Indoor Location for NG-9-1-1

Carol Davids	davids@iit.edu
Javier Moreno	jmorenov@hawk.iit.edu
Cruz Tovar	ctovar@hawk.iit.edu
Bartlomiej Dworak	bdworak@hawk.iit.edu

Introduction

- The goal of this project is to design, build and demonstrate a service that automatically provides the indoor location of a call that originates from a mobile device and is destined for a Public Service answering Point (PSAP).
- The inspiration for this project came from the "Roadmap for Improving E911 Location Accuracy," written by APCO, NENA and four national wireless carriers in response to the FCC's Third Notice of Proposed Rulemaking on this topic.

Demo

A team of three IIT graduate students worked for one semester to develop the POC system.

http://1drv.ms/1zWmBCR

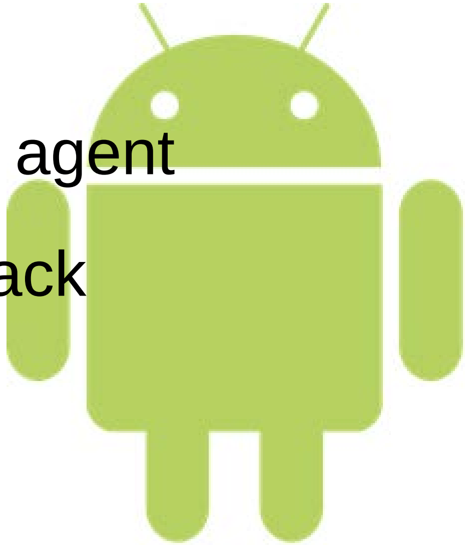
System Components

The system has three major components:

- A mobile Android application
- A commercial session border controller, donated by Oracle
- The WiFi Access Points and their Management System administered by the school's IT department
- The Next Generation 9-1-1 test bed in the IIT RTC Lab, including the PSAP where the call is received and the indoor location is displayed.

Android Phone

- Android app based on ***Sipdroid***
    - Open source
    - Provides all the features of a user agent
    - Allows modifications of the SIP stack
        - Custom call parameters
        - Custom SIP messages



Location Server

- The Location Server is a Linux server running custom software to determine the user’s location.

- The Android phone submits it’s MAC address to the location server and the server provides the location back to the phone via the HTTP call.

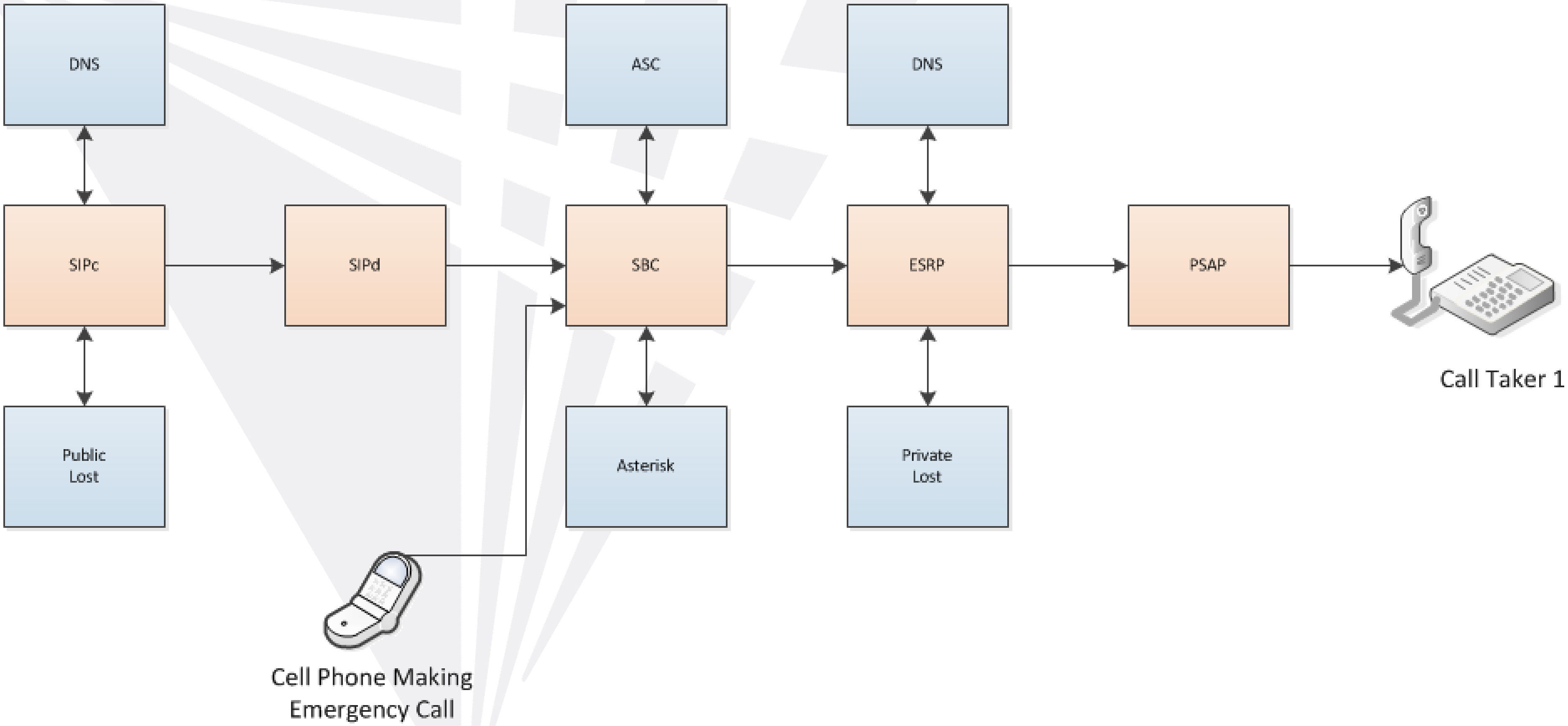
Session Border Controller

- The SBC protects the ESINet on the NENA i3 architecture.Our SBC was donated by Oracle.

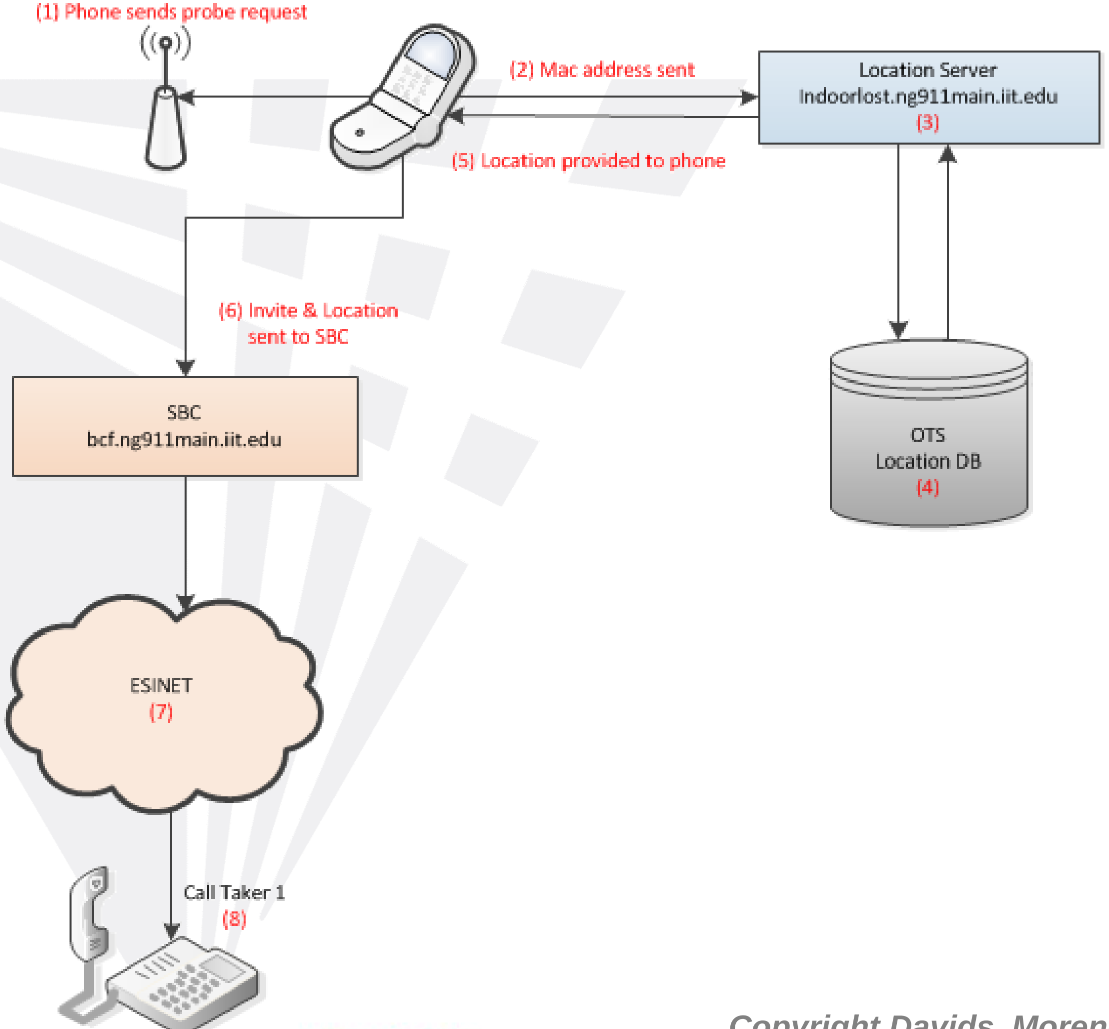
- Currently, the SIP phone knows the location of the SBC and the SBC knows the identity of the SIP phone.

- This is a design issue that needs development. *

NG9-1-1 Network Elements



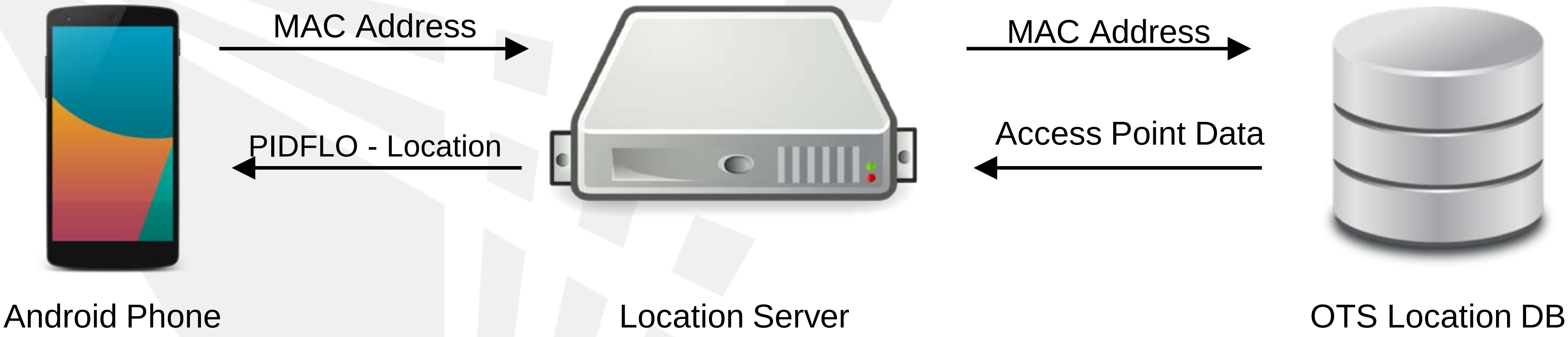
Call Flow / Process



Android Phone makes the call

- Caller dials 9-1-1
- Sipdroid recognizes emergency number
- Sipdroid gets phone MAC address
    - Checks whether WiFi is enabled first
- MAC address is sent to location server

Android Phone asks for Location

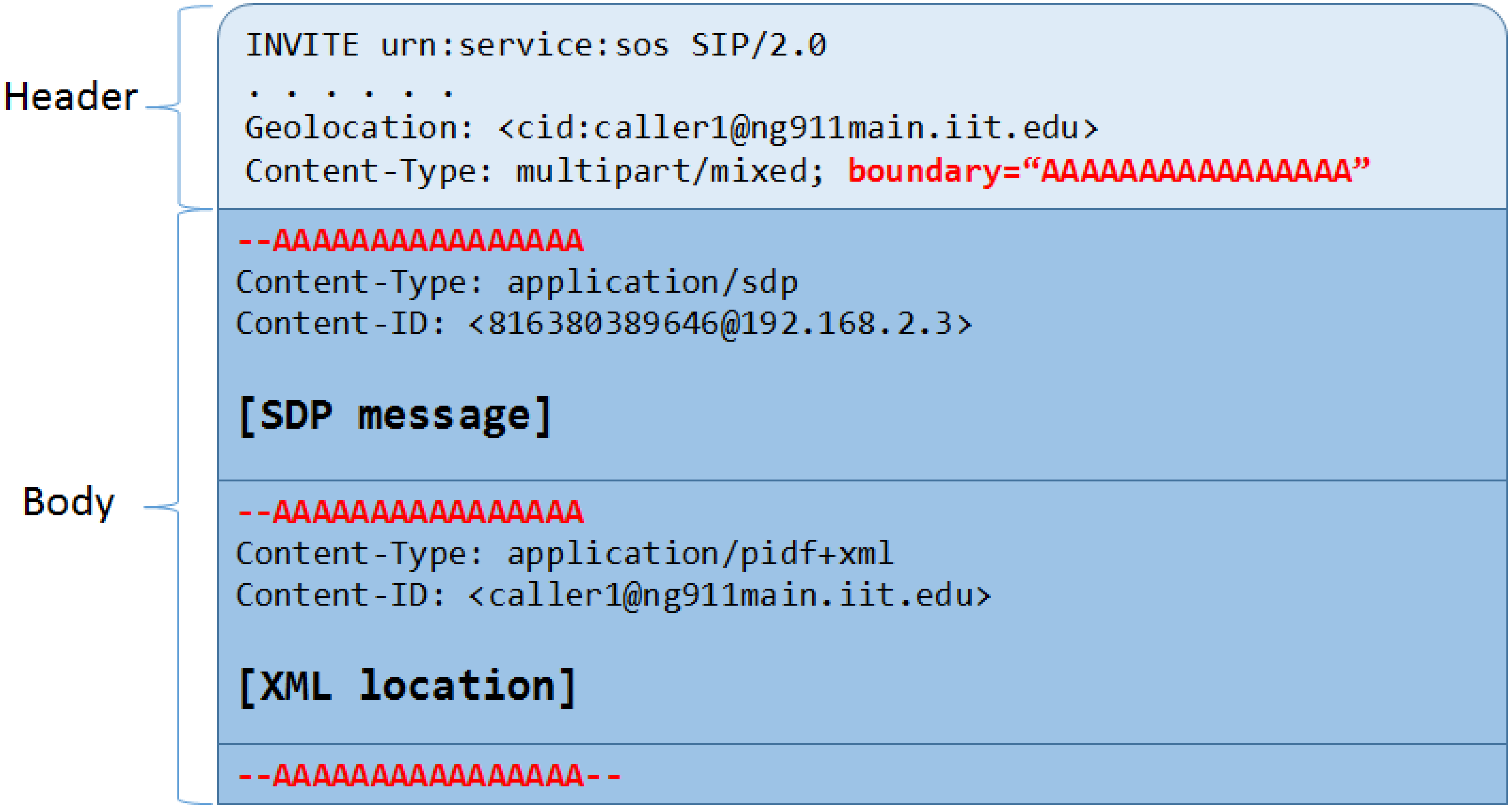


- In this step the Android phone sends its MAC address to the location server and the location server searches the OTS database and attempts to do a reverse lookup of the phone’s location.
- The OTS server returns a list of access points that have discovered the phone and the access point to which the phone is associated to if the phone is connect to wifi.

INVITE is sent to SBC

- Android phone receives location information in XML format (PIDF-LO)
- PIDF-LO: Presence Information Data Format Location Object [RFC5139]
- Sipdroid embeds location in SIP INVITE message
- How to do this?
  - `Geolocation:` header field
  - MIME multipart message body

INVITE message

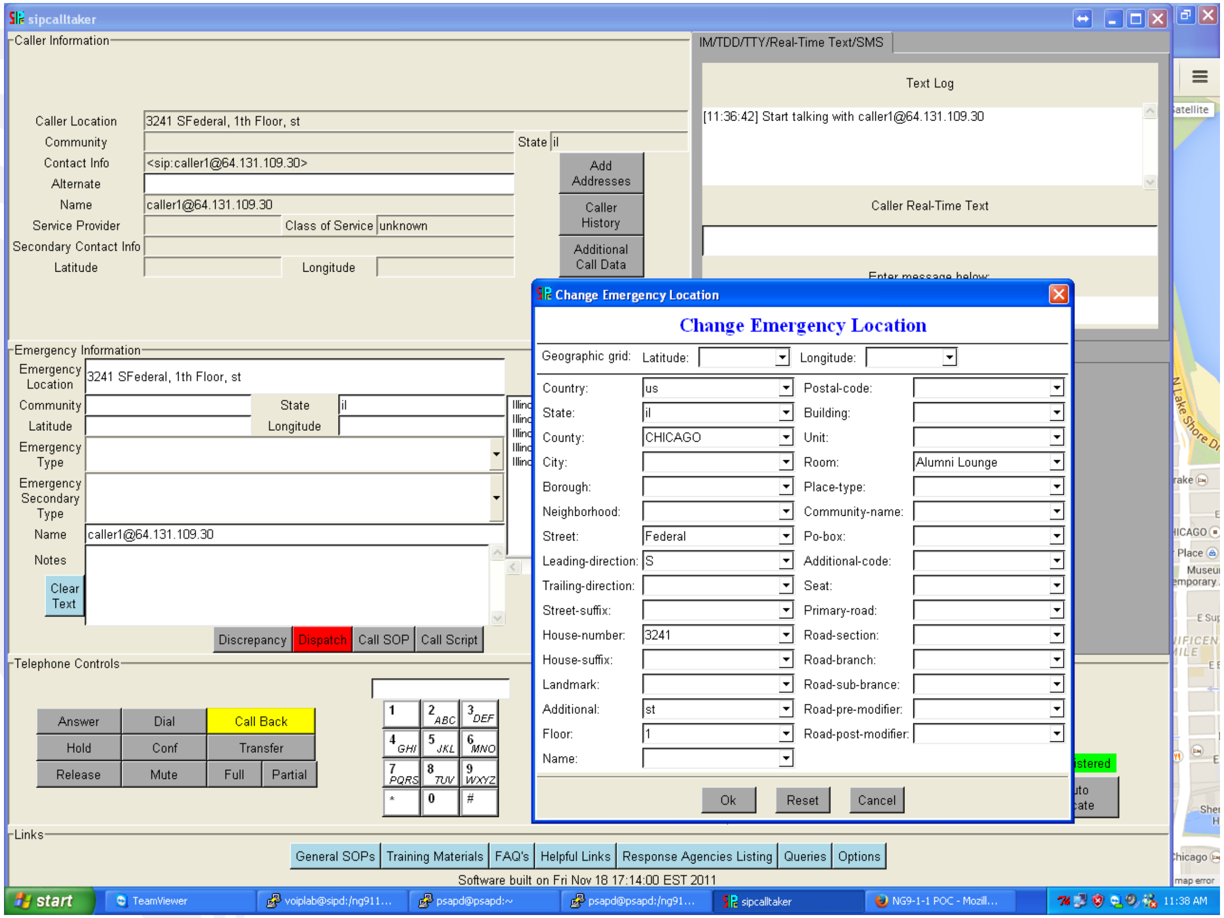


```
INVITE urn:service:sos SIP/2.0
. . . . . .
Geolocation: <cid:caller1@ng911main.iit.edu>
Content-Type: multipart/mixed; boundary="AAAAAAAAAAAAAAAA"
--AAAAAAAAAAAAAAAA
Content-Type: application/sdp
Content-ID: <816380389646@192.168.2.3>

[SDP message]
--AAAAAAAAAAAAAAAA
Content-Type: application/pidf+xml
Content-ID: <caller1@ng911main.iit.edu>

[XML location]
--AAAAAAAAAAAAAAAA--
```

Call Taker receives location information



sipcalltaker

Caller Information

Caller Location	3241 SFederal, 1th Floor, st
Community	
State	il
Contact Info	<sip:caller1@64.131.109.30>
Alternate	
Name	caller1@64.131.109.30
Service Provider	
Class of Service	unknown
Secondary Contact Info	
Latitude	
Longitude	

IM/TDD/TTY/Real-Time Text/SMS

Text Log

[11:36:42] Start talking with caller1@64.131.109.30

Caller Real-Time Text

Enter message below:

Emergency Information

Emergency Location	3241 SFederal, 1th Floor, st
Community	
State	il
Latitude	
Longitude	
Emergency Type	
Emergency Secondary Type	
Name	caller1@64.131.109.30
Notes	

Change Emergency Location

Change Emergency Location

Geographic grid: Latitude: Longitude:

Country:	us	Postal-code:	
State:	il	Building:	
County:	CHICAGO	Unit:	
City:		Room:	Alumni Lounge
Borough:		Place-type:	
Neighborhood:		Community-name:	
Street:	Federal	Po-box:	
Leading-direction:	S	Additional-code:	
Trailing-direction:		Seat:	
Street-suffix:		Primary-road:	
House-number:	3241	Road-section:	
House-suffix:		Road-branch:	
Landmark:		Road-sub-brance:	
Additional:	st	Road-pre-modifier:	
Floor:	1	Road-post-modifier:	
Name:			

Software built on Fri Nov 18 17:14:00 EST 2011

Challenges and Choices

- Blue Tooth beacons are another option. We selected the WiFi beacons because they are already part of the building's infrastructure.
- The AP management system only collects information every ~60 seconds. This can lead to stale location information.
- Our querying server must not compromise the security of the AP management system.
- Selection of the Android SIP Client is a challenge. Many did not allow us to add MIME body and the geolocation header to the SIP INVITE.
- Public proxy function must be able to pass the MIME body. Asterisk did not. As a B2BUA, it removed our MIME body and replaced it with its own SDP body.

Next Steps include…

- Second demo planned for the end of July 2015
- System to be working throughout the campus
- Beta test with the Campus Security Department
- Augment the location algorithm
- Improve update speed of the WiFi management system
- Add commercial PSAP and other commercial elements to the RTC Lab ESINet
- Register the Android app to Kamailio or other Proxy.

Thank you!

Questions/Comments

?