

# Протокол сигнализации SIP

# Определение

«**SIP\*** - является протоколом управления прикладного уровня для создания, изменения и завершения сеансов связи с одним или большим количеством участников. В понятие сеанса входят мультимедиа конференции, обучение на расстоянии, Internet-телефония и подобные приложения» (RFC 2543)

\*SIP – Session Initiation Protocol – Протокол инициализации сессии

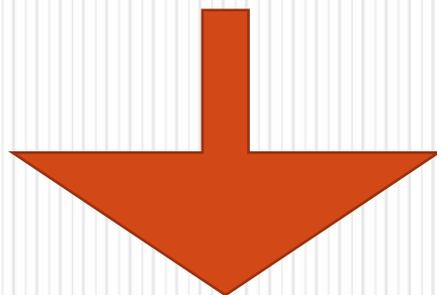
# История создания протокола

## **Session Invitation Protocol:**

- Установление сеанса
- Элементарные возможности согласования
- Работа только поверх UDP
- SDP для описания сеанса

## **Simple Conference Invitation Protocol:**

- Базировался на TCP
- Использовал HTTP и SMTP



## **Session Initiation Protocol:**

- Работа поверх UDP
- Использование SDP
- Поддержка TCP
- ...

# История создания протокола

- Разработкой протокола занимается группа ММUSIC (Multiparty Multimedia Session Control) комитета IETF
- 1996 – создания первой версии SIP
- 1999 – 11 версий документа Draft-ietf-mmusic-SIP
- В настоящее время действующей редакцией является RFC 3261

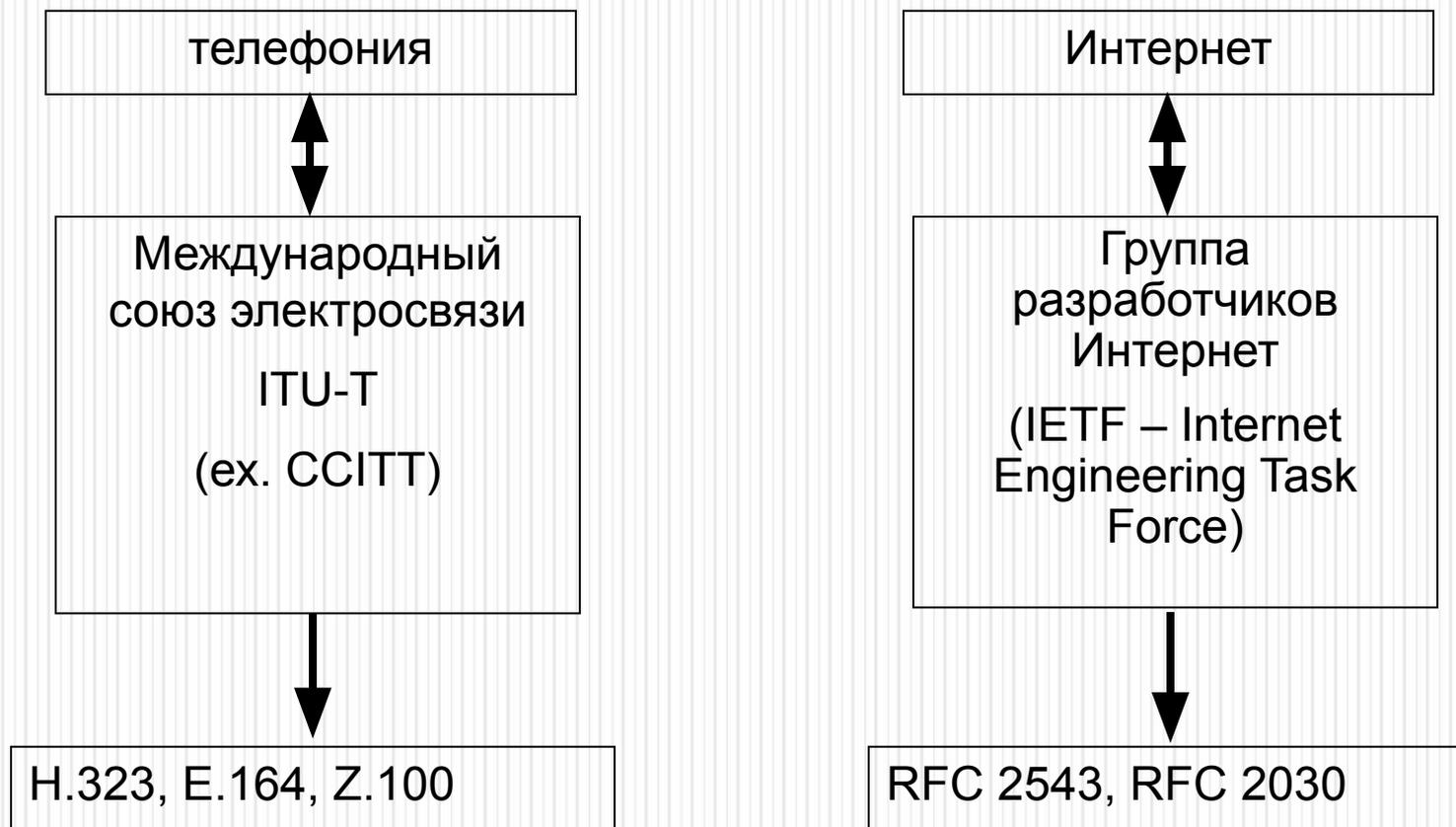
# Принципы, заложенные в основу протокола

- Персональная мобильность пользователя
- Масштабируемость сети
- Расширяемость протокола SIP
- Интеграция в стек существующих протоколов Интернет
- Взаимодействие с различными протоколами сигнализации

# Сравнение систем сигнализации

Характеристики	Сигнализация в традиционных телефонных сетях	Протокол SIP
Функции протоколов	Управление установлением и разрушением соединений, управление коммутационным (транспортным) оборудованием	
Тип сети	TDM	IP
Место применения	Существуют 2 группы протоколов: 1. Между оборудованием доступа и коммутационным узлом или Softswitch (v5.1/5.2 или MEGACO/H.248, MGCP, SIP) 2. Между коммутационными узлами или Softswitch (оКС7, EDSS-1 или SIP/SIP-T, H.323)	
Сетевой интеллект	Интеллект в центральных узлах сети	Интеллект может быть рассредоточен по оконечным элементам сети
Набор сигнальных сообщений	Запрос установления соединения, разрушение соединения, КПВ, занято, ответ и пр.	
Типы соединений	Телеф. Сеансы связи	Мультимедийные сеансы связи
Вид коммутации	Комм-я каналов	Комм-я пакетов
Открытость	Исп-ся исключит-но в сетях сигнализации	Может исп-ся и оконечными терминалами

# Организации стандартизации



# Принципы, заложенные в основу SIP

1. Расширяемость протокола – возможность дополнения протокола новыми функциями
2. Масштабируемость сети – возможность увеличения элементов в сети при её расширении
3. Интеграция в стек существующих протоколов Интернет
4. Взаимодействие с другими протоколами сигнализации
5. Персональная мобильность - возможность быть доступными в любом месте с любым терминалом в любое время (сообщение REGISTER) → единый номер для всех услуг электросвязи

# Особенности протокола SIP

- Основан на HTTP → проверенная технология для работы в Интернет
- Использует и UDP, и TCP
- Работает поверх различных транспортных протоколов (IP, IPX, X.25, ATM)
- Использует адресацию типа e-mail (vova@loniis.ru)
- Текстовый формат сообщений → простота и удобство техобслуживания и программирования
- Высокая информативность сообщений → минимальное время установления соединения

# Возможности протокола SIP

SIP поддерживает пять аспектов организации и завершения мультимедийной связи:

- Определение местоположения пользователя
- Определение готовности пользователя участвовать в сеансе
- Установление сеанса связи как для вызывающей, так и для вызываемой сторон, управление сеансом связи
- Передача пользовательской информации

Организация конференций трех видов:

- В режиме многоадресной рассылки
- При помощи устройства управления конференцией, которому участники передают информацию в режиме точка-точка, а оно, в свою очередь, обрабатывает эту информацию и рассылает участникам конференции
- Соединение каждого пользователя с каждым в режиме точка-точка
- Определение функциональной возможности терминалов пользователей

# Основные характеристики протокола SIP

**Назначение:** для IP-коммуникаций

**Архитектура:** peer-to-peer

**Преимственность:** не пытается воспроизвести ТфОП

**Стандарты:** IETF-стандарт RFC

**Интеллект:** рассредоточен по элементам сети

**Сложность:** еще простой, хотя уже содержит 13 запросов

**Масштабируемость:** высшая степень

**Передача информации:** речь, данные, видео

# Основные характеристики протокола SIP

**Описание функциональных возможностей конечного оборудования:** использование протокола SDP для обмена данными о функциональных возможностях

**Контроль доступа:** поддерживается

**Качество обслуживания:** процедуры QoS

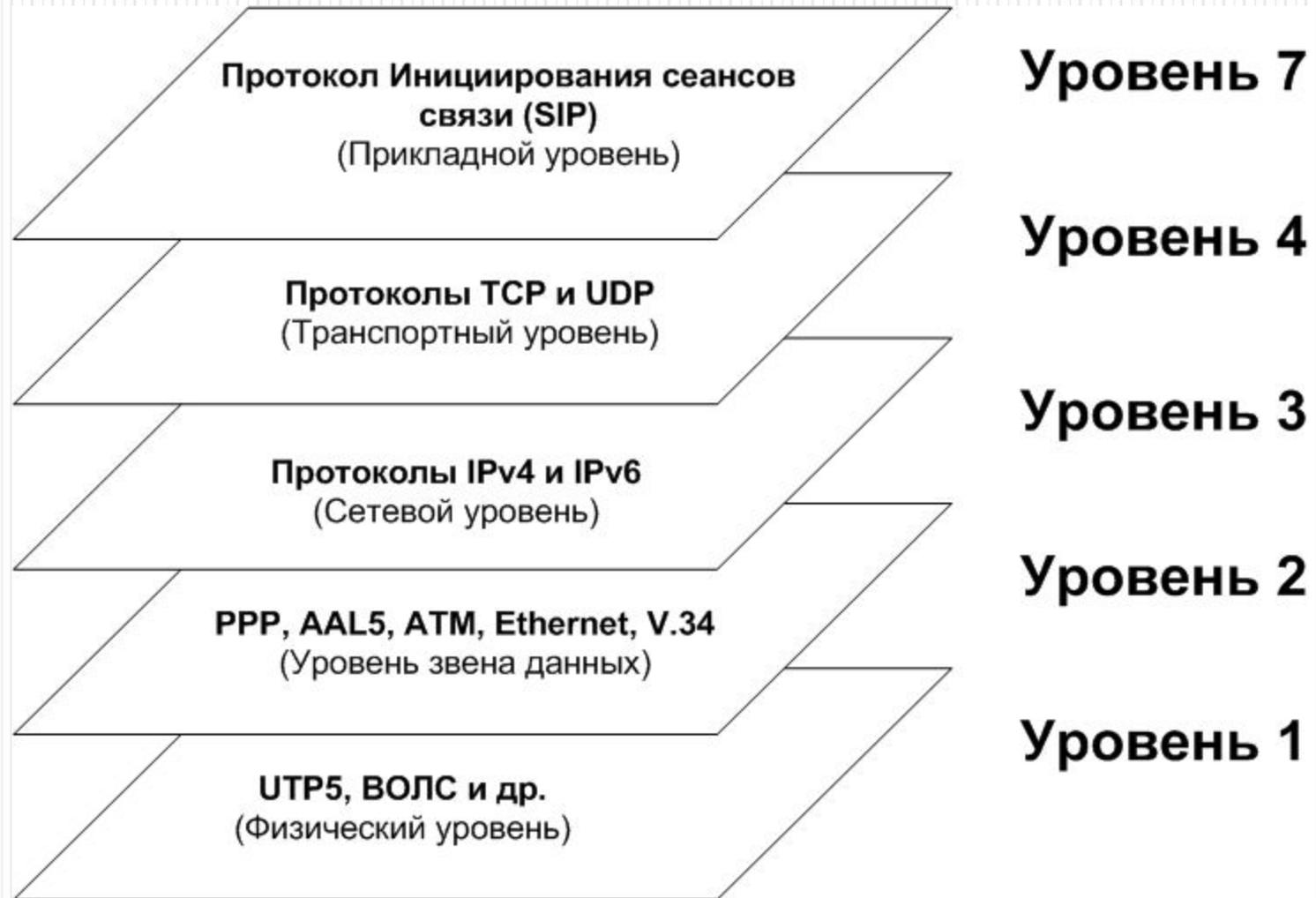
**Адресация:** поддержка IP- адресов и имен доменов через DNS

**Обнаружение закольцованных маршрутов:** с помощью специальных заголовков

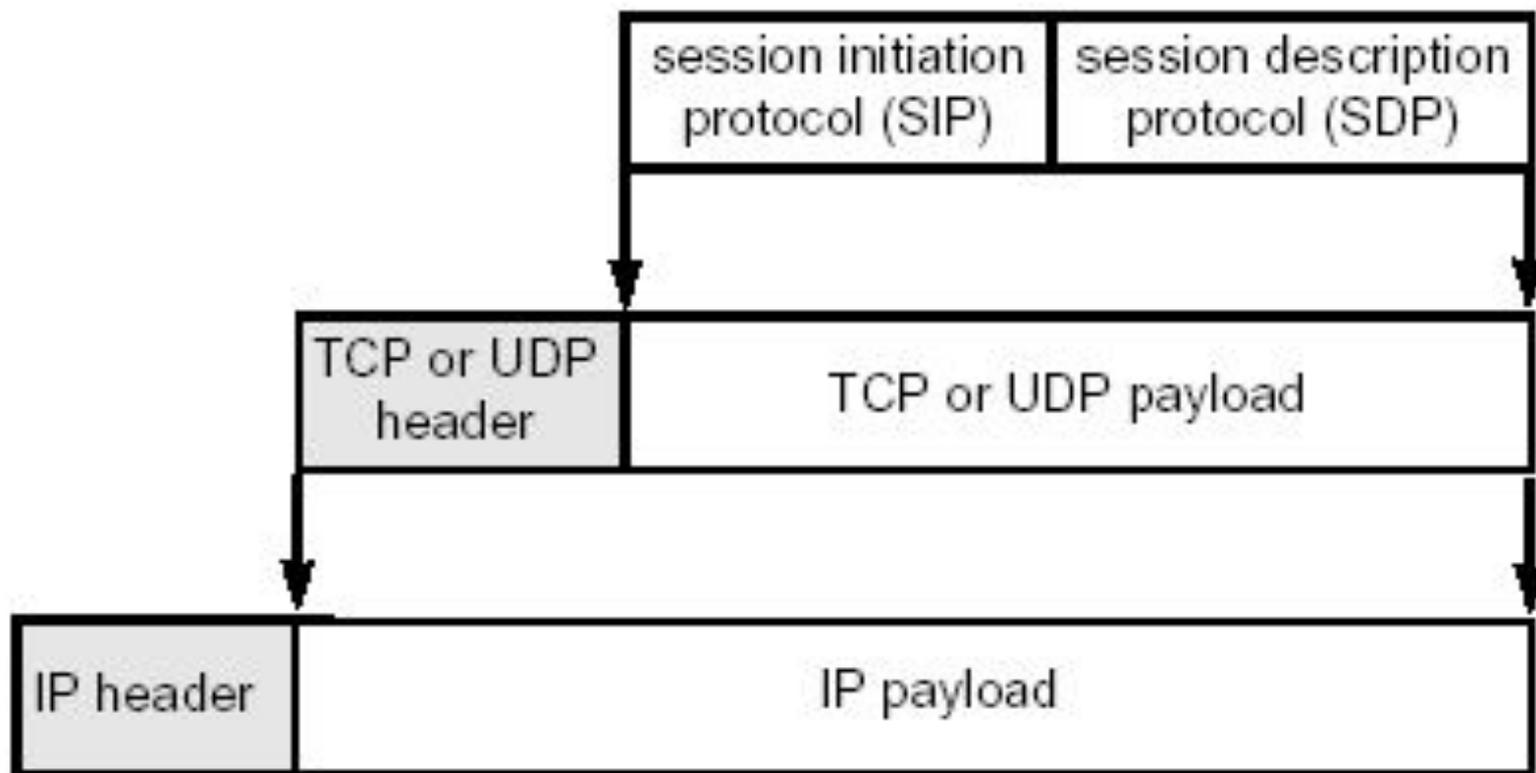
**Защита информации:** протоколы IPSec, TLS, SSL и HTTP Digest

**Кодирование:** текстовое кодирование

# Место протокола SIP в стеке протоколов TCP/IP



# Формирование сообщения сигнализации SIP



# Адресация в SIP

**В Интернет – URL (Uniform Resource Location)**

**В SIP – SIP URL (*sip:name@host*)**

тип адреса

пример

- «*имя@домен*» - sip:vova@loniis.ru
- «*имя@хост*» - sip:vova@rts.loniis.ru
- «*имя@IP-адрес*» - sip:vova@192.168.100.1
- «*№ телефона@шлюз*» - sip:2947678@gateway.ru

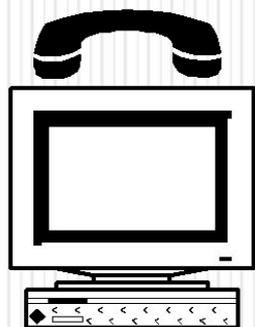
# Уровни протокола SIP

- **Первый уровень** – отвечает за синтаксис и кодирование
- **Второй уровень** – транспортный – определяет, как клиент передает запросы и принимает ответы, и как сервер получает запросы и передает ответы по сети
- **Третий уровень** – уровень транзакций – производит повторную передачу сообщений прикладного уровня, определяет соответствие ответов запросу и уведомляет верхний уровень о срабатывании таймера.
- **Четвертый уровень** – пользователь транзакций – создает/отменяет клиентские запросы

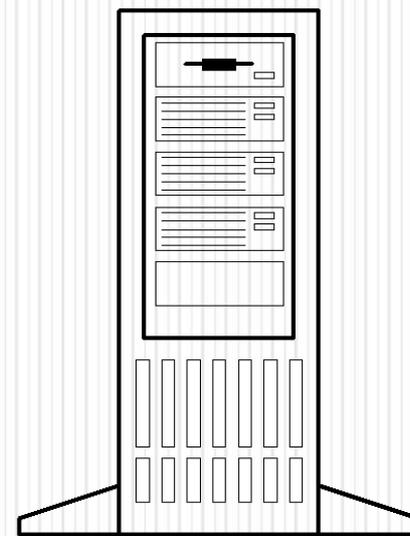
# Понятие транзакции

**Транзакция** - это запрос, переданный клиентской стороной серверной стороне с использованием транспортного уровня SIP, вместе со всеми ответами на этот запрос, переданными серверной стороной клиенту.

# Архитектура «КЛИЕНТ-сервер»



Клиент



Сервер

# Элементы сети SIP

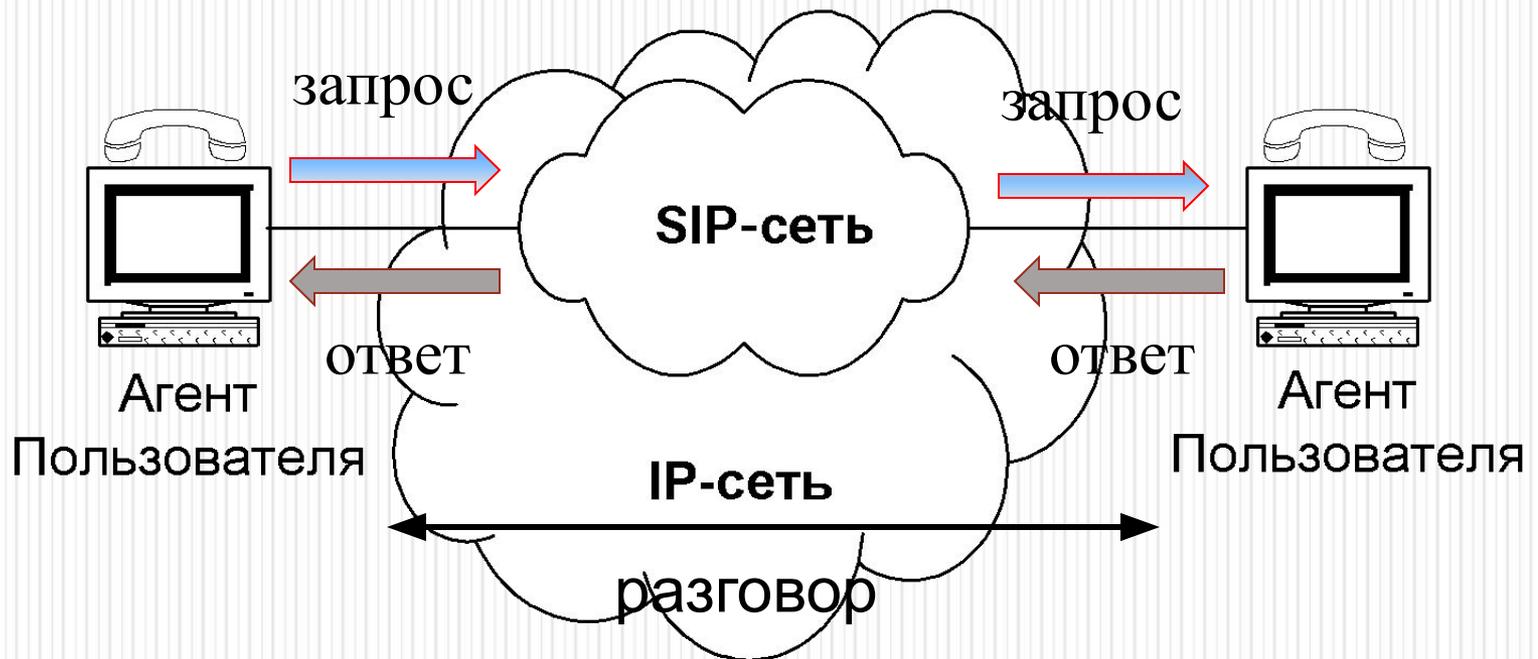
- Агент пользователя (UA – User Agent)
- Прокси-сервер (proxy server)
- Сервер переадресации (redirect server)
- Сервер определения местоположения (location server) (не стандартизирован в RFC 2543)

# Агент пользователя

## **Агент пользователя (User Agent):**

- **Клиент агента пользователя (User Agent Client)** – часть программного обеспечения агента пользователя, которая создает новые запросы, отправляет их и обрабатывает принятые ответы.
- **Сервер агента пользователя (User Agent Server)** - часть программного обеспечения агента пользователя, которая принимает запросы и генерирует ответы, основываясь на действиях пользователя, полученных сообщениях, результатах выполнения программ или на каких-либо других событиях.

# Агент пользователя



# Клиент агента пользователя UAS

- Запрос, составленный клиентом агента пользователя включает в себя :
  - Стартовую строку, в кот. указан тип запроса
  - Поле request-URI и версию SIP
  - Базовый набор полей заголовков: To, from, Cseq, Call-ID, Max-Forwards и Via

Эти заголовки:

- Обязательны для всех SIP-запросов
- Яв-ся основными частями SIP-сообщения
- Обеспечивают большинство услуг маршрутизации сообщений, в т.ч. адресацию, маршру-ю ответов, сохранение очередности сообщений, уникальную идентификацию

# Сервер агента пользователя

## UAS

- Пошаговая обработка запроса:
  - Аутентификация
  - Анализ типа запроса
  - Анализ полей заголовков
  - ...
- Если запрос принимается – должны быть произведены изменения состояния соединения, если не принимается – ни одно из изменений производиться не должно

# Сервер агента пользователя

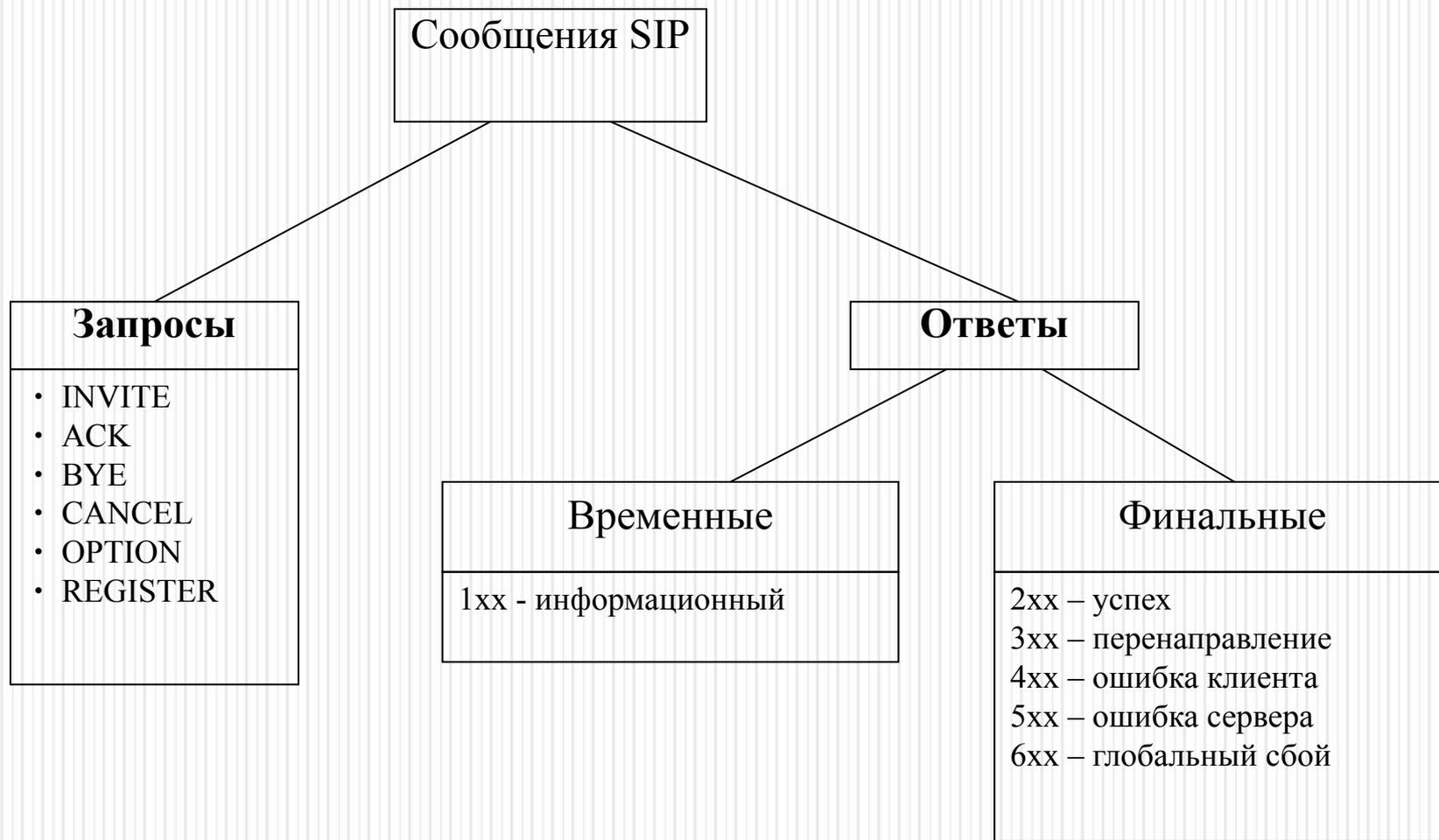
## UAS

принимает запросы, обрабатывает их и, в зависимости от типа запроса, выполняет определенные действия.

Бывает двух типов:

- **Без сохранения состояний (Stateless)** – принимает запросы, перенаправляет их дальше и забывает
- **С сохранением состояний (Stateful)** – принимает запросы, перенаправляет их и ждет ответы

# Сообщения SIP



# Структура сообщения SIP



# Стартовая строка

- Начальная строка любого SIP сообщения. Если сообщение является запросом, то в этой строке указывается тип запроса, адресат и номер версии протокола. Если сообщение является ответом на запрос, в стартовой строке указывается номер версии протокола, тип ответа и его короткая расшифровка, предназначенная только для пользователя.

# Заголовки

служат для передачи информации об отправителе, адресате, пути следования и других сведений, т.е. переносят необходимую для обслуживания данного сообщения информацию. О типе заголовка можно узнать из его имени. В протоколе SIP определено 4 типа заголовков:

- Общие заголовки
- Заголовки содержания
- Заголовки, передающие дополнительную информацию о запросе
- Заголовки, передающие дополнительную информацию об ответе

# Заголовки. Формат

- Каждое поле состоит из имени поля, знака «:» и значение поля:

*Имя поля: значение поля*

- Поля заголовков могут быть расширены на несколько строк, тогда каждая следующая строка отделяется пробелом или знаком табуляции
- Порядок следования заголовков не имеет значения
- Формат значения заголовка зависит от имени заголовка
- Заголовок может иметь неограниченное число параметров
- Одно и то же имя параметра не может использоваться более одного раза

# Заголовки

- Заголовок **Call-ID** – уникальный идентификатор сеанса связи (call reference - DSS-1): **2345call@rts.loniis.ru**
- Заголовок **To** – определяет адресата. Если необходим визуальный вывод имени пользователя, например, на дисплей, то имя пользователя также размещается в поле **To**.
- Заголовок **From** – идентифицирует отправителя запроса; по структуре аналогичен полю **To**.
- Заголовок **CSeq** - уникальный идентификатор запроса, относящегося к одному соединению. Он служит для корреляции запроса с ответом на него. **CSeq: 2 INVITE.**

# Заголовки

Заголовок **Via** указывается весь путь, пройденный запросом: каждый прокси-сервер добавляет поле со своим адресом.

Например, запрос на своем пути обрабатывался двумя прокси-серверами: сначала сервером Ioniis.ru, потом sip.telecom.com. Тогда в запросе появятся следующие поля:

**Via: SIP/2.0/UDP**

**sip.telecom.com:5060;branch=721e418c4.1**

**Via: SIP/2.0/UDP Ioniis.ru:5060**

Заголовок **Content-Type** определяет формат описания сеанса связи. Само описание сеанса, например, в формате протокола SDP включается в тело сообщения.

Заголовок **Content-Length** указывает размер тела сообщения

# Сжатые имена заголовков

Сжатая форма имени	Полная форма имени
C	Content-Type
E	Content-Encoding
F	From
I	Call-ID
K	Supported
L	Content-Length
M	Contact (от “moved”)
S	Subject
O	Event
R	Refer-To
T	To
U	Allow-Events
V	Via

# Тело сообщения

- **Запросы:**

- Содержит описание сеансов связи
- Тело сообщения есть не во всех сообщениях (например, сообщение ВУЕ не содержит тела сообщения)

- **Ответы:**

- Любые ответы могут содержать тело сообщения, но содержимое тела в них может быть разным

# Пример сообщения SIP

Сообщение SIP

Стартовая строка

INVITE sip:7170@iptel.org SIP/2.0

Via: SIP/2.0/UDP 195.37.77.100:5040;rport  
Max-Forwards: 10  
From: "jiri" <sip:jiri@iptel.org>;tagi=76ff7a07-c091-4192-84a0-d56e91fe104f  
To: <sip:jiri@bat.iptel.org>  
Call-ID: d10815e0-bf17-4afa-8412-d9130a793d96@213.20.128.35  
CSeq: 2 INVITE  
Contact: <sip:213.20.128.35:9315>  
User-Agent: Windows RTC/1.0  
Proxy-Authorization: Digest username="jiri", realm="iptel.org",  
algorithm="MD5", uri="sip:jiri@bat.iptel.org",  
nonce="3cef75390000001771328f5ae1b8b7f0d742da1feb5753c",  
response="53fe98db10e1074  
b03b3e06438bda70f"  
Content-Type: application/sdp  
Content-Length: 451

Заголовок

v=0  
o=jku2 0 0 IN IP4 213.20.128.35  
s=session  
c=IN IP4 213.20.128.35  
b=CT:1000  
t=0 0  
m=audio 54742 RTPi/AVP 97 111 112 6 0 8 4 5 3 101  
a=rtpmap:97 red/8000  
a=rtpmap:111 SIREN/16000  
a=fmtp:111 bitrate=16000  
a=rtpmap:112 G7221/16000  
a=fmtp:112 bitrate=24000  
a=rtpmap:6 DVI4/16000  
a=rtpmap:0 PCMU/8000  
a=rtpmap:4 G723/8000  
a=rtpmap: 3 GSMi/8000  
a=rtpmap:101 telephone-event/8000  
a=fmtp:101 0-16

Тело сообщения

# Пример сообщения SIP

Request URI  
Тип сообщения  
Версия протокола

**INVITE sip:7170@iptel.org SIP/2.0**

Заголовок VIA. Указывает один из узлов, используемых для проведения транзакции и идентифицирует место, куда должен быть отправлен ответ

**Via: SIP/2.0/UDP 195.37.77.100:5040;rport**

Максимальное число пересылок на пути к месту назначения

**Max-Forwards: 10**

Идентификатор инициатора сообщения

**From: "jiri" <sip:jiri@iptel.org>;tagi=76ff7a07-c091-4192-84a0-d56e91fe104f**

Адрес получателя

**To: <sip:jiri@bat.iptel.org>**

**Call-ID: d10815e0-bf17-4afa-8412-d9130a793d96@213.20.128.35**

Идентификация транзакции

**CSeq: 2 INVITE**

URI

Информация о клиенте Агента пользователя

**Contact: <sip:213.20.128.35:9315>**

Идентифицирует пользователя прокси-серверу

**User-Agent: Windows RTC/1.0**

**Proxy-Authorization: Digest username="jiri", realm="iptel.org", algorithm="MD5", uri="sip:jiri@bat.iptel.org", nonce="3cef75390000001771328f5ae1b8b7f0d742da1feb5753c", response="53fe98db10e1074b03b3e06438bda70f"**

Информация о типе тела сообщения

**Content-Type: application/sdp**

Размер тела сообщения

**Content-Length: 451**

```
v=0
o=jku2 0 0 IN IP4 213.20.128.35
s=session
c=IN IP4 213.20.128.35
b=CT:1000
t=0 0
m=audio 54742 RTPi/AVP 97 111 112 6 0 8 4 5 3 101
a=rtpmap:97 red/8000
a=rtpmap:111 SIREN/16000
a=fmtp:111 bitrate=16000
a=rtpmap:112 G7221/16000
a=fmtp:112 bitrate=24000
a=rtpmap:6 DVI4/16000
a=rtpmap:0 PCMU/8000
a=rtpmap:4 G723/8000
a=rtpmap: 3 GSMi/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-16
```

Тело сообщения  
содержит описание типа  
медиа данных, которые  
будут передаваться  
пользователем

# Запросы

- Запросы предназначены для выполнения широкого круга задач при предоставлении базовых и дополнительных услуг связи в стационарных сетях и сетях подвижной связи.
- С помощью запросов клиент сообщает о своем текущем местоположении, приглашает пользователей принять участие в сеансах связи, модифицирует уже установленные сеансы, завершает их и т.д.

# Запросы

Тип запроса	Описание запроса
<b>INVITE</b>	Приглашает пользователя к сеансу связи. Содержит SDP-описание сеанса
<b>ACK</b>	Подтверждает прием окончательного ответа на запрос <b>INVITE</b>
<b>BYE</b>	Завершает сеанс связи. Может быть передан любой из сторон, участвующих в сеансе
<b>CANCEL</b>	Отменяет обработку запросов с теми же заголовками <b>Call-ID, To, From</b> и <b>CSeq</b> , что и в самом запросе <b>CANCEL</b>
<b>REGISTER</b>	Переносит адресную информацию для регистрации пользователя на сервере определения местоположения
<b>OPTION</b>	Запрашивает информацию о функциональных возможностях сервера

# Запросы

<b>Тип запроса</b>	<b>Описание запроса</b>
<b>UPDATE</b>	Предлагает новые параметры сеанса связи до прихода окончательного ответа на запрос INVITE
<b>INFO</b>	Переносит дополнительную информацию во время сеанса связи.
<b>PRACK</b>	Аналог сообщения ACK для предварительных ответов
<b>SUBSCRIBE NOTIFY</b>	Используются для предоставления дополнительных услуг
<b>REFER</b>	Команда перевода вызова
<b>MESSAGE</b>	Обеспечивает передачу пользовательской информации без установления сеанса связи
<b>PUBLISH</b>	Обеспечивает передачу информации о состоянии агента пользователя.

# Структура запроса

Тип запроса	Пробел	Request- URI	Пробел	Версия протокола	CRLF
----------------	--------	-----------------	--------	---------------------	------

# Структура запроса

- SIP – запросы характеризуются наличием строки Request-Line в стартовой строке.
- Request-Line состоит из названия типа запроса, Request-URI и версии протокола, разделенных пробелом.
- Request-Line заканчивается символами возврата каретки и перевода строки (CRLF).
- Оба символа вместе или по одиночке не должны встречаться в других частях строки.
- Использование линейного пробела не допускается.

# Тип запроса

**6 типов запросов (RFC 3261):**

- REGISTER
- INVITE
- ACK
- CANCEL
- BYE
- OPTION

# Request-URI

Указывает пользователя или услугу, к которой адресован запрос. Поле Request-URI не должно содержать пробелов и управляющих символов, а также не должно быть заключено в угловые скобки

«<>»

# Версия протокола

И запросы и ответы содержат данные действующей версии SIP-протокола, принимая во внимание порядок, соответствие требованиям и изменение численного индекса версии

# Запрос INVITE

- приглашает вызываемого пользователя принять участие в сеансе связи.
- содержит описание сессии, в котором передается вид принимаемой информации и параметры, необходимые для приема информации, также может указываться вид информации, который вызываемый пользователь желает передавать.
- В ответе на запрос INVITE указывается вид информации, которая будет приниматься вызываемым пользователем, кроме того, может указываться вид информации, которую вызываемый пользователь собирается передавать (возможные параметры передачи информации).

# Пример запроса INVITE

INVITE sip: alexander@serv1.loniis.ru SIP/2.0

Via: SIP/2.0/UDP kton.loniis.ru

From: Anton <sip: anton@loniis.ru>

To: Alexander <sip: alexander@loniis.ru>

Call-ID: 3298420296@kton.loniis.ru

Cseq: 1 INVITE

Content-Type: application/sdp

Content-Length: ...

v=0

o=bell 53655765 2353687637 IN IP4 128.3.4.5

C=IN IP4 kton.loniis.ru

m=audio 3456 RTP/AVP 0 3 4

# Запросы

## ● АСК

- подтверждает прием ответа на команду **INVITE**.
- оборудование вызывающего пользователя показывает, что оно получило окончательный ответ на свой запрос **INVITE**.
- может содержаться окончательное описание сессии, передаваемое вызывающим пользователем.

## ● CANCEL

- отменяет обработку ранее переданных запросов.

# Запросы. Register

- В этом запросе пользователи сообщают свое текущее местоположение.
- В этом сообщении содержатся следующие заголовки:
  - **To** содержит адресную информацию, которую надо сохранить или модифицировать на сервере.
  - **From** содержит адрес инициатора регистрации.
  - **Contact** содержит новый контактный адрес пользователя, по которому должны посылаться все дальнейшие запросы **INVITE**.
  - **Expires** указывается время в секундах, по истечении которого регистрация заканчивается.

# Запрос INFO

- предназначен для обмена сигнальной информацией по SIP сигнальному тракту в процессе установления и поддержания соединения.
- Запрос INFO не изменяет состояния SIP вызовов, также как не изменяет состояния сессий, инициированных при помощи протокола SIP.
- Обеспечивает передачу дополнительной информации прикладного уровня, которая в дальнейшем может способствовать более производительному функционированию приложений, использующих протокол SIP для доставки данной информации.

# Запрос INFO

- Возможными применениями для типа запроса INFO являются:
  - Перенос текущих сигнальных сообщений ТфОП между шлюзами ТфОП в течении разговорной сессии
  - Перенос DTMF сигналов, сгенерированных в ходе сессии
  - Перенос защищённой информации сигналов беспроводных систем для поддержки беспроводной мобильности приложений
  - Перенос информации об остатке на счёте (билингвой информации)
  - Перенос изображений и другой не потоковой информации между участниками сессии

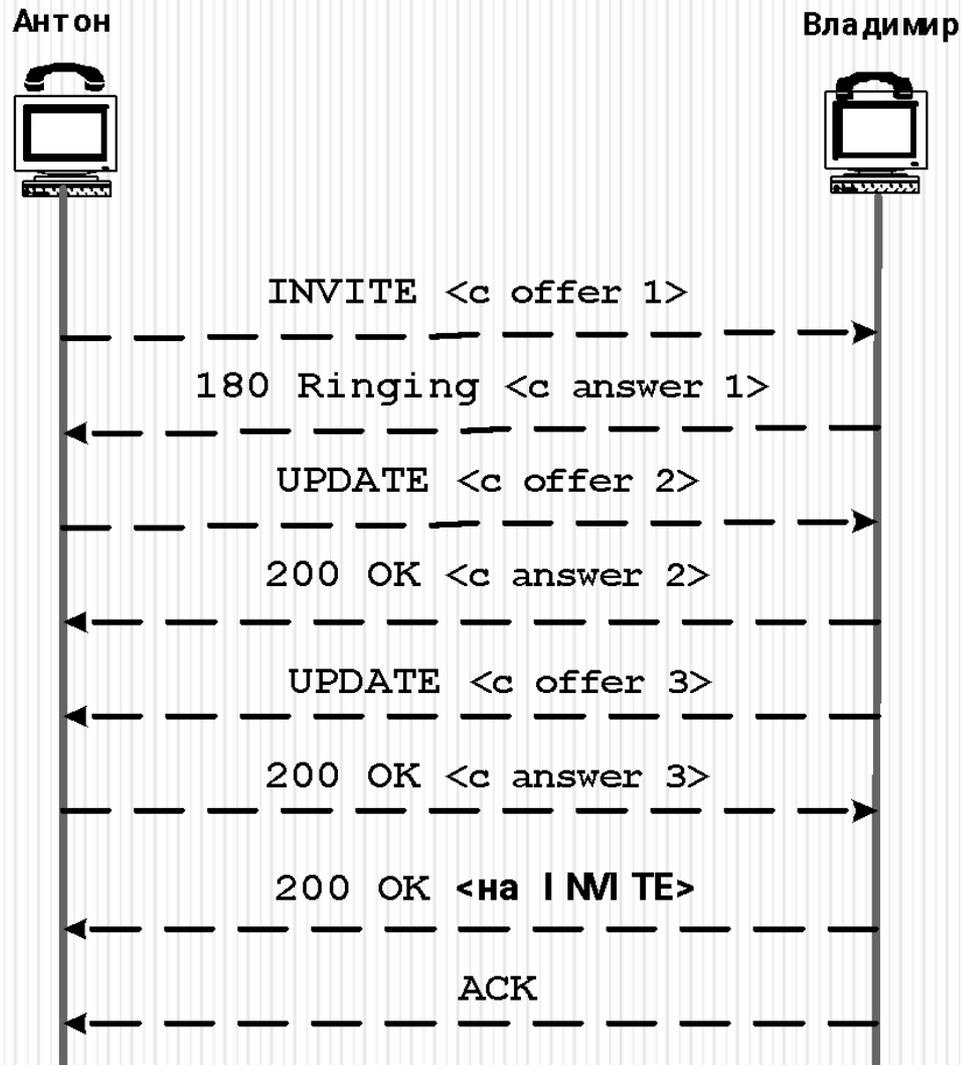
# Запрос UPDATE

- Используется, когда необходимо изменить параметры сеанса:
- Вызывающая сторона создаёт начальную INVITE-транзакцию.
- В поле заголовка Allow запроса INVITE среди прочих типов запроса указывается UPDATE для того, чтобы указать способность вызывающей стороны принимать запросы этого типа. INVITE-транзакция протекает надлежащим образом. Любой ответ (предварительный или окончательный) от вызываемой стороны также содержит заголовок Allow с указанным в нём значением *UPDATE*.

# Запрос UPDATE

- После установления диалога (находящегося на ранней стадии или установленного), вызывающая сторона может создать запрос **UPDATE**, который содержит информацию **offer** (предложение с описанием сеанса связи в формате **SDP**), предназначенное для обновления параметров сессии.
- Ответ на этот запрос переносит информацию **answer** (ответ на предложение с указанием принятых параметров также в формате **SDP**)
- Запрос **UPDATE** является запросом, обновляющим текущий адрес удалённого пользователя (**target refresh request**).

# Запрос UPDATE. Пример



# NOTIFY

- Для услуг, которые требуют взаимодействия между конечными точками, необходима возможность запрашивать асинхронное уведомление о событиях
- Эти услуги включают:
  - услуги автоматического обратного вызова (связанные с событиями изменения состояния терминала пользователя)
  - интерактивные списки контактов «buddy lists» (связанные с событиями присутствия пользователя), оповещение об ожидающем сообщении (связанные с событиями изменения состояния почтового ящика)
  - передача информации о состоянии вызова при взаимодействии сетей Internet и ТфОП.

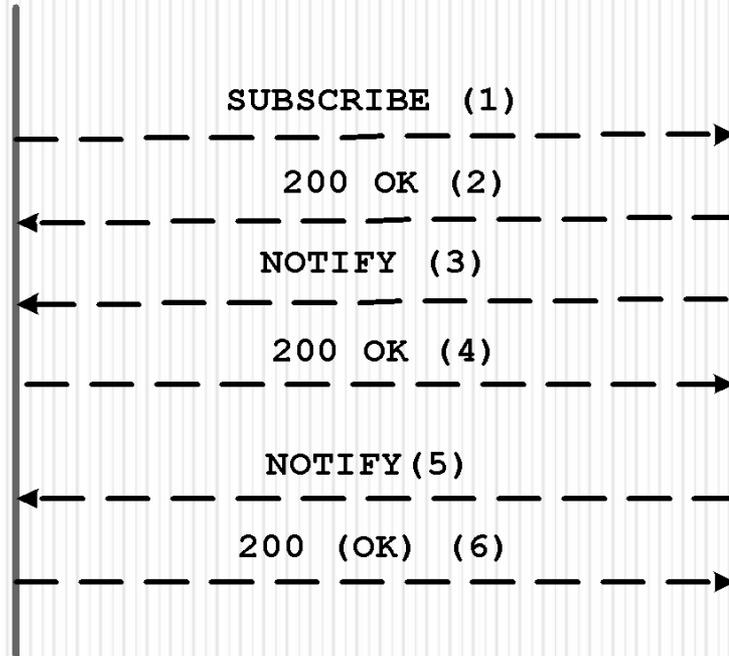
# NOTIFY

- Для услуг, которые требуют взаимодействия между конечными точками, необходима возможность запрашивать асинхронное уведомление о событиях
- Эти услуги включают:
  - услуги автоматического обратного вызова (связанные с событиями изменения состояния терминала пользователя)
  - интерактивные списки контактов «buddy lists» (связанные с событиями присутствия пользователя), оповещение об ожидающем сообщении (связанные с событиями изменения состояния почтового ящика)
  - передача информации о состоянии вызова при взаимодействии сетей Internet и ТфОП.

# NOTIFY

UA подписчика

UA уведомителя



(1) – запрос подписки на предоставление информации о состоянии

(2) – подтверждение подписки

(3) – передача информации о текущем состоянии

(4) – подтверждение приёма

(5) – передача информации о текущем состоянии

(6) – подтверждение приёма

# NOTIFY

- Запрос **SUBSCRIBE** используется для запроса информации о текущем состоянии и информации об обновлениях состояния удалённого ресурса.
- **SUBSCRIBE** – это запрос, создающий диалог.
- Если начальное сообщение **SUBSCRIBE** представляет собой запрос вне диалога, его формирование проходит с привлечение процедур по созданию запроса вне диалога для UAC.
- Идентификация событий, на которые осуществляется подписка, обеспечивается с помощью трёх компонентов запроса **SUBSCRIBE**: поля **Request URI**, заголовка **Event** и опционально тела сообщения.

# NOTIFY

- Запрос **SUBSCRIBE** должен быть подтверждён окончательным ответом.
- После того, как подписка была успешно создана или обновлена, уведомитель должен незамедлительно отослать сообщение **NOTIFY**, чтобы сообщить подписчику текущее состояние ресурса
- Запрос **NOTIFY** посылается в том же диалоге, который был создан ответом на запрос **SUBSCRIBE** (если не существовало заранее установленного диалога).
- Тип запроса **NOTIFY** используется для уведомления узла **SIP** о том, что событие, информация о котором запрашивалась в запросе **SUBSCRIBE**, произошло.

# Сообщение Message

- Запрос типа MESSAGE предназначен для передачи мгновенных текстовых сообщений (instant messages), используя модель, похожую на функционирование двустороннего пейджера или работу телефона при отправке SMS.
- Каждое текущее сообщение (IM) независимо – информация о том, что происходит взаимодействие, переговоры между участниками, может быть отражена только в пользовательском интерфейсе клиента
- Такой подход полностью противоположен сессионной модели, где происходит однозначное взаимодействие участников с чётко определённым началом и концом.

# Сообщение Message

- Когда один пользователь решает послать другому пользователю текущее текстовое сообщение (IM), отправитель формирует запрос типа MESSAGE и передаёт его.
- Поле Request-URI запроса как обычно будет указывать публичный адрес получателя текущего сообщения, однако оно может содержать и адрес устройства в случаях, когда клиент владеет информацией о текущем местонахождении получателя
- Тело сообщения будет включать текстовое сообщение, которое необходимо доставить.

# Сообщение Message

- Запрос MESSAGE пройдёт через группу прокси-серверов и будет доставлен получателю.
- Получив запрос, UA получателя перейдёт к его обработке и в случае успеха в итоге отошлёт окончательный ответ с кодом 200 (ОК); это означает, что текстовое сообщение было доставлено пользователю, но указывает на то, что пользователь с ним ознакомился.

# Сообщение Message



MESSAGE sip:alexander@niits.ru SIP/2.0  
Via: SIP/2.0/TCP  
serv3.niits.ru;branch=z9hG4bK776sgdkse  
Max-Forwards: 70  
From: sip:anton@niits.ru;tag=49583  
To: sip:alexander@niits.ru  
Call-ID: asd88asd77a@1.2.3.4  
CSeq: 1 MESSAGE  
Content-Type: text/plain  
Content-Length: 18

Александр, иди сюда.

# ОТВЕТЫ

Шесть типов ответов:

**1xx** – информационные

**2xx** – успех

**3xx** – перенаправление

**4xx** – ошибка клиента

**5xx** – ошибка сервера

**6xx** – глобальная ошибка

# ОТВЕТЫ 1XX

## Информационные ответы

- **100 Trying** - Запрос обрабатывается, например, сервер обращается к базам данных, но местоположение вызываемого пользователя в настоящий момент не определено
- **180 Ringing** - Местоположение вызываемого пользователя определено. Ему дается сигнал о входящем вызове
- **183 Session Progress** – Этот ответ используется для того, чтобы заранее получить от шлюзов, стоящих на пути к вызываемому пользователю, описание сессии для проключения разговорного тракта в предответном состоянии

# Ответы 2xx

## Ответы об успешной обработке запроса

- **200 OK** - Команда успешно выполнена
- **202 Accepted** – Запрос принят для обработки, но она еще не завершена.

# Ответы 3xx

## Ответы переадресации вызова

- **300 Multiple Choices** - Вызываемый пользователь доступен по нескольким адресам. Вызывающий пользователь может выбрать любой из них.
- **301 Moved Permanently** - Пользователь изменил свое местоположение, его новый адрес указан в поле Contact
- **302 Moved Temporarily** Пользователь временно изменил свое местоположение, его новый адрес указан в поле Contact

# ОТВЕТЫ 4XX

- **400 Bad Request** - В запросе обнаружена синтаксическая ошибка
- **401 Unauthorized** – Запрос требует проведения процедуры авторизации пользователя
- **404 Not Found** – Вызываемый пользователь не обнаружен
- **407 Proxy Authentication Required** – Перед вызовом требуется пройти процедуру аутентификации
- **486 Busy Here** – Вызываемый пользователь в данный момент либо не желает либо не имеет возможности принять еще один вызов в дополнение к уже принятым

# ОТВЕТЫ 5XX

## ОТВЕТЫ об отказе сервера

- **500 Internal Server Error** - Внутренняя ошибка сервера
- **501 Not Implemented** - Сервер не может обслужить запрос, потому что в сервере не реализованы соответствующие функции
- **503 Service Unavailable** – Обслуживание временно невозможно вследствие перегрузки или из-за проведения технического обслуживания

# ОТВЕТЫ 6XX

**Ответы о полной невозможности установить соединение**

- **600 Busy Everywhere** Вызываемый пользователь занят и не желает принимать вызов в данный момент. Ответ может указывать подходящее для вызова время.
- **603 Decline** Вызываемый пользователь не может или не желает принять входящий вызов, не указывая причины отказа
- **604 Does Not Exist Anywhere** – Вызываемый пользователь не существует

# Диалог

- Равноправное взаимодействие двух агентов пользователя по протоколу SIP, которое длится определённое время.
- Диалог устанавливает последовательность сообщений между UA и обеспечивает верную маршрутизацию запросов.
- Идентифицируется каждым агентом пользователя с помощью идентификатора диалога (dialog ID), который состоит из значения Call-ID, локальной метки (local tag) и удалённой метки (remote tag).

# Диалог

- У участвующих в диалоге сторон идентификатор диалога имеет свои отличия: локальная метка одного UA идентична удалённой метке другого и наоборот.
- Идентификатор диалога напрямую связан со всеми запросами, имеющими параметр «tag» в поле заголовка To.

# Диалог

- Компоненты, описывающие состояние диалога и используемые для передачи сообщений в ходе диалога:
  - идентификатор диалога
  - локальный порядковый номер (для упорядочивания запросов UA, направляемых своему собеседнику),
  - удалённый порядковый номер (для упорядочивания запросов от собеседника к UA)
  - локальный URI
  - удалённый URI
  - текущий адрес удалённого пользователя (remote target)
  - булев флаг «secure»
  - установленный маршрут (**route set**)

# Создание диалога

- Диалоги создаются путём возврата ответов, не информирующих об ошибках, на запросы определённых типов.
- В данной версии протокола установить диалог возможно только ответами 101-199 и 2xx с параметром «tag» в поле заголовка To на запрос INVITE.
- Диалог, установленный предварительным ответом на запрос, называется диалогом «на ранней стадии» (early dialog).

# Диалог

- Как только соединение между двумя агентами пользователя установлено, любой из них может стать инициатором новых транзакций, необходимых в рамках диалога:
  - UA, отсылающий запросы будет выполнять роль клиента в транзакции,
  - UA, принимающий запросы, будет выполнять роль сервера.

Эти роли могут отличаться от тех, которые исполняли агенты пользователя в транзакции создания диалога.

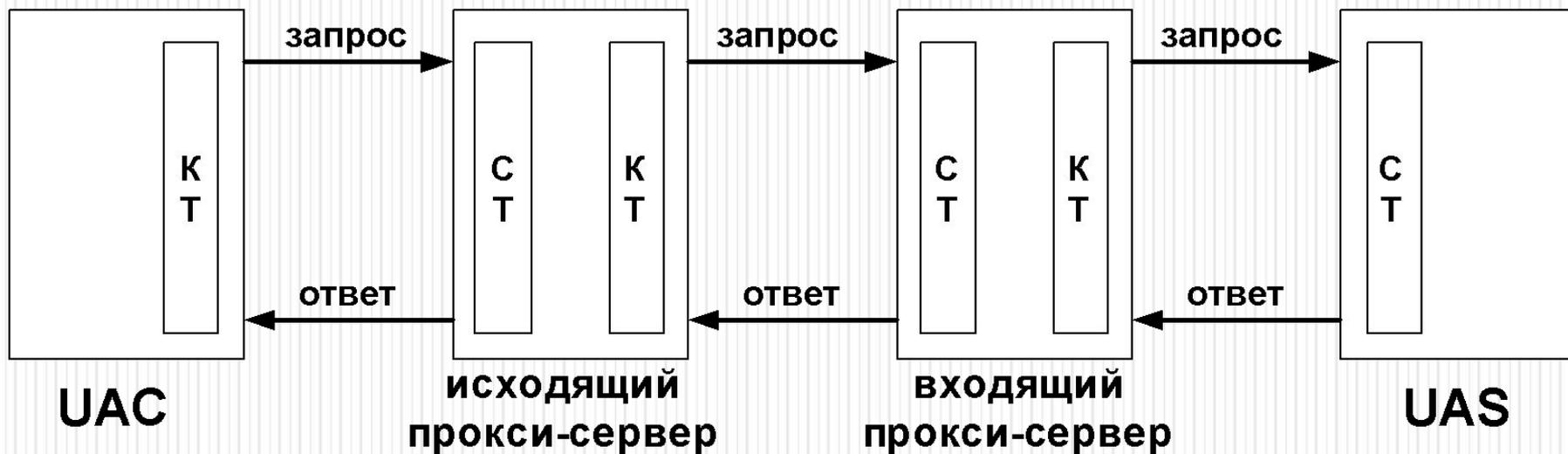
# Завершение диалога

- Когда на любой запрос вне диалога приходит окончательный ответ класса, отличного от 2xx, диалоги, находящиеся на «ранней стадии», созданные посредством предварительных ответов, разрушаются.
- Для разрушения установленных диалогов используется запрос ВУЕ

# Транзакции

- Транзакция состоит из запроса и любого количества ответов на него.
- Транзакция имеет клиентскую сторону и серверную сторону, соответственно они носят название клиентской транзакции и серверной транзакции.
  - Клиентская транзакция занимается отправкой запросов,
  - Серверная транзакция – отправкой ответов.
- Они создаются в агентах пользователя и прокси-серверах с сохранением состояний (stateful).

# клиентской и серверной транзакции.



КТ - клиентская транзакция  
СТ - серверная транзакция

# Функционирование клиентских транзакций

- Когда **TU** нужно инициировать новую транзакцию, он создает клиентскую транзакцию и передает ей:
  - SIP-запрос, предназначенный для отправки,
  - IP-адрес,
  - номер порта,
  - тип транспортного протокола, на которые должен быть передан запрос.

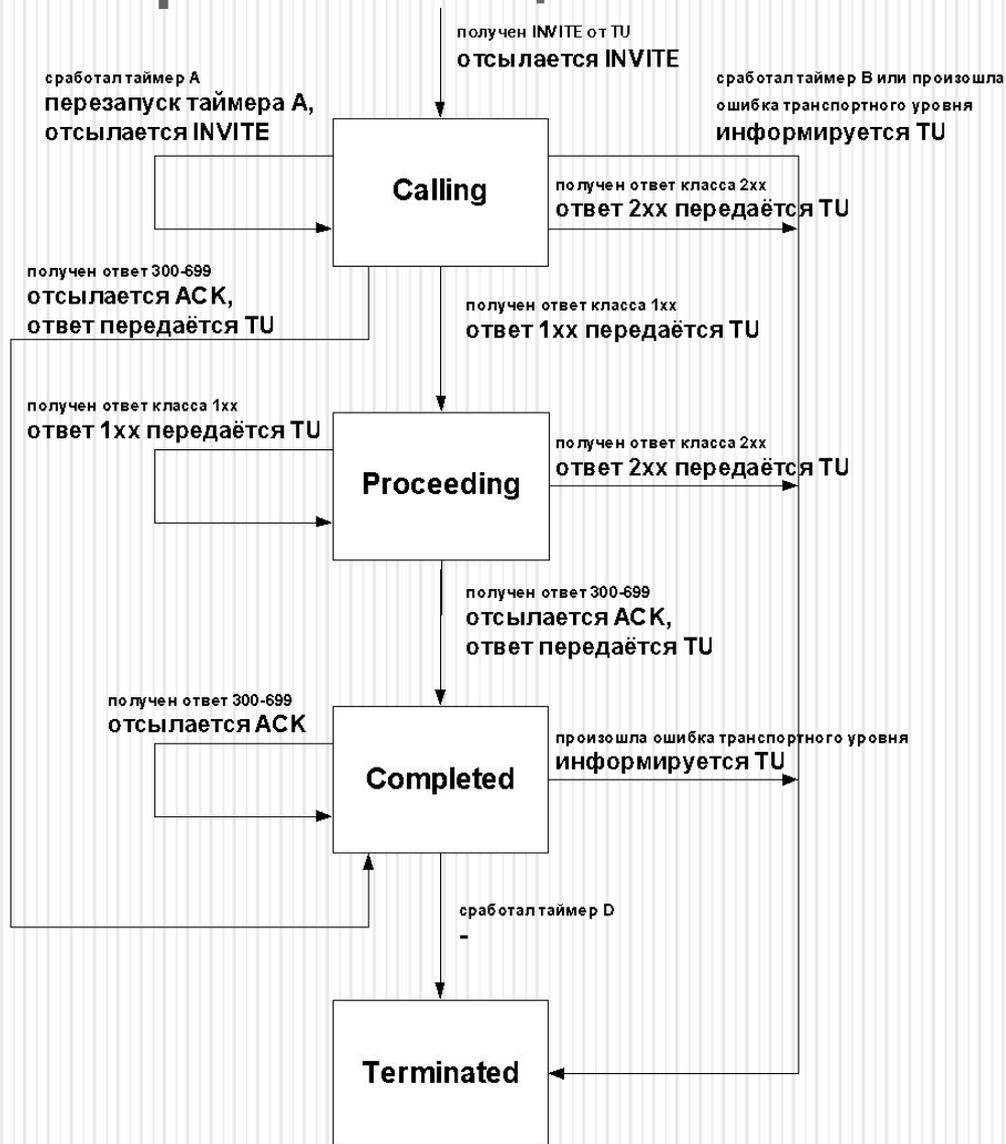
# Функционирование клиентских транзакций

Существует два конечных автомата клиентских транзакций в зависимости от типа запроса, который передаёт TU:

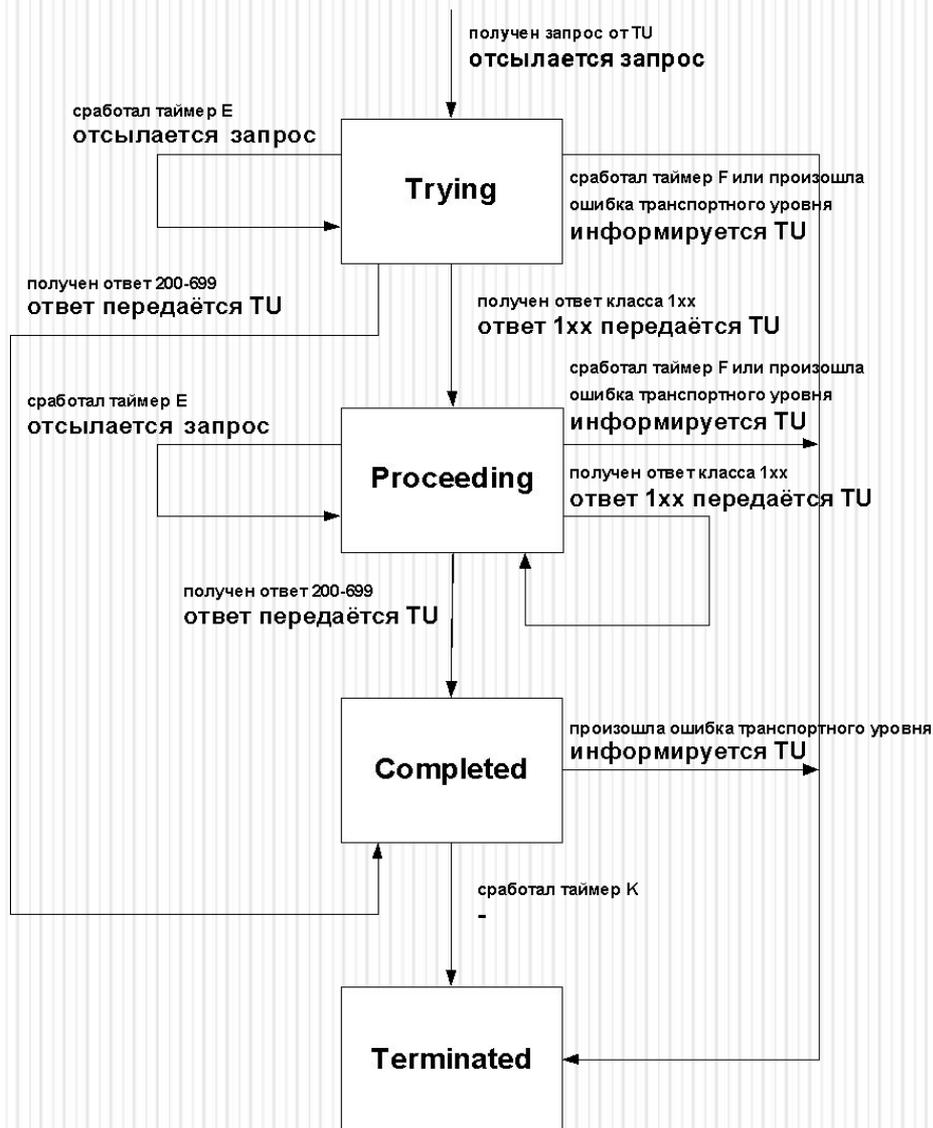
- Один из них поддерживает клиентские транзакции для запросов INVITE (клиентская INVITE-транзакция)
- Второй тип поддерживает клиентские транзакции всех типов запросов, исключая INVITE и ACK

Не существует отдельной клиентской транзакции для запроса ACK: когда у TU возникает необходимость передать этот запрос (при подтверждении ответа класса 2xx на запрос INVITE), он отправляет его напрямую транспортному уровню SIP для доставки по сети.

# Конечный автомат клиентской INVITE-транзакции



# Конечный автомат клиентской не-INVITE-транзакции



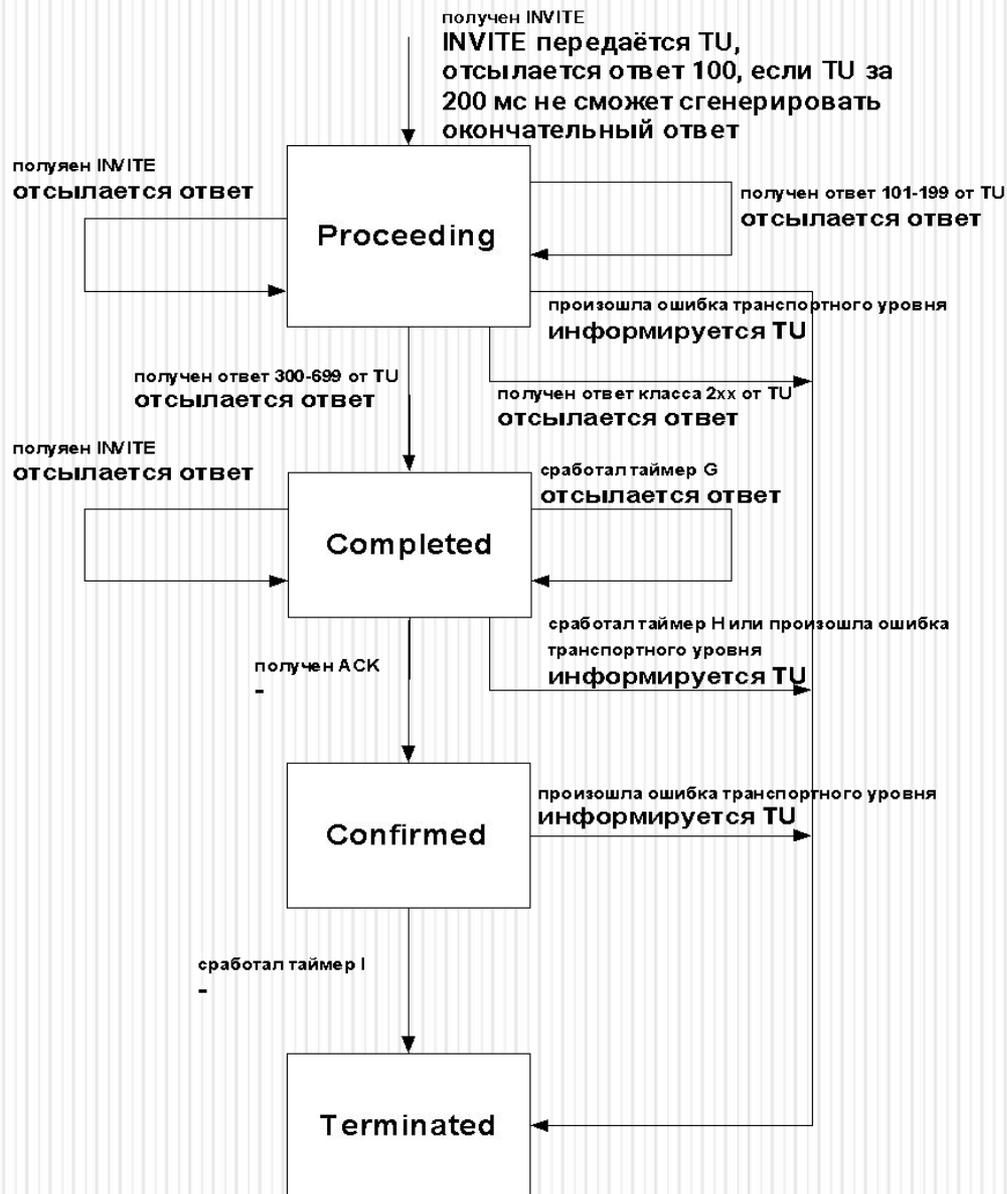
# Соответствие ответов клиентским транзакциям

- Когда транспортный уровень клиента принимает ответ, он должен выяснить, какой клиентской транзакции он принадлежит.
- Для этой цели существует параметр «branch» в верхнем заголовке *Via*.
- Ответ соответствует клиентской транзакции при выполнении двух условий:
  - Верхний заголовок *Via* ответа имеет то же значение параметра «branch», что и аналогичный верхний заголовок запроса, инициировавшего транзакцию.
  - Поле типа запроса в заголовке *CSeq* соответствует типу запроса, создавшему транзакцию.

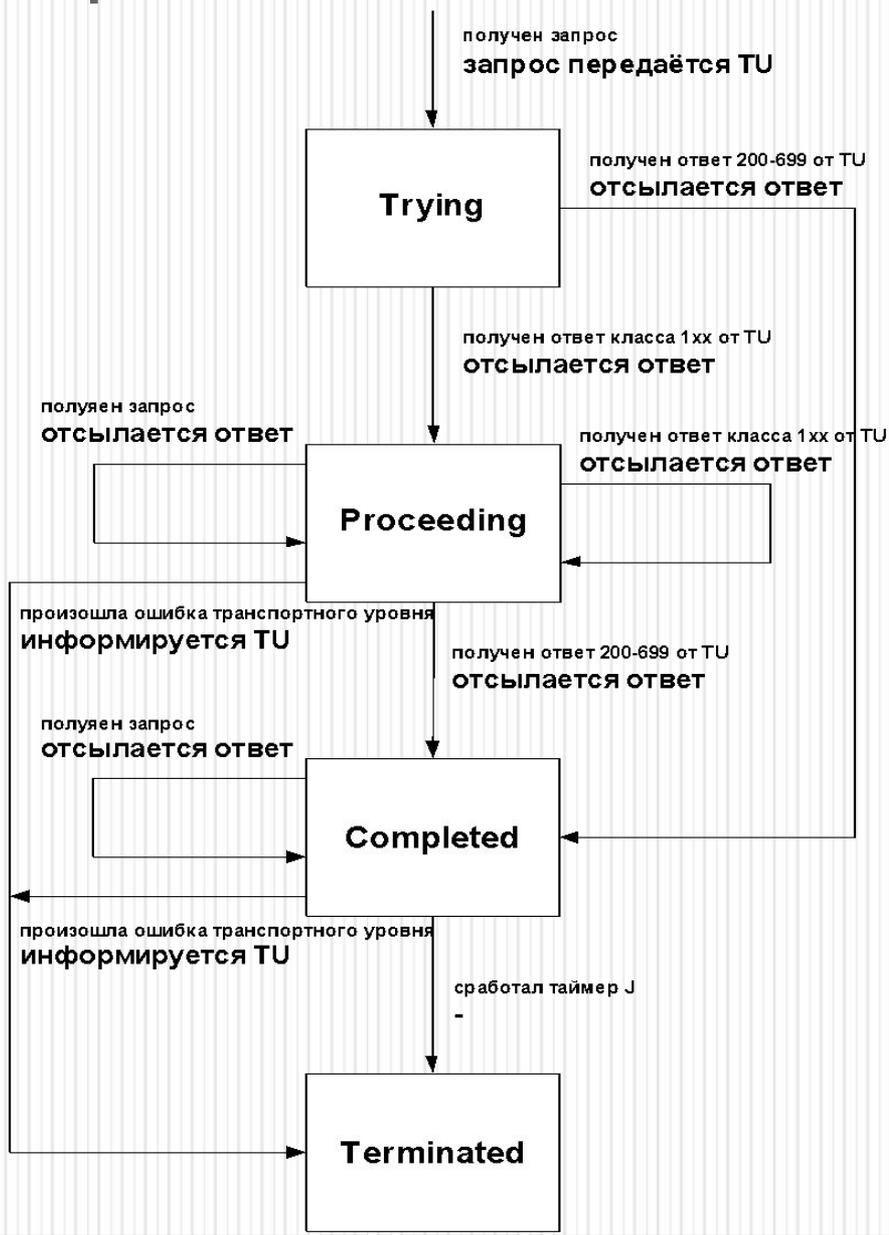
# Функционирование серверных транзакций

- Серверная транзакция отвечает за доставку запросов ТУ и надёжную передачу ответов.
- Серверные транзакции создаются при получении запросов при условии, что создание транзакции желаемо для запроса.
- Как в клиентских транзакциях механизм функционирования зависит от типа запроса.

# Серверная Invite-транзакция



# Серверная не-TCVCE-транзакция

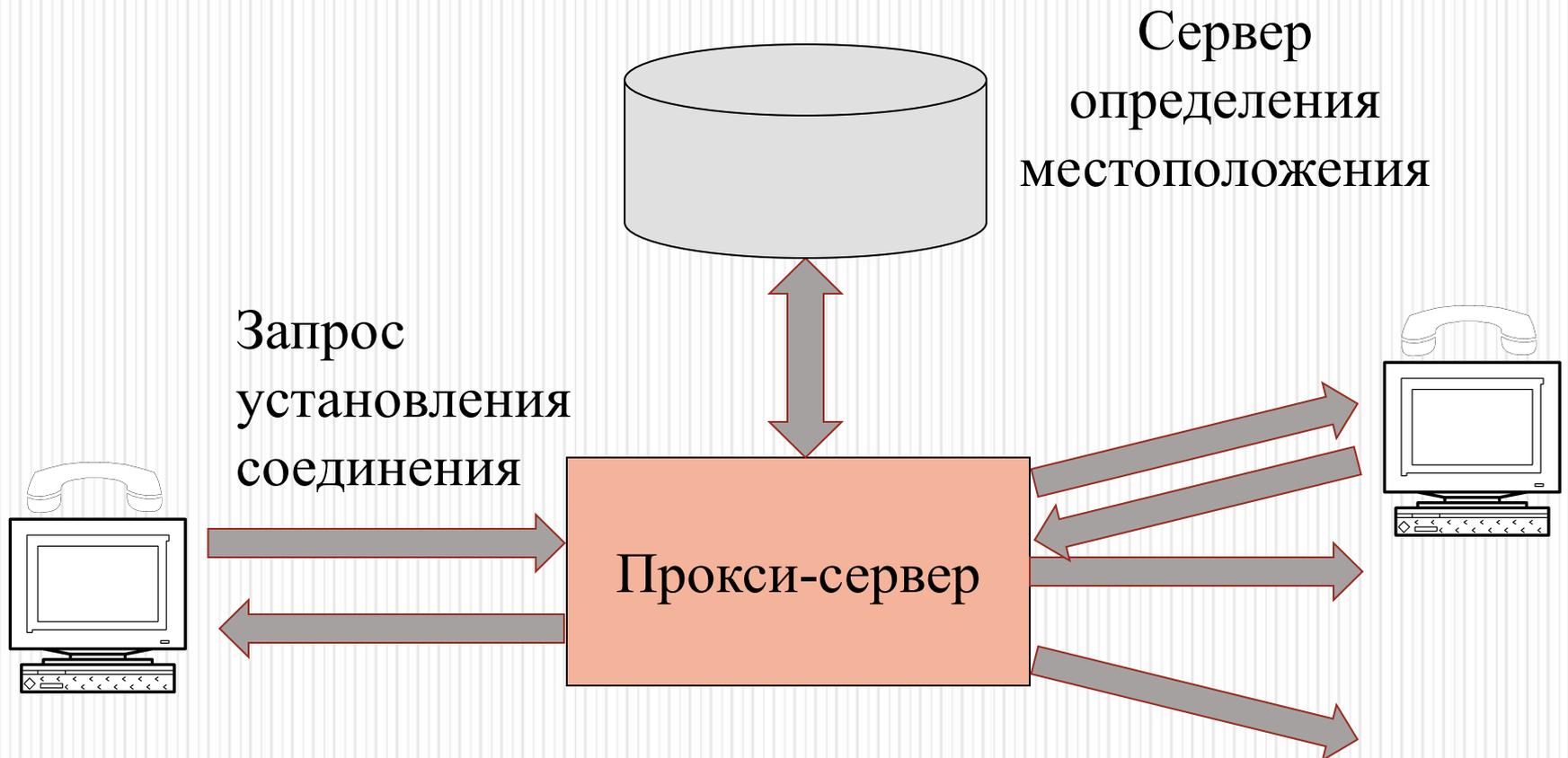


# Соответствие запросов серверным транзакциям

Запрос соответствует транзакции, когда:

- параметр «**branch**» запроса совпадает с аналогичным параметром в первом заголовке **Via** запроса, создавшего транзакцию.
- имя хоста и номер порта в первом значении заголовка **Via** совпадают с аналогичными в первом значении заголовка **Via** запроса, создавшего транзакцию.
- тип запроса соответствует типу запроса, создавшего транзакцию, за исключением запроса АСК, для которого тип запроса, создавшего транзакцию, будет **INVITE**.

# Прокси-сервер



# Прокси-сервер

- это элементы сети SIP, которые маршрутизируют SIP-запросы серверам агента пользователя и SIP-ответы – клиентам агента пользователя.
- Запрос может проходить несколько прокси-серверов на пути к UAS.
- Каждый из них будет принимать решения о дальнейшей маршрутизации, внося изменения в запрос перед его пересылкой следующему элементу сети.
- Ответы будут маршрутизироваться через ту же группу прокси-серверов, которая была пройдена запросами, но в обратном порядке.

# Прокси-сервер с сохранением состояний

- В режиме с сохранением состояний прокси-сервер действует как механизм обработки SIP-транзакций.
- При функционировании прокси-сервер задействует серверную транзакцию и одну или несколько клиентских транзакций
- Входящие запросы обрабатываются серверной транзакцией.
- От серверной транзакции запросы направляются ядру прокси-сервера.
- Ядро прокси-сервера определяет, куда маршрутизировать запрос, выбирая один или несколько мест назначения для следующей пересылки запроса.

# Прокси-сервер с сохранением состояний



**КТ - клиентская транзакция**

**СТ - серверная транзакция**

# Функции прокси-сервера с сохранением состояния

- Для каждого запроса элемент, выполняющий роль прокси-сервера, должен осуществлять следующие функции:
  - Проверка правильности составления запроса
  - Предварительная обработка маршрутной информации
  - Определение адреса (адресов) для пересылки
  - Пересылка запроса
  - Обработка всех ответов

# Сервер без сохранения состояния

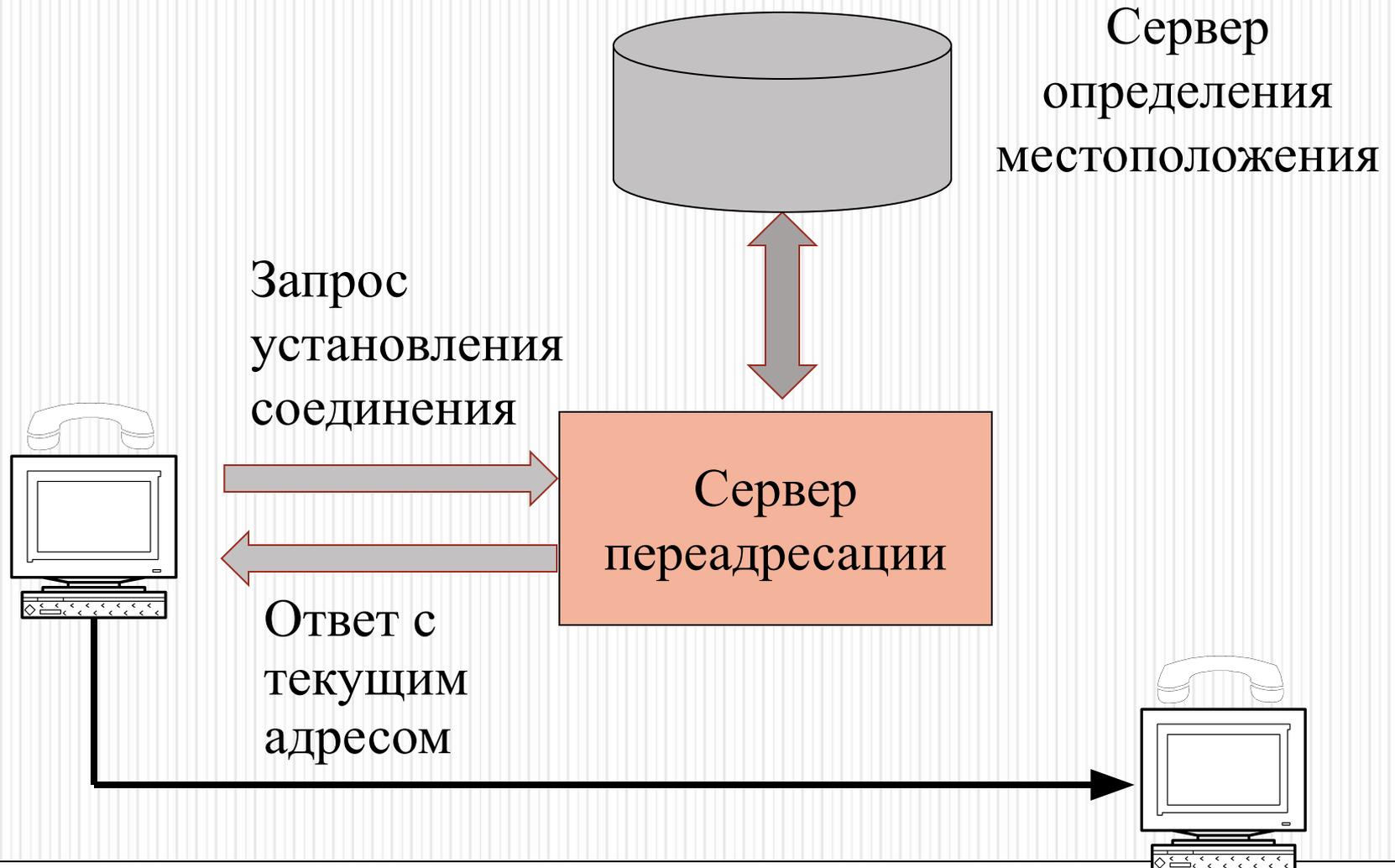
- Работает как ретранслирующий узел сети
- Он пересылает каждый запрос следующему элементу, принимая решения о маршрутизации исходя из информации, содержащейся в запросе.
- Полученные ответы он просто пересылает обратно.
- Он удаляет информацию о прошедшем сообщении, как только сообщение было ретранслировано.

# Сервер переадресации

**Сервер переадресации** предназначен для определения текущего адреса пользователя

- Не генерирует своих запросов
- Не терминирует вызовы
- Не содержит клиентскую часть программного обеспечения

# Сервер переадресации

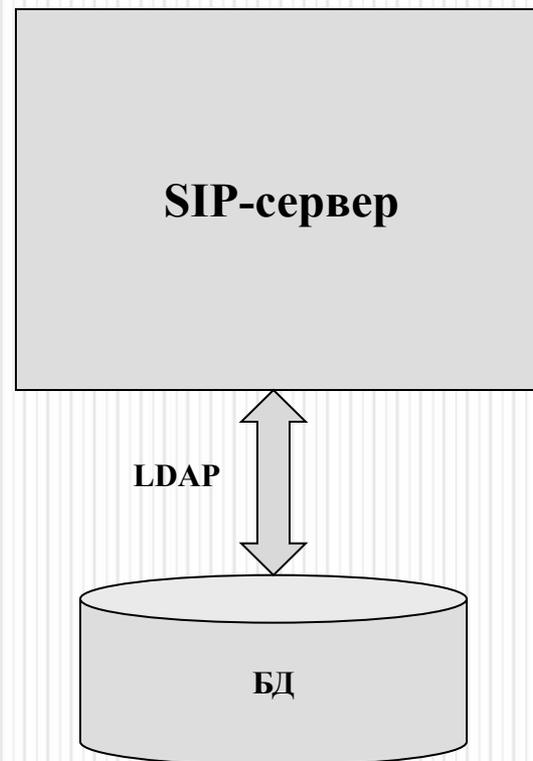
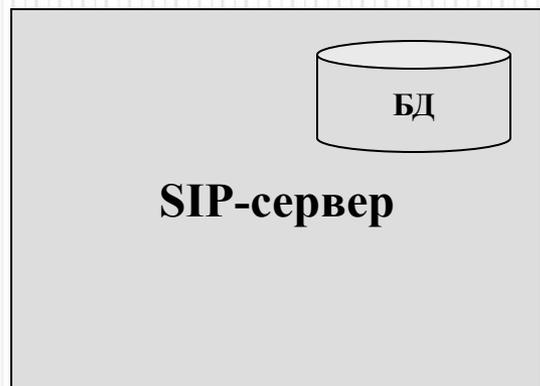


# Сервер определения местоположения

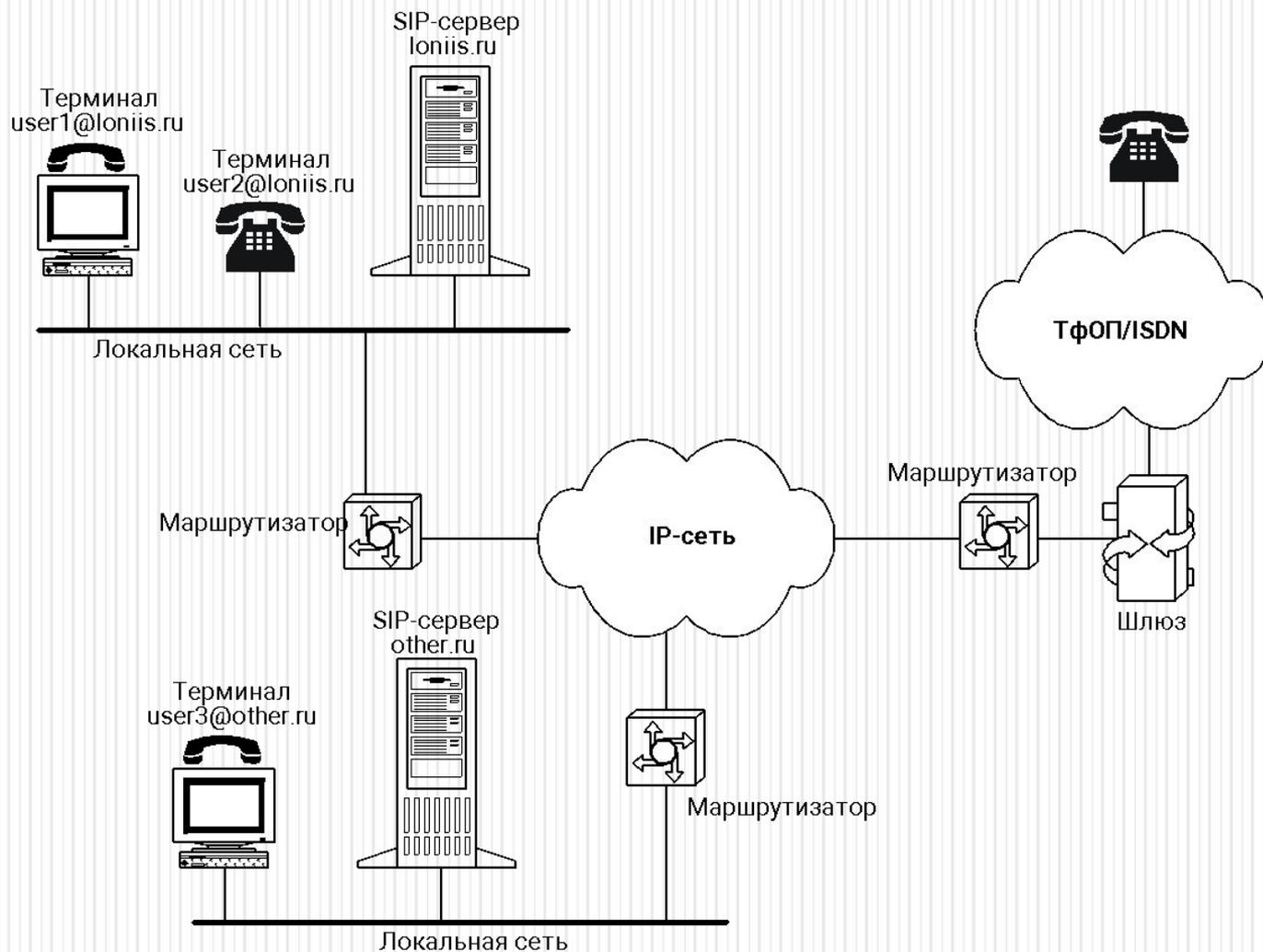
- Служит для хранения текущего адреса пользователя.
- Позволяет агентам регистрировать свое местоположение , обеспечивая тем самым мобильность пользователя
- Может быть совмещен с прокси-сервером
- О своем местоположении пользователь информирует сервер при помощи сообщения REGISTER. 2 режима регистрации:
  - Новый адрес сообщается один раз
  - Новый адрес сообщается через определенные промежутки времени

# Сервер определения местоположения

**Локальная    Удаленная**



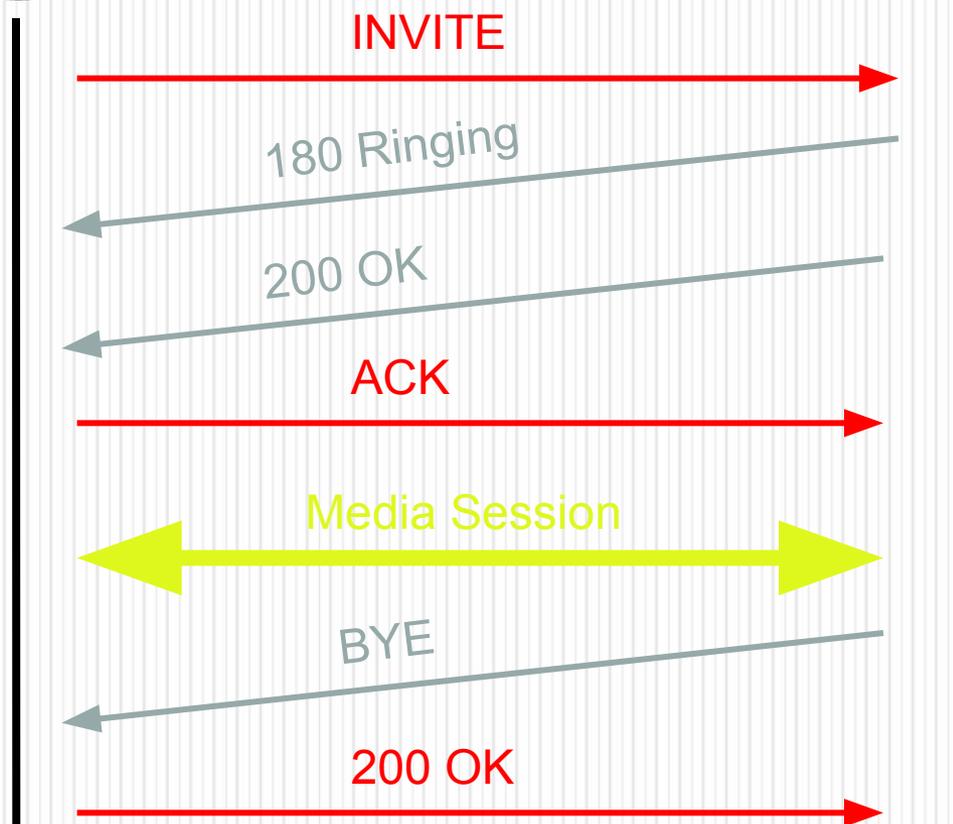
# Пример построения SIP-сети



Tesla



Marconi



Пример установления соединения в SIP сети



SUBSCRIBE

200 OK

NOTIFY

200 OK

⋮

NOTIFY

200 OK

MESSAGE

200 OK

MESSAGE

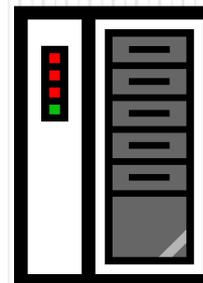
200 OK



Пример услуги «мгновенного обмена сообщениями»



Alice



Registrar Server



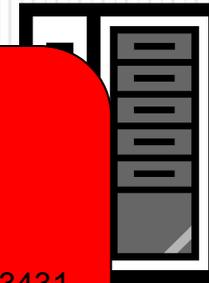
Пример регистрации в сети SIP

# Регистрация в сети SIP



Alice

```
REGISTER sip:registrar.udel.edu
Via: SIP/2.0/UDP 128.175.13.50:5060;
    branch=z9hG4bKus19
Max-Forwards: 70
To: Alice <sip:alice@eecis.udel.edu>
From: Alice <sip:alice@eecis.udel.edu>;tag=3431
Call-ID: 23@128.175.13.50
CSeq: 1 REGISTER
Contact: sip:alice@128.175.13.50
Content-Length: 0
```



Registrar Server

```
To: Alice <sip:alice@eecis.udel.edu>
From: Alice <sip:alice@eecis.udel.edu>;tag=3431
Call-ID: 23@128.175.13.50
CSeq: 1 REGISTER
Contact: <sip:alice@128.175.13.50>;expires=3600
Content-Length: 0
```

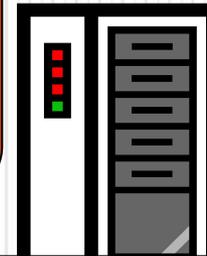


# SIP Redirect Server

INVITE sip:bob@yahoo.com SIP/2.0  
Via: SIP/2.0/UDP 100.101.102.103:5060  
branch=z9hG4bKmp17a  
To: Bob <sip:bob@yahoo.com>  
From: Alice <sip:alice@udel.edu>;tag=  
Subject: Where are you exactly?  
Contact: <sip:alice@pc.udel.edu>

ACK

SIP/2.0 302 Moved Temporarily  
Via: SIP/2.0/UDP 100.101.102.103:5060;  
branch=z9hG4bKmp17a  
To: Bob <sip:bob@yahoo.com>;tag=64  
From: Alice <sip:alice@udel.edu>;tag=42  
Subject: Where are you exactly?  
Contact: sip:alice@200.201.202.203

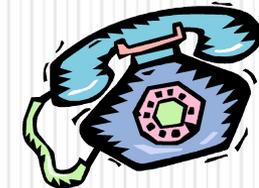


Redirect Server



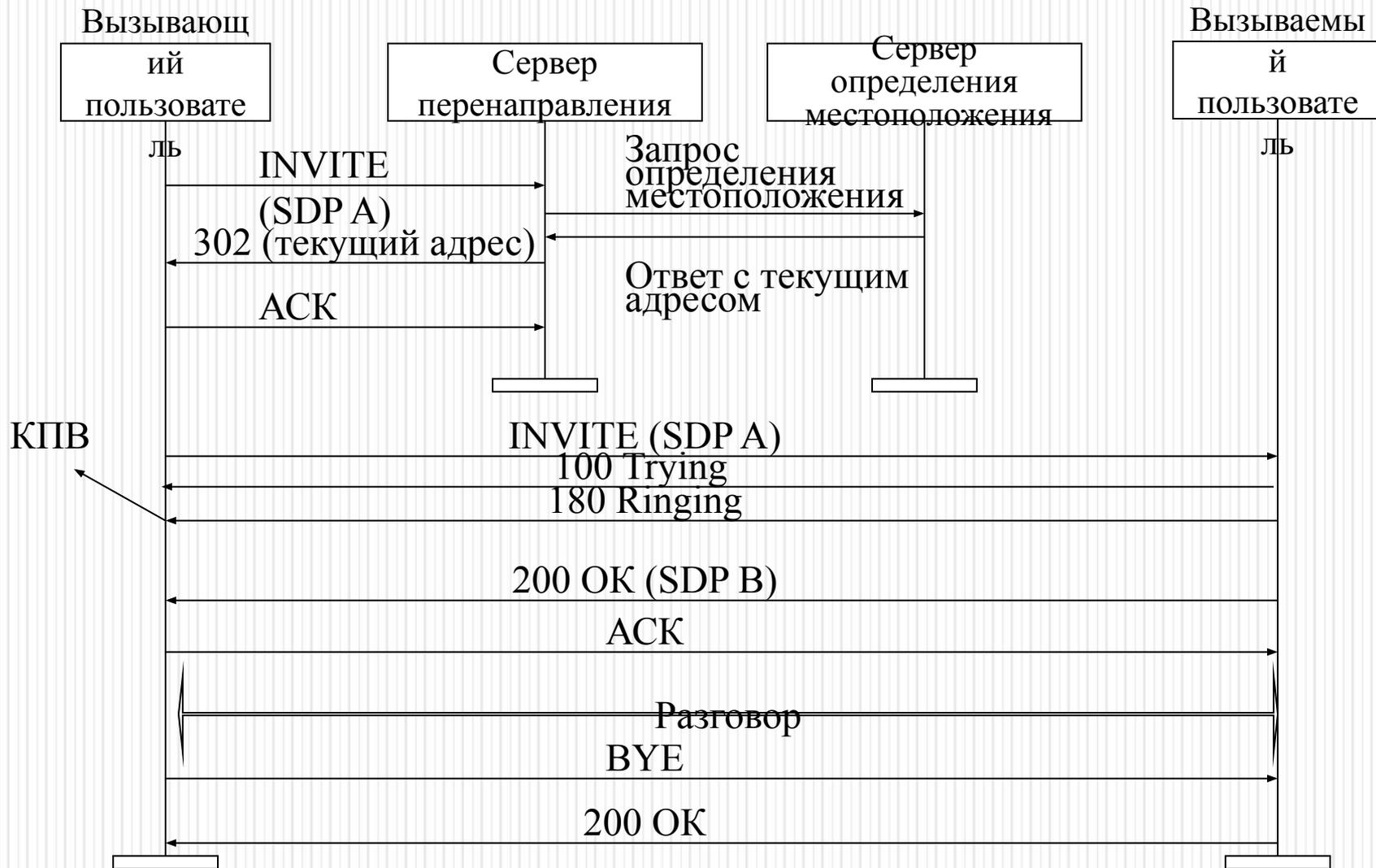
Alice

INVITE sip:bob@200.201.202.203 SIP/2.0  
Via: SIP/2.0/UDP 100.101.102.103:5060;  
branch=z9hG4bKmp17a  
To: Bob <sip:bob@yahoo.com>  
From: Alice <sip:alice@udel.edu>;tag=13473  
Subject: Where are you exactly?  
Contact: <sip:alice@pc.udel.edu>

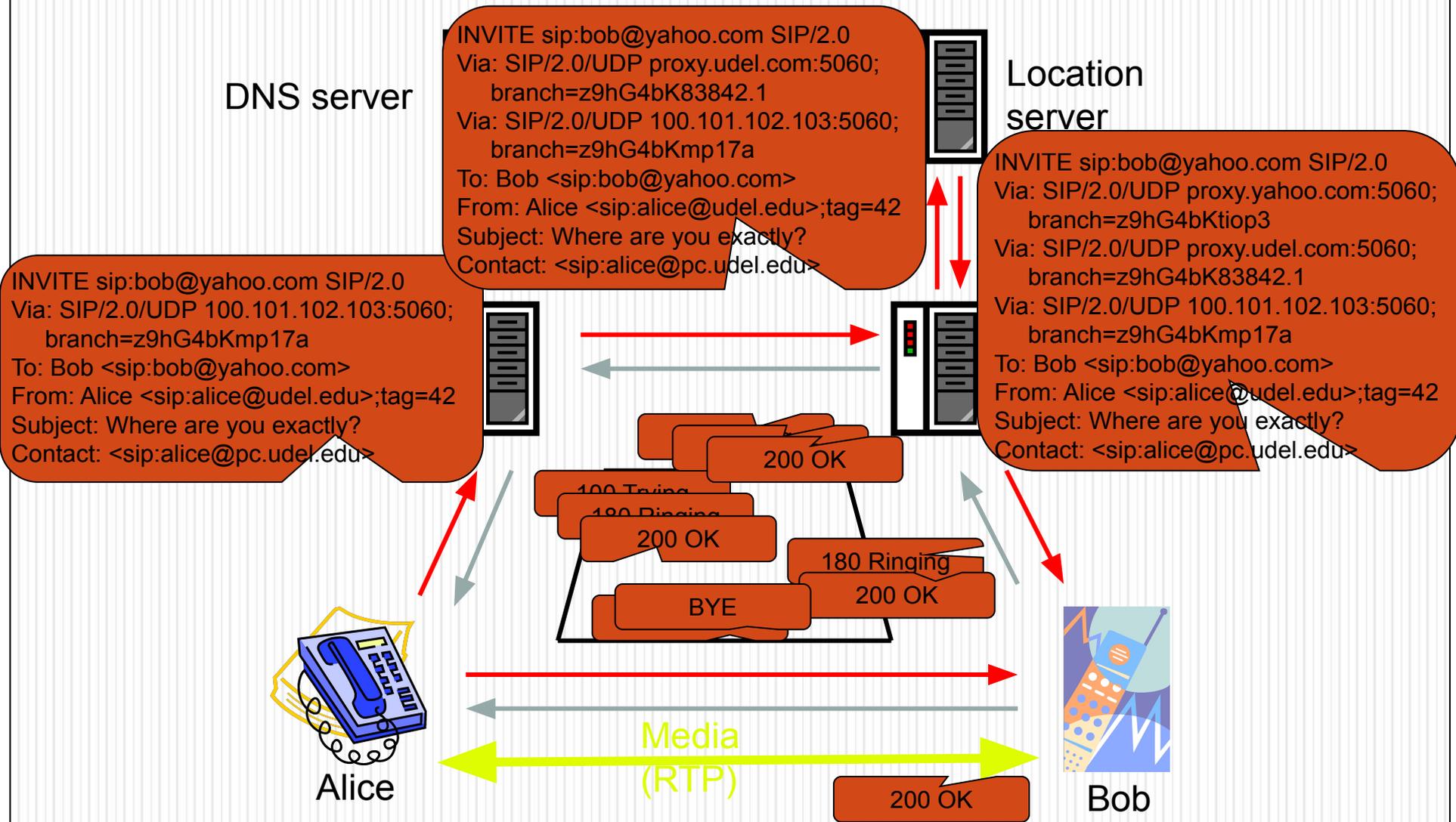


Bob

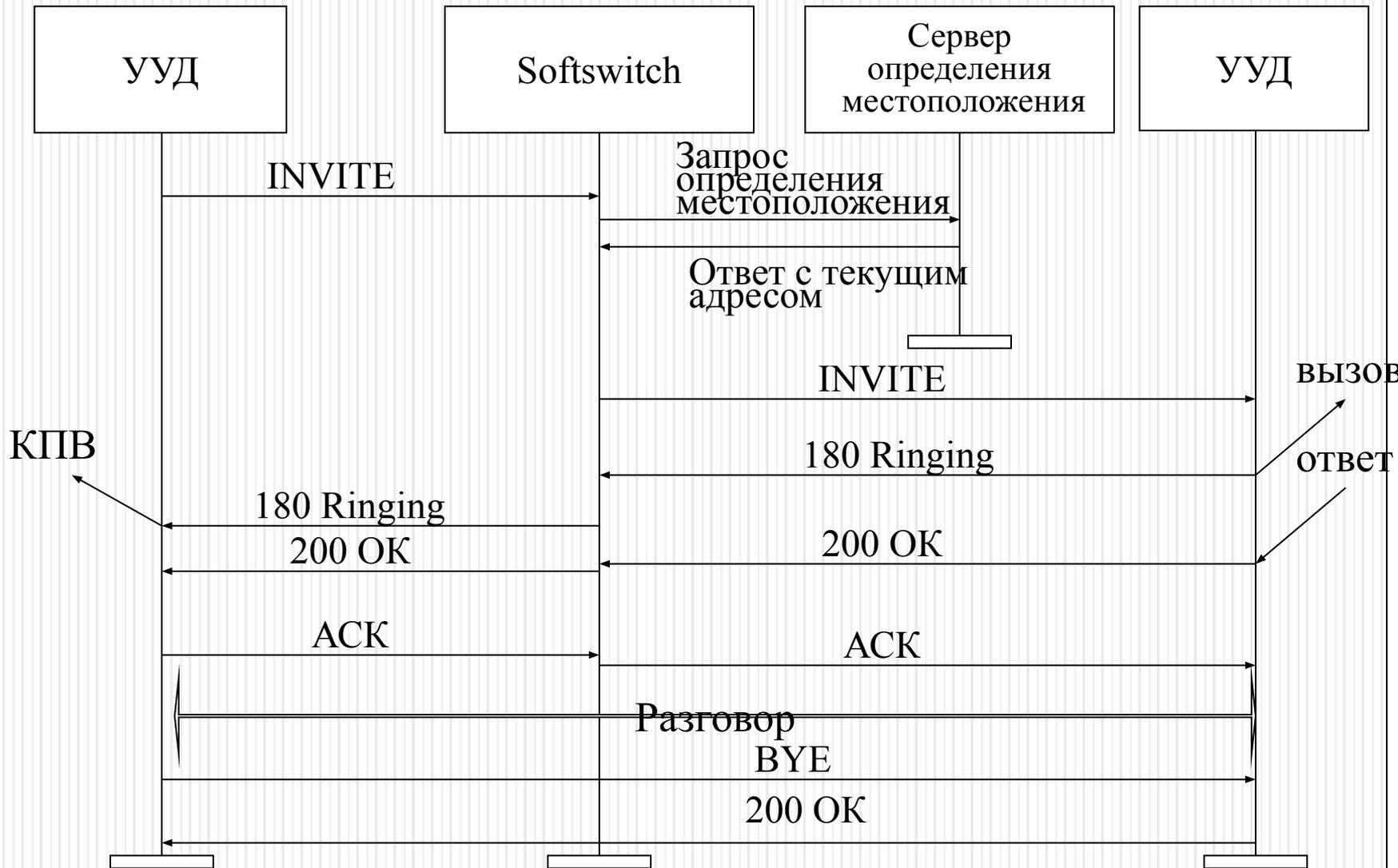
# Алгоритм работы сервера перенаправления



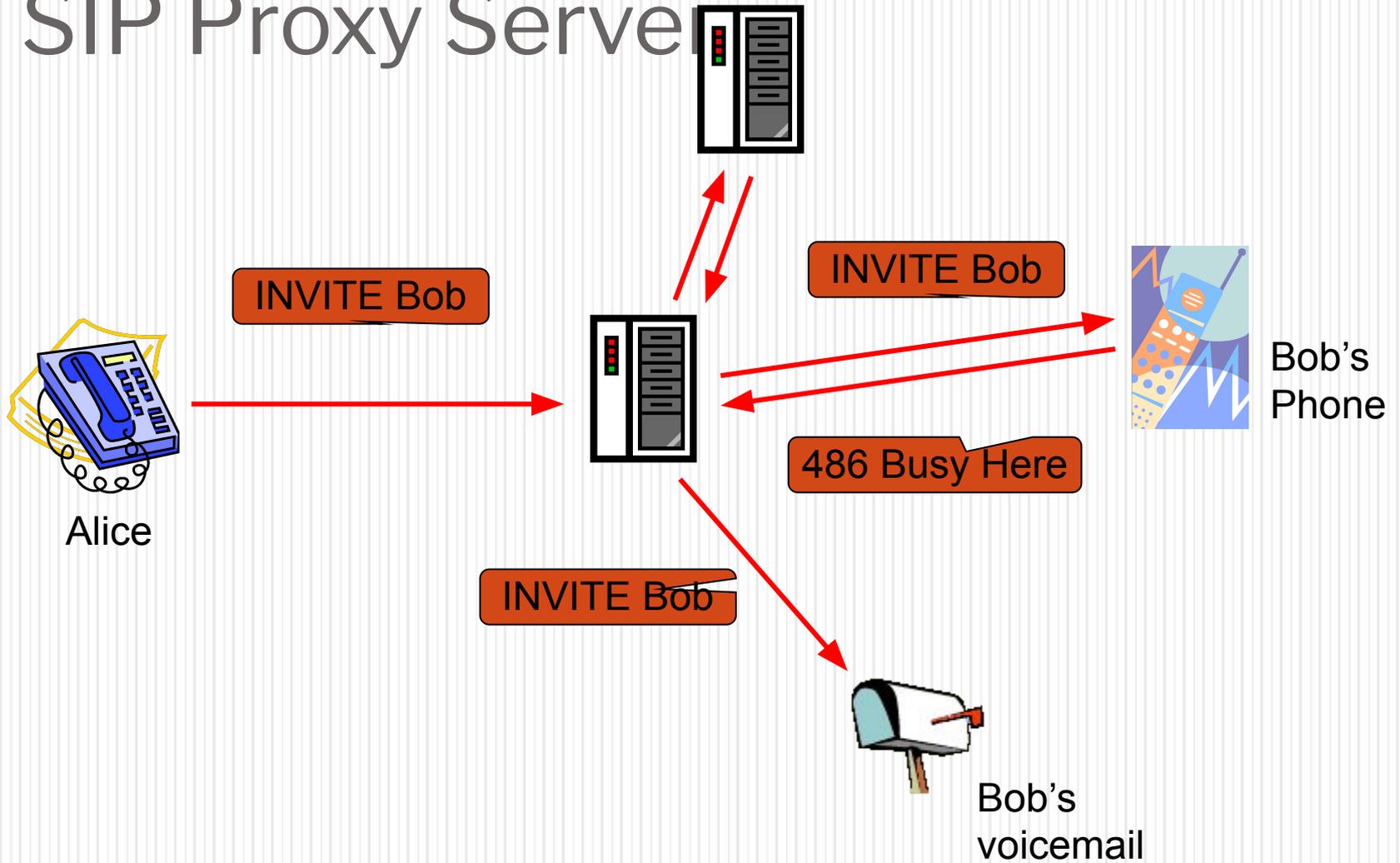
# Алгоритм работы прокси-сервера



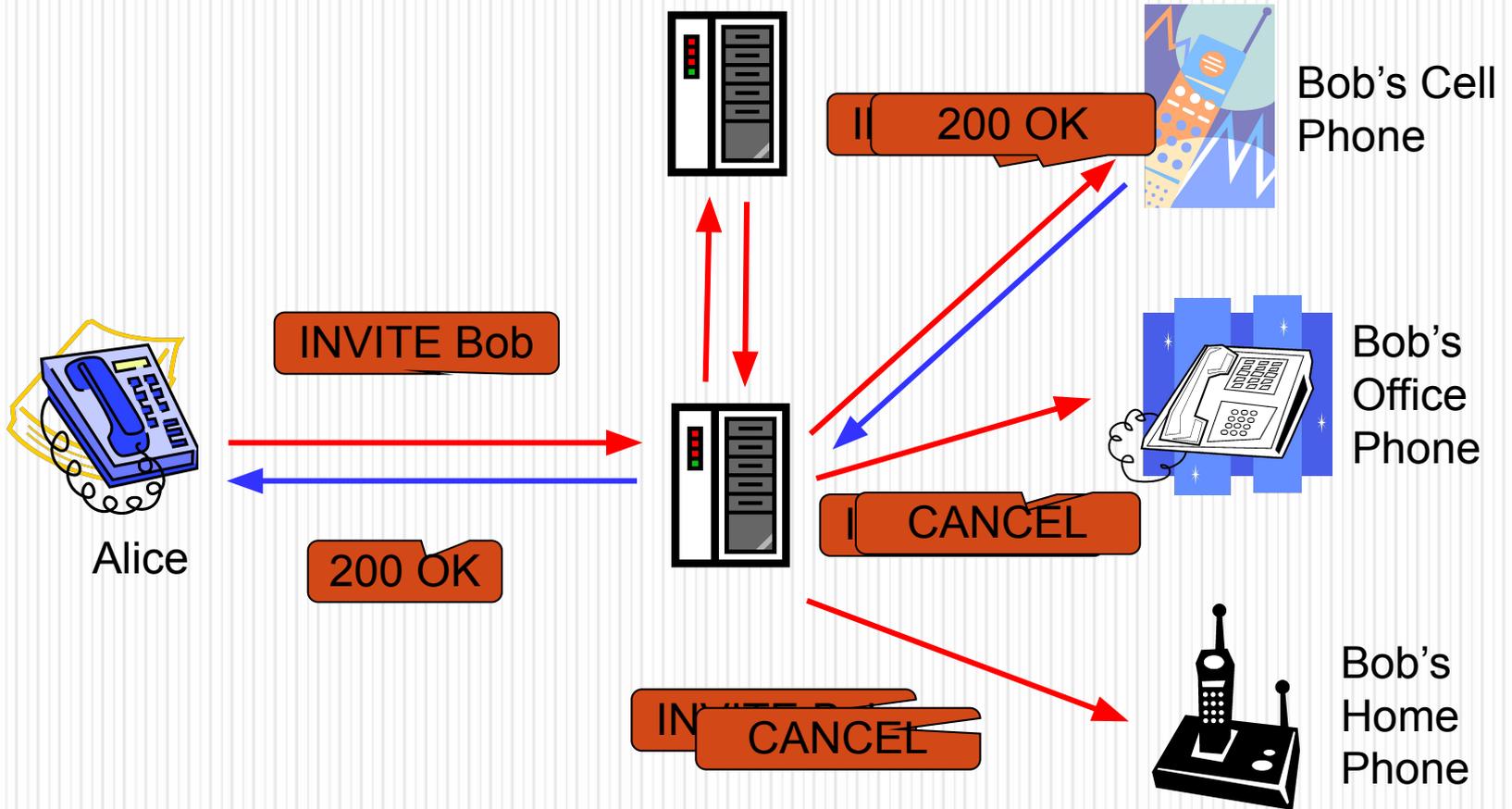
# Алгоритм работы прокси-сервера или Softswitch NGN



# SIP Proxy Server



# SIP Proxy Server



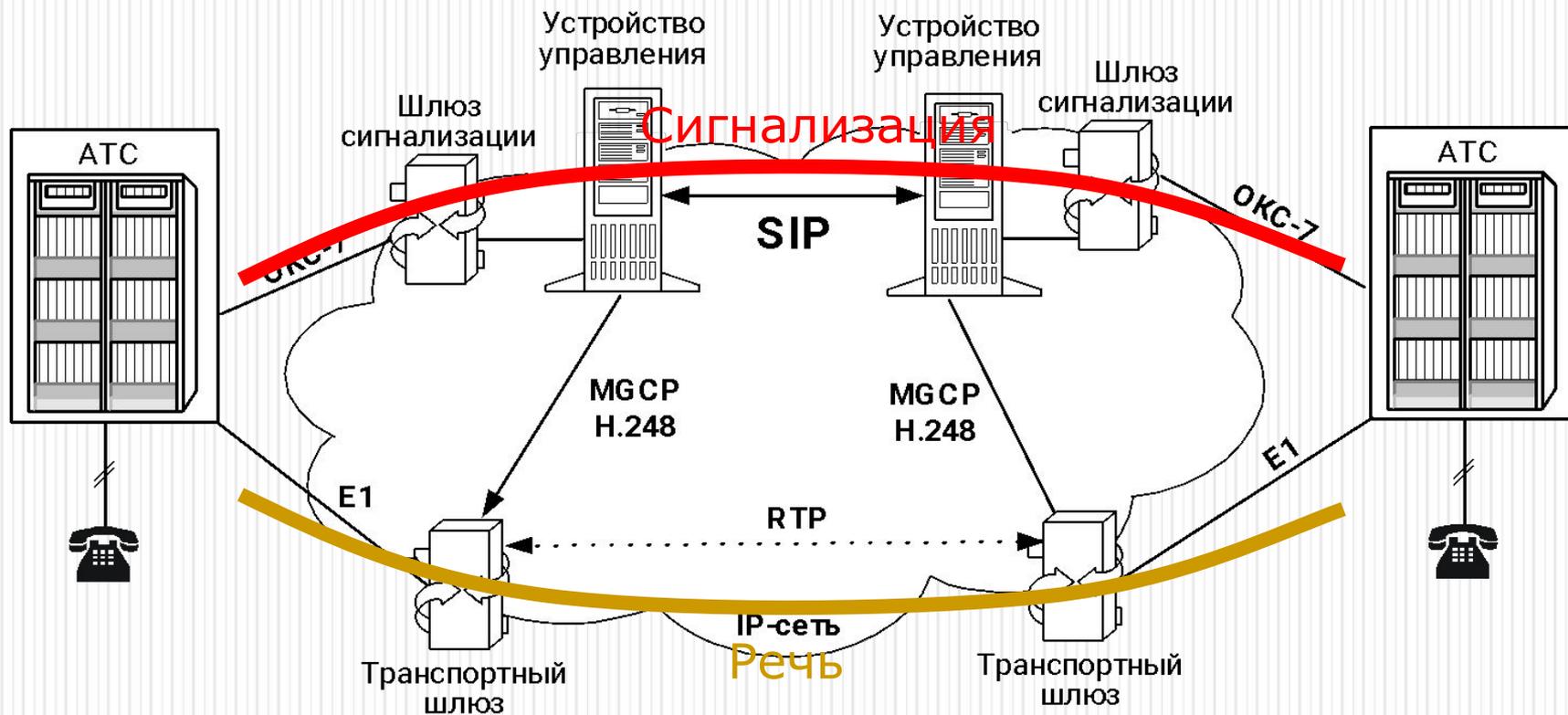
# Транспортный уровень протокола SIP

- Отвечает за перенос запросов и ответов через сеть с использованием ее транспортных протоколов
- Отвечает за управление соединениями таких протоколов как TSP и SCTR
- Имеет клиентскую и серверную стороны
- Соединение контролируется как на клиентской так и на серверной стороне

# Транспортный уровень протокола SIP

- Соединения идентифицируются указателем, состоящим из:
  - Адреса
  - Порта
  - Транспортного протокола на удаленном конце
- Соединение должно сохраняться в течение некоторого интервала времени после того, как последнее сообщение было передано или получено через это соединение

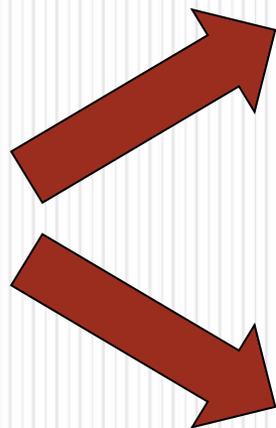
# Передача речи и команд управления



# SIP-T (SIP for Telephony)

Требование к сети IP-телефонии это возможность так называемой **прозрачности** услуг относительно ТфОП. Традиционные телефонные услуги, такие как call waiting, услуга 800 и т.д. реализуются с помощью системы сигнализации №7.

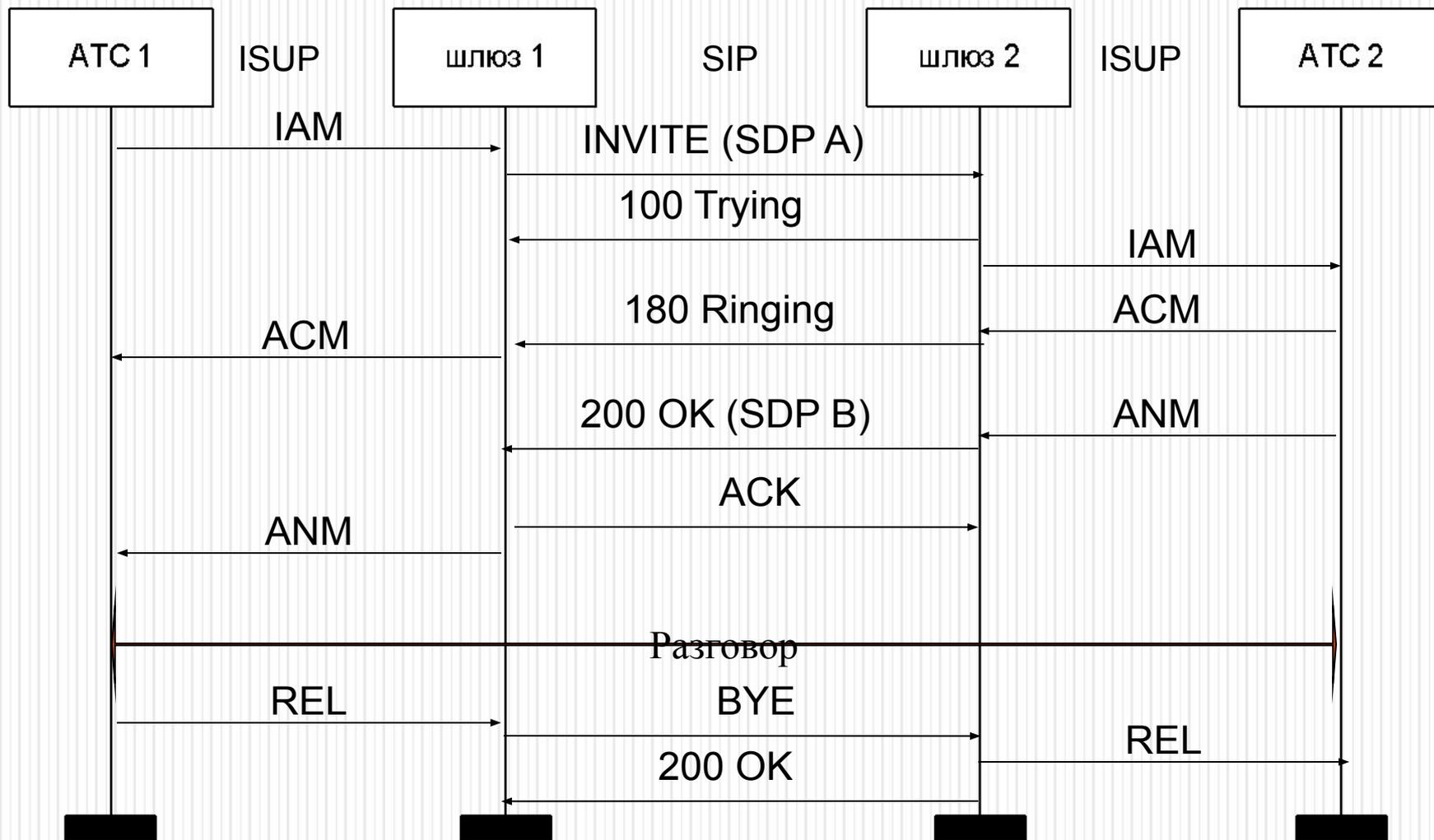
SIP-T



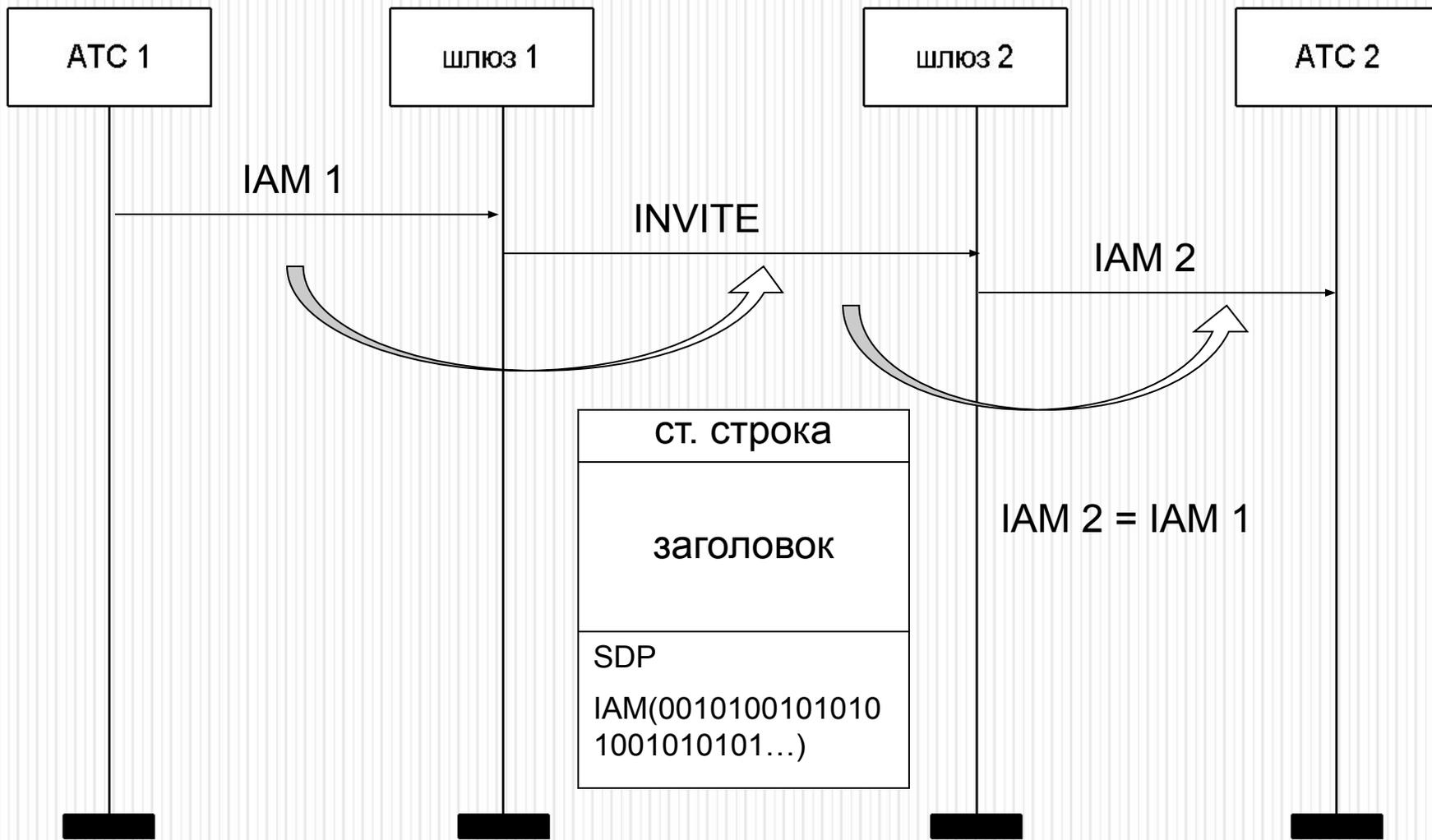
Инкапсуляция сообщений ОКС7/DSS-1 в сообщения SIP

Использование информации из сообщений ОКС7/DSS-1

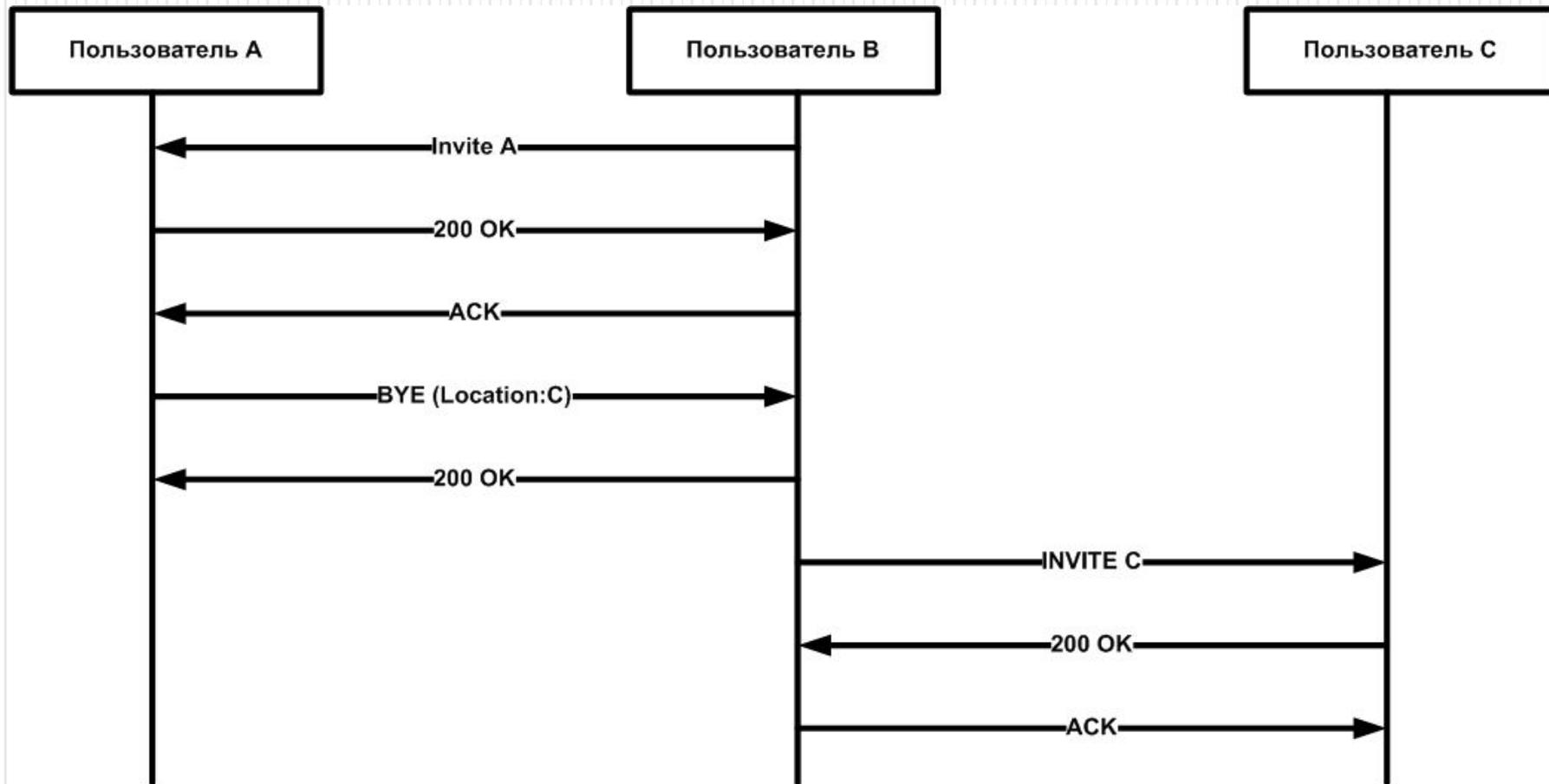
# Взаимодействие с ТфОП



# Инкапсуляция

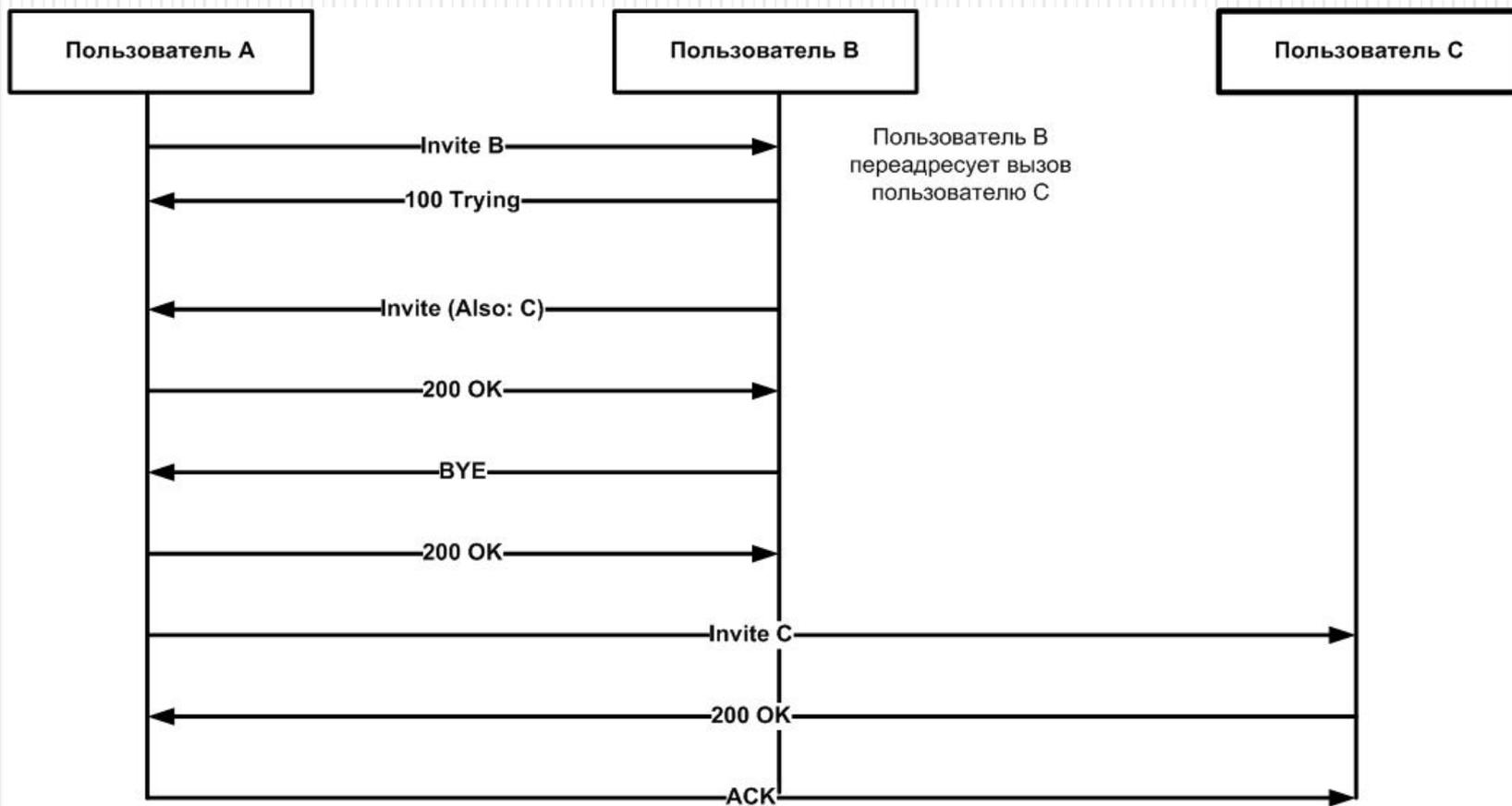


# Дополнительные услуги:



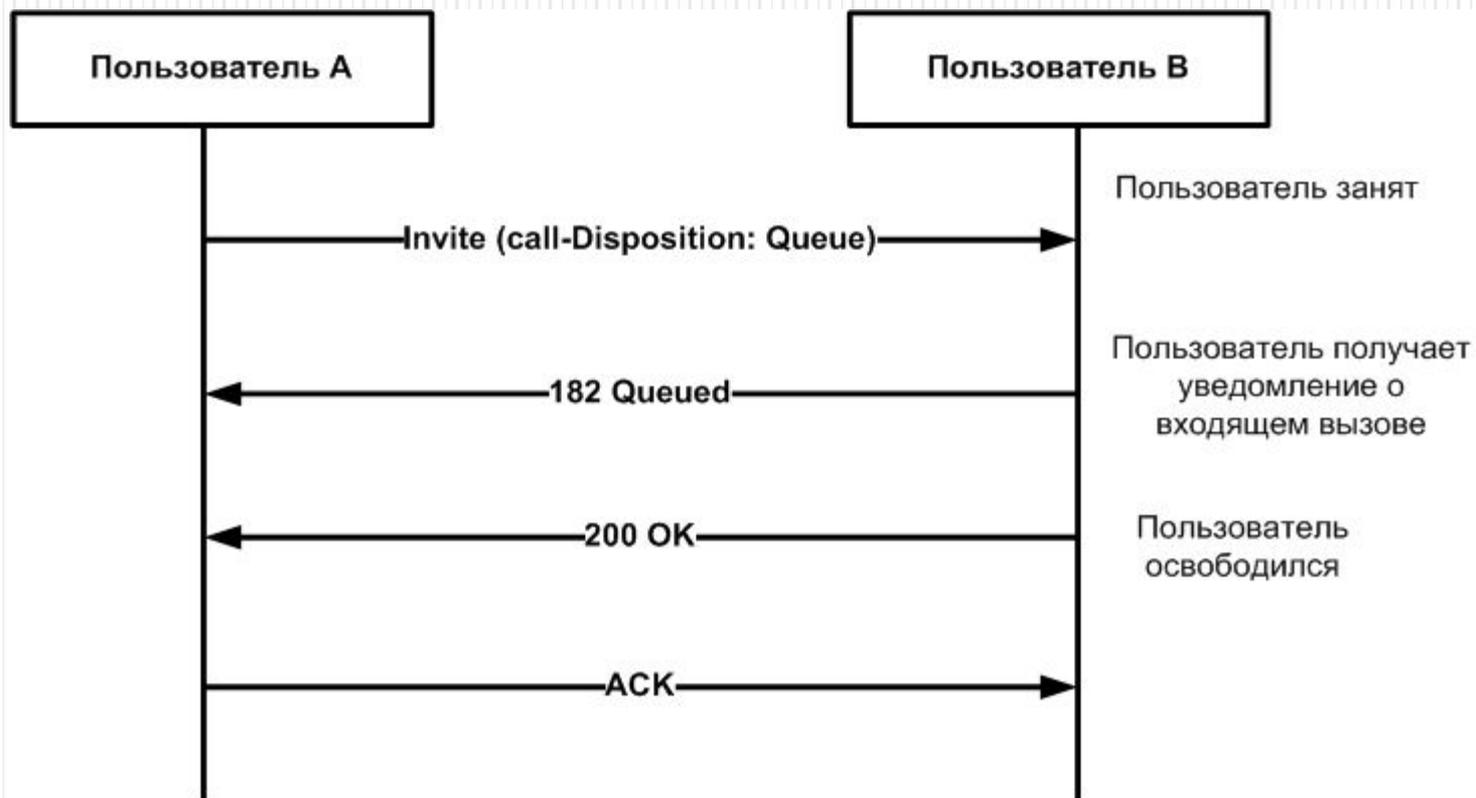
- Услуга «Переключение связи»

# Дополнительные услуги:



- Услуга «Переадресация вызова»

# Дополнительные услуги:



- Услуга «Уведомление о вызове во время связи»

# Применения SIP

- 1. Сотовые сети нового поколения 3G**
- 2. SIP для установления мультимедийных сессий**
- 3. SIP for Telephony (SIP-T)**

# Заключение

- SIP – перспективный современный подход к построению сетей IP-телефонии
- SIP – удобный и простой для реализации и техобслуживания
- SIP легко интегрируем в существующий стек протоколов Интернет
- SIP выбран в качестве протокола установления соединения в сотовых сетях поколения 3G
- SIP начинает использоваться в программных продуктах современных компаний (напр., Microsoft)
- SIP стремятся поддержать почти все производители оборудования IP-телефонии (Cisco, 3Com, Alcatel-Lucent)