръжка (и прекратяване на края)
  - Промяна на параметрите на сесията
    - Добавяне на нов медийн поток
    - Промяна на кодирането
    - Поканване на нови участници в сесията
- **Други**
  - Чакащо повикване, прехвърляне на повикване, шифроване, удостоверяване самоличността на участниците в сесията, ...

## SIP: Крайни компоненти

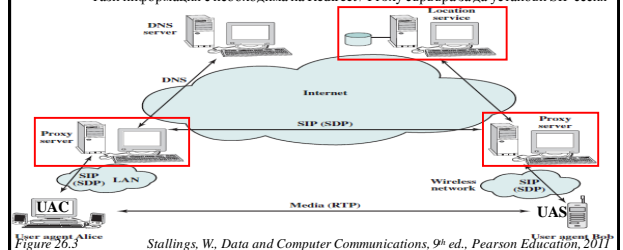
### Потребителски агенти (user agents, UAs)

- **UA Client (UAC):** издава SIP заявки и получава обратно отговори
- **UA Server (UAS):** приема или отхвърля SIP заявки от клиента



## SIP: Междинни компоненти

- **Proxy Server**
  - Действа като **сървър** и **клиент** (прави заявки от името на други клиенти)
  - Препраща заявки до друг обект, намиращ се по-близо до викания потребител.
  - Може да налага **политики** (например, проверка дали даденият потребител има право да извърши повикването)
  - Може да действа и като **пренасочващ сървър (redirect server)**, който (по време на създаване на сесия) определя адреса на викания и го връща на викания за да се свържи директно с него.
- **Registrar/Location Server**
  - Пазят информация за местоположението на потребителите (база данни от съпоставени SIP адреси и IP адреси)
  - Приема информация за текущия SIP адрес и асоциирания с него IP адрес, на който виканият желае да приема повиквания.
  - Тази информация е необходима на Redirect / Proxy сървъра за да установи SIP сесия



## SIP: URI

- **Uniform Resource Identifier**
- **Идентифициране на SIP ресурси**
  - Потребител на *on-line* услуга
  - Пощенска кутия в система за съобщения (*sip:ivan@ul.ie*)
  - IPv4/IPv6 адрес (*sip:ivan@123.45.67.89*)
  - Телефонен номер (*sip:ivan@123-456-7890*)
    - Web страница може да го съдържа
    - Позволява телефонно повикване чрез едно кликане върху бутон "Обади ми се" на личната web страница
  - Група в организацията
    - Например, отдел продажби, отдел услуги, ...
- **Обикновен URI:** *sip:user@domain*
  - Може да включва също и парола, номер на порт и други параметри
- **Сигурен URI:** *sips:user@domain*

## RE-INVITE – modify a session

## SIP: Методи

Method	Description	
INVITE	Request initiation of a session	Establish media session between UAs & confirm reliable message exchange
ACK	Confirm that a session has been initiated	by either party; session terminated when other party confirms.
BYE	Request termination of a session	Used during session establishment
OPTIONS	Query a host about its capabilities	(but does <u>not</u> undo completed call)
CANCEL	Cancel a pending request	User notifies server about his/her current SIP address and associated IP address on which s/he wishes to receive calls
REGISTER	Inform a redirection server about the user's current location	

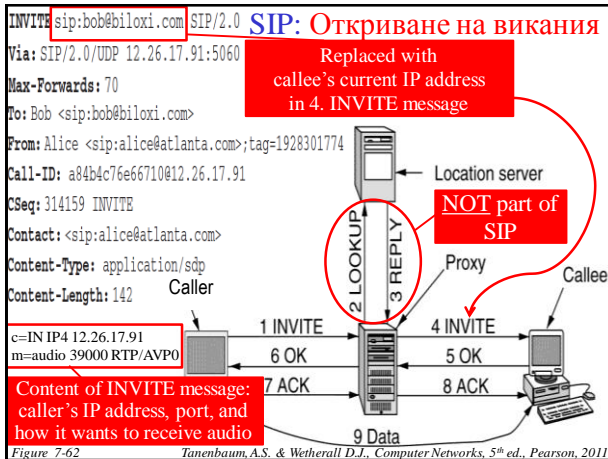
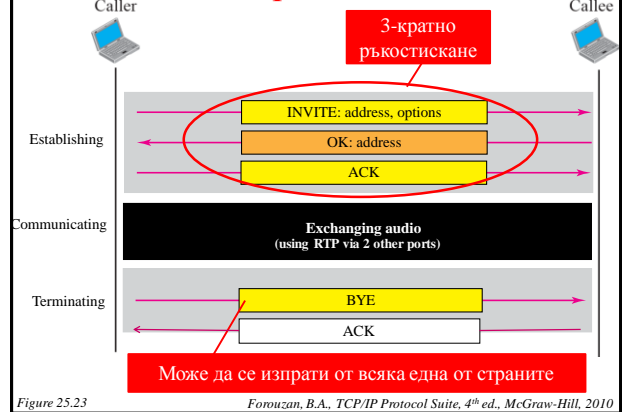
Figure 7-61

Tanenbaum, A.S. &amp; Wetherall D.J., Computer Networks, 5th ed., Pearson, 2011

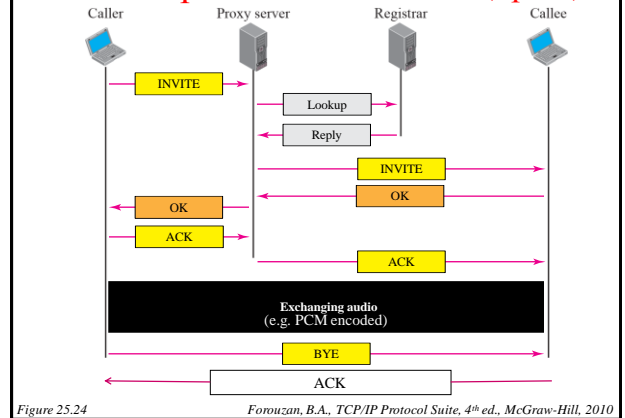
## SIP: Отговори

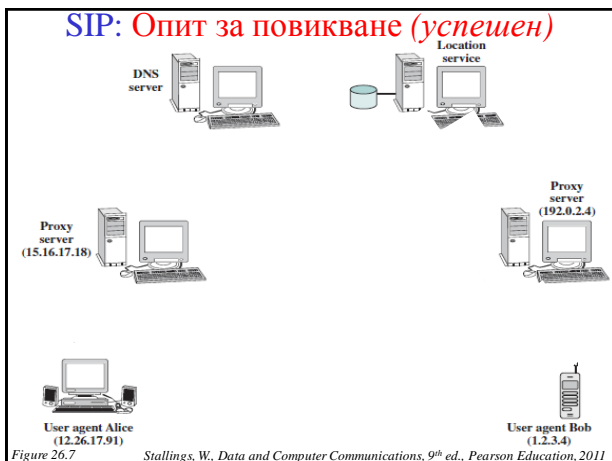
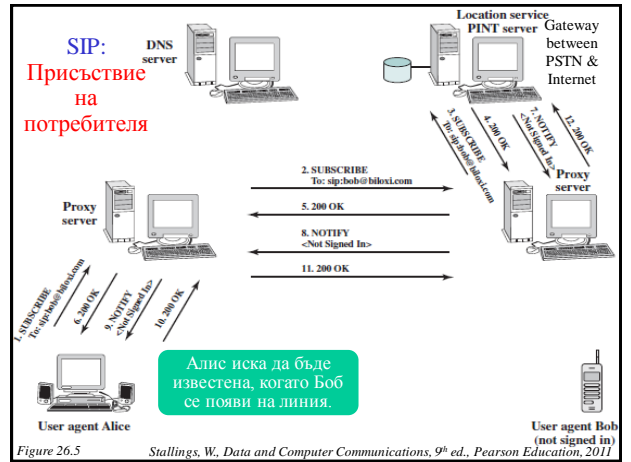
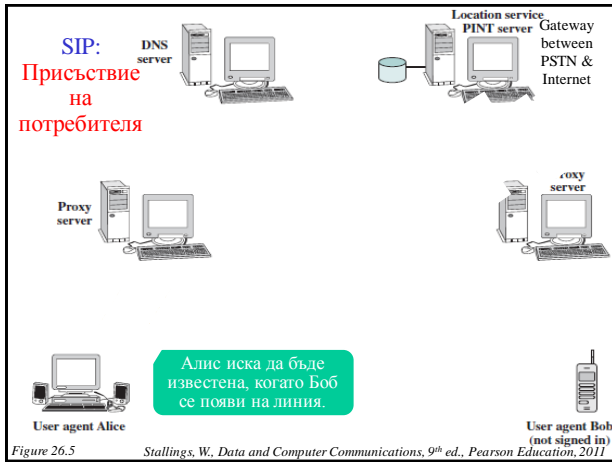
- **Временни (1xx)**
  - Заявката е получена и се обработва
- **Успех (2xx)**
  - Действието е успешно получено, разбрано и възприето.
- **Пренасочване (3xx)**
  - По-нататъшни действия са необходими за да се изпълни заявката
- **Клиентска грешка (4xx)**
  - Лош синтаксис на заявката или заявката не може да се изпълни от дадения сървър
- **Грешка в сървъра (5xx)**
  - Сървърът не е успял да изпълни очевидно валидна заявка
- **Глобален неуспех (6xx)**
  - Заявката не може да се изпълни на никой сървър

## SIP: Проста сесия



## SIP: Откриване на викания (прод.)



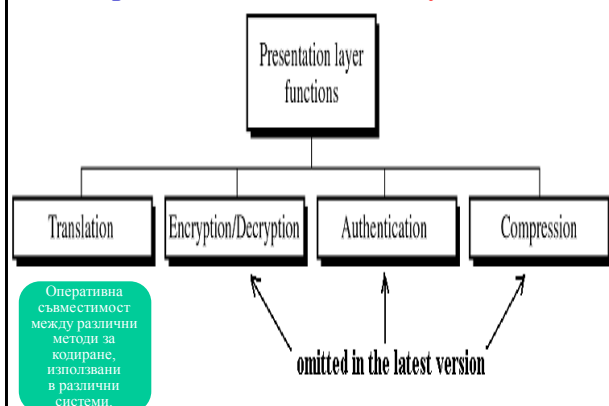


### Session Description Protocol (SDP)

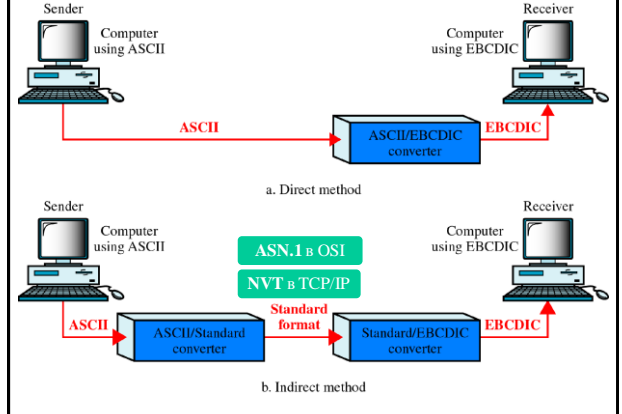
- Описва съдържанието на сесия
  - Интернет телефония (VoIP), интернет радио, мултимедия.
- Медийни потоци
  - Сесията може да включва множество потоци с различно съдържание
  - Аудио, видео, данни, контрол, приложение (подобно на MIME типове)
- Адреси на местоназначението/доставката
  - Може да бъде *multicast*
- Портове (на източника и местоназначението)
  - За всеки поток
- Типове полезен товар (*payload*)
  - Мултимедия формат за използване по време на сесията
- Начално и крайно време, и време за повтаряне на сесията.
  - За разпръсквателни сесии (*broadcast*), напр. излъчване на телевизионни и радио програми.
- Източник (*originator*)
  - За разпръсквателни сесии
  - При технически затруднения на получателя
- RFC 4566

## Представителен слой

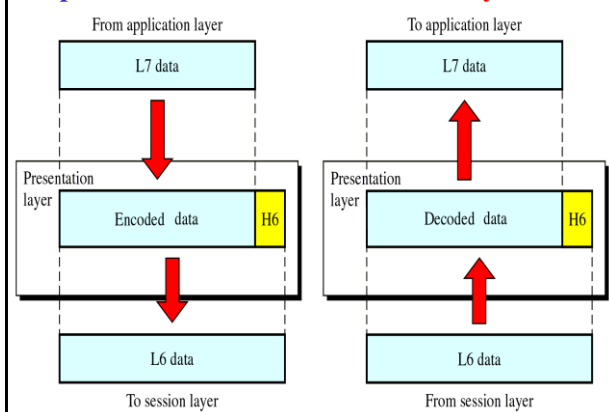
### Представителен слой: Функции



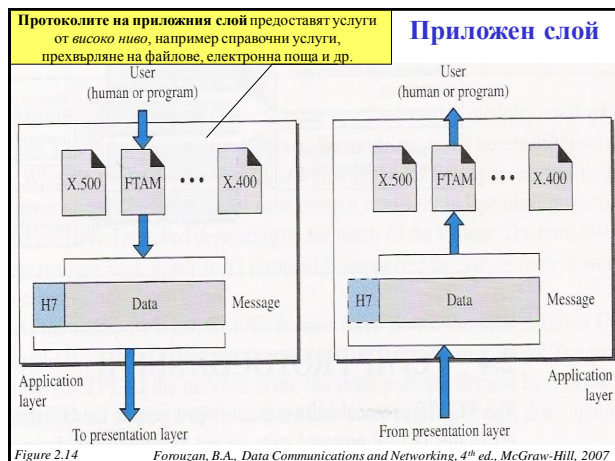
### Преки и косвени методи за транслиране



### Представителен слой: PDU капсулация



## Приложен слой



### Приложен слой: Протоколи

Usage	OSI	TCP/IP
Email	<b>X.400</b>	SMTP, POP, IMAP
File Transfer	<b>FTAM</b>	FTP
Virtual Terminal	<b>VT</b>	TELNET
Network Management	<b>CMIP</b>	SNMP
Directory Services	<b>X.500</b>	LDAP
Job Transfer and Manipulation	<b>JTM</b>	---
Manufacturing Messaging Service	<b>MMS</b>	---
Remote Database Access	<b>RDA</b>	---
Distributed Transaction Processing	<b>DTP</b>	---
Domain Name System	---	DNS
BOOTstrap	---	BOOTP
Dynamic Host Configuration	---	DHCP
HyperText Transport	---	HTTP

### Приложение и протокол на приложния слой

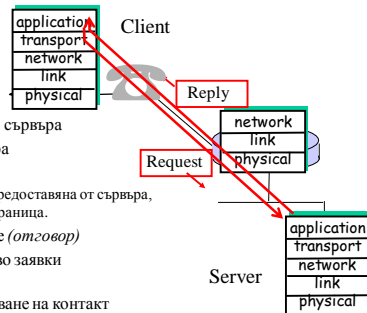
- **Приложение**
  - Комуникаращ (разпределен) процес
  - Работи в "потребителското пространство" на хостовете
  - Използва протокол на приложния слой за да осъществи контакт с други приложения
  - Трябва да взаимодейства с протоколния софтуер преди осъществяването на контакта
  - Пасивното приложение (*сървърът*) информира локалния протокол софтуер, че е готово да приема входящи съобщения.
  - Активното приложение (*клиентът*) използва протокола на приложния слой за осъществяване на контакт със сървъра
  - Двете приложения обменят съобщения чрез съответния протокол
  - Предоставяне на услуги на *потребителско ниво*
- **Протокол на приложния слой**
  - Дефинира съобщения за обмен между приложения и съответните действия
  - Използва услугите, предоставяни от протоколи на долния слой.

### Приложен слой: Модел 'клиент-сървър'

По този модел мрежовото приложение има 2 части: *клиент и сървър*

#### Клиент

- Инициира комуникация със сървъра
  - Изпраща заявка към сървъра
    - Възможно дори с данни
    - За използване на услуга, предоставяна от сървъра, например заявка за web-страница.
  - Изчаква обратно съобщение (*отговор*)
  - Може да изпраща множество заявки
- Сървърът** е 'слушател'
- Чака пасивно за осъществяване на контакт от страна на клиент
  - Приема заявки от клиенти
  - Предоставя услуги
  - Връща резултати
    - Например, изпраща обратно исканата web-страница.



### Клиент: Характеристики

- Приложна програма
  - Иска предоставяне на услуга от сървър
  - Също може да изпълнява и други допълнителни функции
- Извиква се директно от потребителя
- Изпълнява се локално на компютъра на потребителя
- Може да заяви множество услуги, но във всеки момент активно осъществява контакт само с 1 сървър.
- Не изисква специален хардуер или сложна операционна система (ОС)

### Сървър: Характеристики

- Привилегирована програма със специално предназначение, посветена на предоставянето на определена услуга.
- Извиква се автоматично при стартиране на системата и продължава да работи до изключването ѝ
- Изпълнява се на отдалечен компютър
  - Няколко различни сървъра могат да работят едновременно на един и същ (достатъчно мощен) компютър
  - Сървърите работят като независими процеси и могат да взаимодействат с множество клиенти едновременно
- Може да предлага услуги само на ограничен брой клиенти, които имат съответния достъп.
- Изисква мощен хардуер и специализирана операционна система

## Взаимодействие `клиент-сервър`

- Клиентът не е ограничен да осъществява достъп само до един единствен сервър за дадена услуга
  - Например, DNS клиент.
- Сервърът не е ограничен да изпълнява по-нататъшни взаимодействия тип `клиент-сервър`
  - Т.е. сервър на една услуга може да е клиент на друга услуга
    - Например, прокси (proxy) работещо като клиент и сервър.

## Клиентска архитектура

- По-проста от сервърната
- Повечето клиенти не могат да се справят с едновременни/паралелни взаимодействия с множество сервъри
- Клиентският софтуер обикновено се изпълнява като конвенционална програма
- Обикновено клиентите, за разлика от сервърите, не изискват специални привилегировани портове.
- Повечето клиенти разчитат на операционната система за осигуряване на сигурност

## Сервърни

### архитектури: Видове

- По вида на връзката
  - Със съединение
    - Надеждни, но им трябва ОС ресурси.
  - Без съединение
    - Трябват им по-малко ресурси, но приложението трябва да се бори само със загубата на съобщения.
- По състоянието на сервъра
  - Без запазване на информация за състоянието (stateless)
    - Всяка транзакция е независима
  - Със запазване на информация за състоянието (stateful)
    - Сервърът поддържа състояние
    - По-бързи, но по-скъпопоставящи за сервъра.
- По вида на обслужване на заявките
  - Паралелни (concurrent)
    - Могат да обслужват множество клиенти едновременно, без да се налага изчакване на края на обслужването на предишни заявки.
    - Създава се нов процес за всеки нов клиент
  - Итеративни (iterative)
    - Могат да приемат/обслужват само по 1 клиентска заявка във всеки един момент

## Итеративен сервър, неизползващ съединение

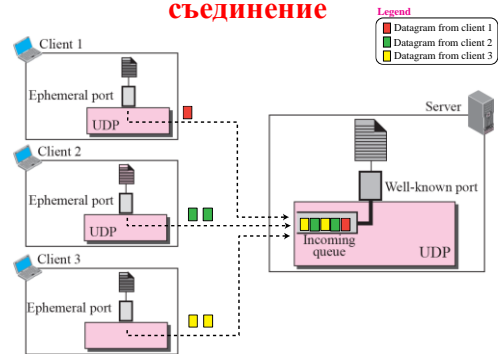
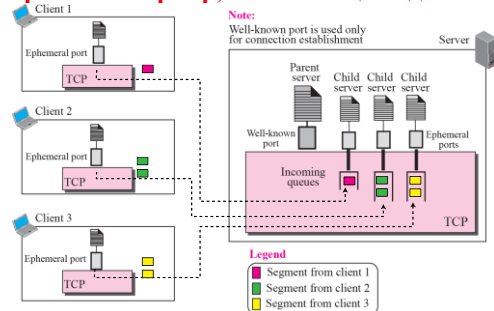


Figure 17.2

Forouzan, B.A., TCP/IP Protocol Suite, 4th ed., McGraw-Hill, 2010

## Паралелен сервър, използващ съединение



One way to run concurrent servers is to dynamically create server processes for each client:

- Parent server accepts incoming requests and starts *child* server (new service thread) for each client
- Child server handles subsequent requests from corresponding client
- Parent server then waits for new requests from other clients

Figure 17.3

Forouzan, B.A., TCP/IP Protocol Suite, 4th ed., McGraw-Hill, 2010