

# RCS-e open source stack



## Roadmap

Date: 21/05/2012

Author: Orange Labs, ASC Devices



Orange Labs



# History...

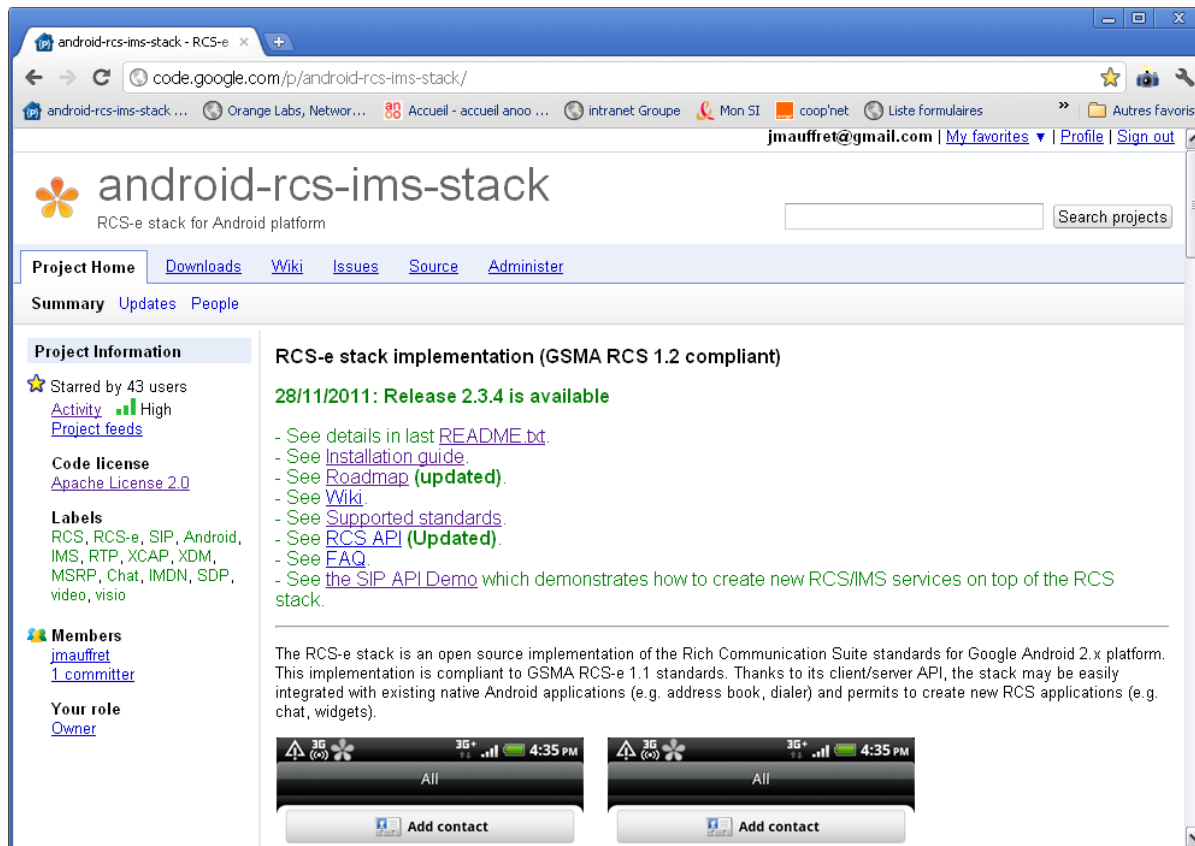
- 2007 / First RCS stack for J2ME platform.
- 2008-2009 / RCS 1.0 stack:
  - GSMA IOT: Helsinki (2008) / First participation.
  - GSMA IOT: Paris (2009).
  - GSMA IOT: Munich (2009).
- Jan. 2010 / First RCS stack for Android platform.
- 2010 / RCS 2.0 stack:
  - GSMA IOT: Paris (2010).
  - GSMA IOT: Beijing (2010).
  - GSMA IOT: Madrid (2010).
- Jan. 2011 / RCS 2.0 stack:
  - GSMA IOT: Helsinki (2011).
  - OrangeLabs stack is taken as Reference Implementation by GSMA.
- Feb. 2011 / Integrate the same SIP stack as Google Android 2.3.
- March 2011 / RCS-e stack implementation based on 2.0 stack started.
- April 2011 / First RCS-e 1.1 release available.
- Mai 2011: RCS-e stack published to open source community.
- End of 2011: native integration by several device manufacturers.



# Open source project

## ■ Open source published at:

<http://code.google.com/p/android-rcs-ims-stack/>



# Roadmaps



# Roadmap 2011

## ■ April 18: Sprint 0

- First RCS-e release.

## ■ May 02: Sprint 1

- Update capabilities in case of error, end of session.
- Adapt capabilities thanks to network coverage.
- Registration management in case of failure.
- H264 software codec (from Android opencore package).
- Visio share from 2 individual video share sessions.

## ■ May 15: Sprint 2

- IMDN and message delivery report (“delivered”, “displayed”).
- Add 1/N participants.
- Close a chat session after an inactivity timeout.
- OTA provisioning with WAP PUSH event and direct HTTP download of XML config.

## ■ May 30: Sprint 3

- RTCP event management.
- H264 packetizer.
- Store & Forward procedure management via WAP PUSH event.



# Roadmap 2011

## ■ June 15: Sprint 4

- Session refresh management (RFC 4028).
- Spam folder for message received from blocked contacts.
- Use an alias (i.e. display name) to initiate sessions.
- RCS API permission checking.
- Dynamic discovery of RCS extensions.
- SIP keep alive if a NAT is auto detected.

## ■ June 30: Sprint 5

- Capabilities discovery enhancement to support large address book.
- SIPS (SIP over TLS).
- Refactoring of the contacts database provider (use of batch procedures).
- Android makefile and RCS API jar file.

## ■ July 30: Sprint 6

- NAT traversal support (RFC6135, draft-ietf-simple-msrp-sessmatch-11).
- GRUU for multidevice support.
- Block FT if contact is blocked.
- New methods in chat API.
- RCS widget.



# Roadmap 2011

## ■ August 15: Sprint 7

- RCS API minor updates.
- New call flow to add a participant in a 1-1 chat (move from a 1-1 session to a 1-N session).
- Generic SIP API to open the stack for new IMS services implementation.

## ■ August 31: Sprint 8

- Registration retry mechanism.
- Bug corrections.

## ■ September 15: Sprint 9

- Bug corrections.
- RCS API evolutions.

## ■ October 03: Sprint 10

- Integrate CR (Change Request) from GSMA.
- Bug corrections.

## ■ October 20: Sprint 11

- Bug corrections.
- Store & Forward tests.
- Integrate CR (Change Request) from GSMA.



# Roadmap 2011

## ■ November 21: Sprint 12

- Bug corrections.

## ■ December 20: Sprint 13

- Bug corrections.
- Findbugs report.





# Roadmap 2012

## ■ January 16: Sprint 14

- IOT tests.
- HTTPS provisioning module.
- Findbugs report.

## ■ January 30: Sprint 15

- IOT tests.
- HTTPS provisioning version management.
- DNS management (NAPTR & SRV).
- End user confirmation request API to accept terms of use.

## ■ February 10: Sprint 16

- IOT tests.

## ■ February 22: Sprint 17

- Bug fix after IOT tests.

## ■ March 12: Sprint 18

- Bug fix after IOT tests.
- Rename package of the NIST SIP stack to avoid concurrency problems with native NIST SIP stack.
- Update source code copyrights.



# Roadmap 2012

## ■ March 30: Sprint 19

- IOT tests.
- GSMA RCS Implementation Guidelines.
- Nonce caching.

## ■ April 13: Sprint 20

- IOT tests.
- GSMA RCS Implementation Guidelines.

## ■ April 30: Sprint 21

- Provisioning terms.
- IOT tests.

## ■ May 21: Sprint 22

- Backup & restore RCS settings.
- Bug correction.



# Features not yet planned

- Secured media (MSRP, RTP).
- Specific APN in roaming.
- OMA-DM provisioning.
- LTE support.



# Main features list



# Capability discovery

- Capability discovery via SIP OPTIONS.
- Capability discovery via Anonymous fetch.
- Periodic polling.
- First time discovery.
- Address book monitoring.
- Integration into native address book.
- Supported capabilities: CS video, IM, FT, Image share, Video share, Presence discovery, Social presence, RCS extensions.
- Update capabilities in case of error, end of session.
- Adapt capabilities thanks to network coverage.



# Chat 1/2

- Chat 1-1.
- Chat group.
- File transfer.
- Chat and file transfer history.
- Multi-session management.
- Is-composing.
- List of participants in the session.
- Add 1 participant.
- Add N participants.
- Conference event subscription.
- Support of plain text UTF-8 format.
- Support of message/CPIM format.
- IMDN and message delivery report (“delivered”, “displayed”).



# Chat 2/2

- Block/unblock contacts (black list).
- Spam folder for message received from blocked contacts.
- Store & Forward.
- Use an alias to initiate a chat session.
- Close a chat session after an inactivity timeout.



# Richcall

- Image sharing.
- Live video & pre-recorded video sharing.
- Capability exchange during call.
- Call monitoring.
- H263 file parser.
- H263+ software codec (from Android opencore package).
- H.264 packetizer.
- H264 software codec (from Android opencore package).
- RTCP event management.
- Visio share from 2 individual video share sessions.
- Stop sharing if call hold or if a multiparty call (see new API methods).





# Social presence

- Resource lists management (granted/blocked/revoked).
- Profile sharing management (status, photo-icon, freetext, favorite link, geoloc).
- Notifications management.
- Permanent and partial profile



# Misc.1/2

- Address book integration via ContactContract API.
- Event log history (one content provider which aggregates Call/SMS/MMS/Chat & FT per contact).
- RCS account based on SyncAdapter API.
- Presence is now an optional package which may be activated by configuration.
- RTP stack is now part of the media player to avoid AIDL exchange between UI and stack.
- New SIP stack (NIST SIP stack deployed under Android 2.3).
- Backup and restore last RCS info if user account has changed.
- OTA provisioning via HTTPS.
- Session refresh management.



# Misc. 2/2

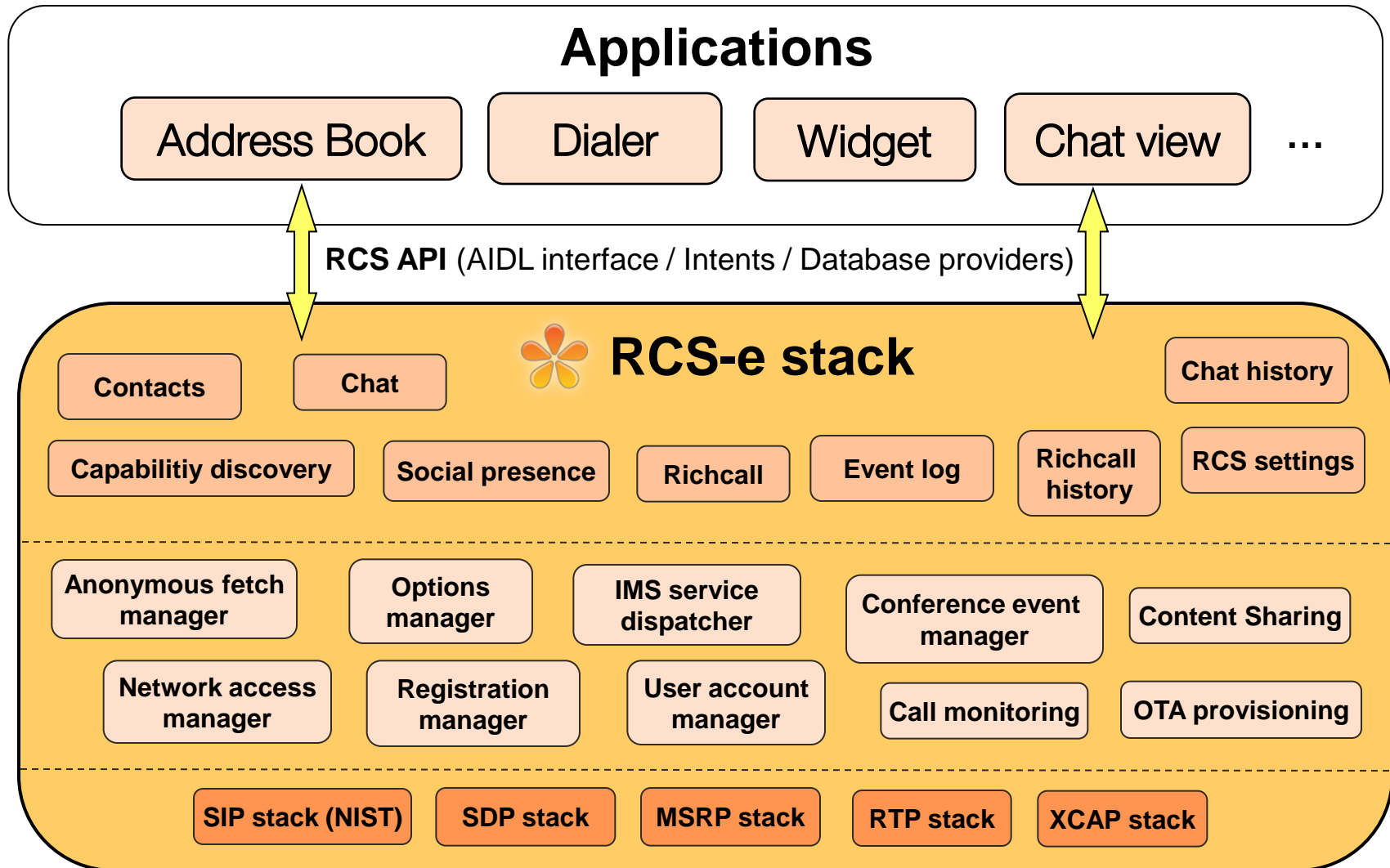
- Support of GIBA & HTTP Digest for IMS attachment.
- Support several network access: mobile or Wi-fi.
- RCS API for multi-applications context.
- SIPS
- GRUU for multidevice management.



# Implementation



# Software architecture



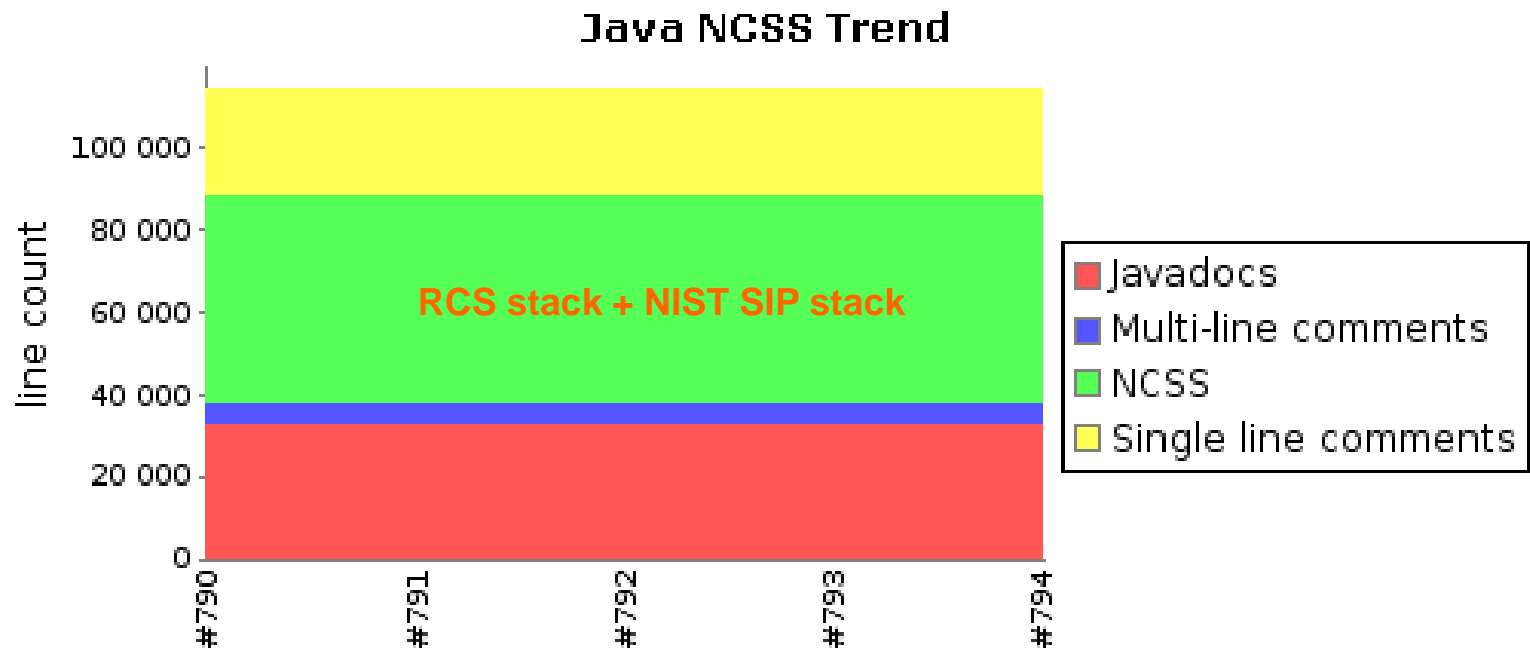
# Supported standards

- See list of supported standards at:

<http://android-rcs-ims-stack.googlecode.com/svn/trunk/core/SUPPORTED-STANDARDS.txt>

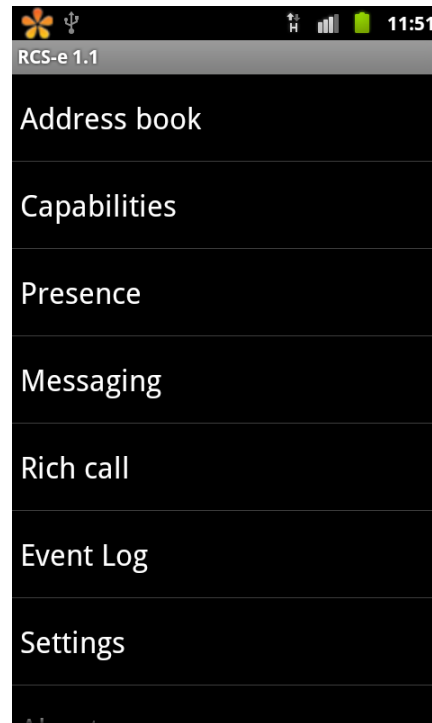


# Metrics



# Tools

- Bug tracking via Mantis.
- Continuous Integration via Hudson.
- RI application which offers a low level UI interface to test RCS stack and its protocols:



RI application



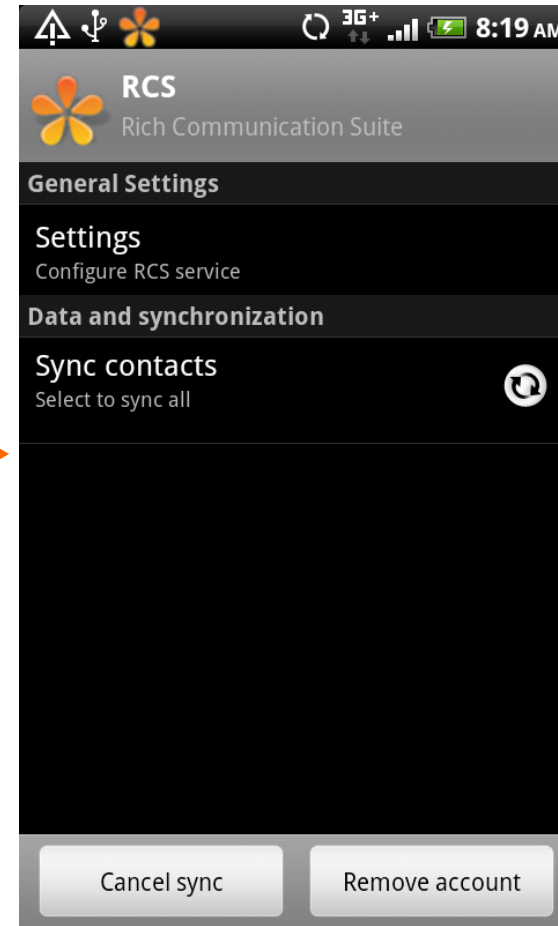
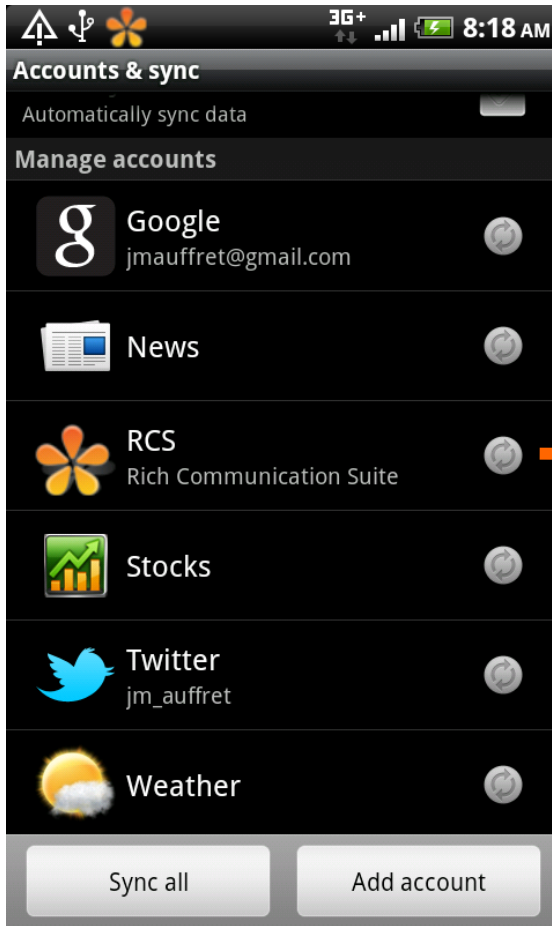


# RCS-e applications

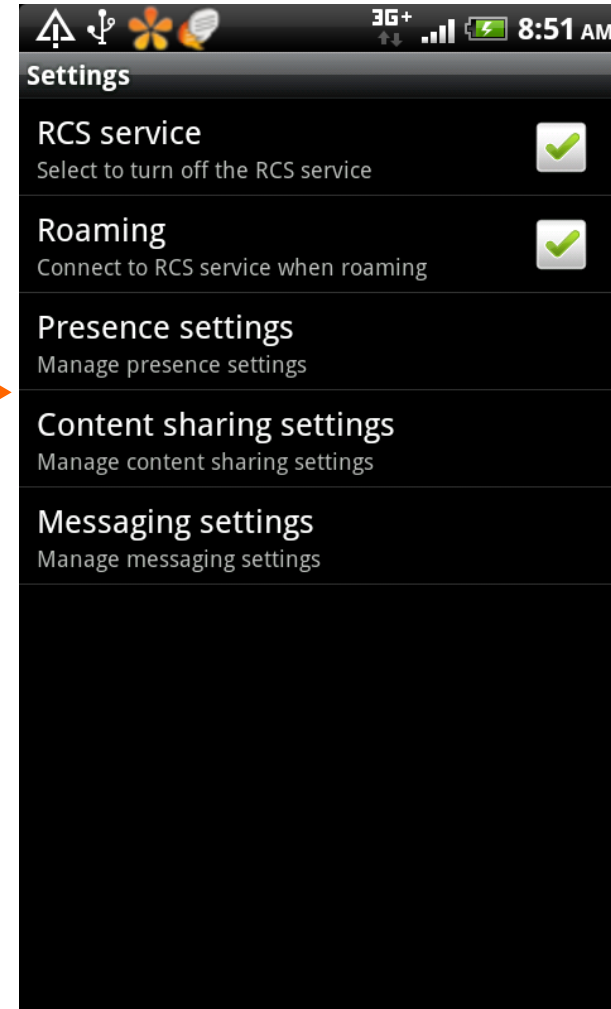
## Screenshots



# RCS account

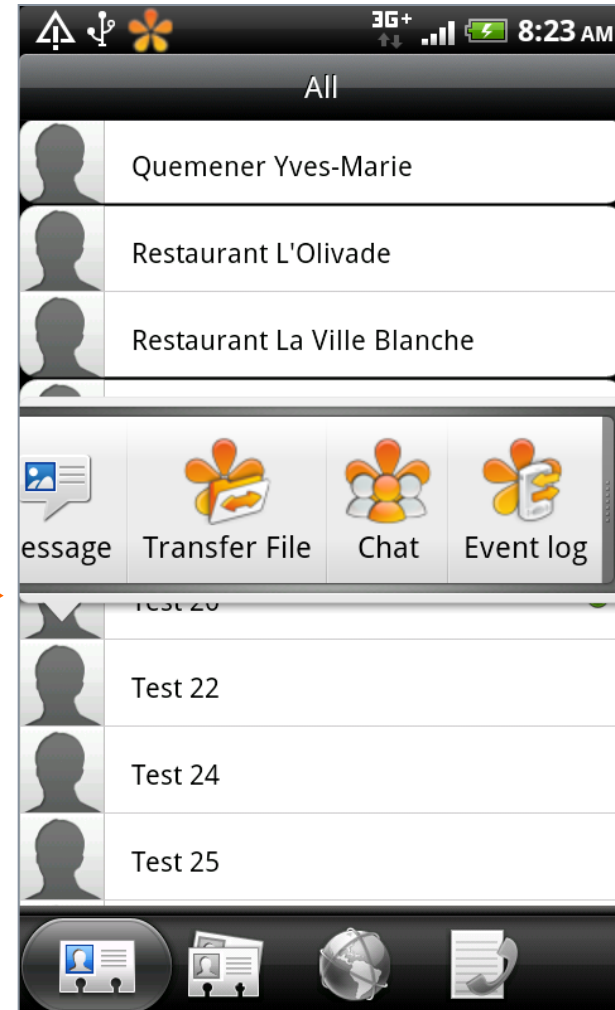
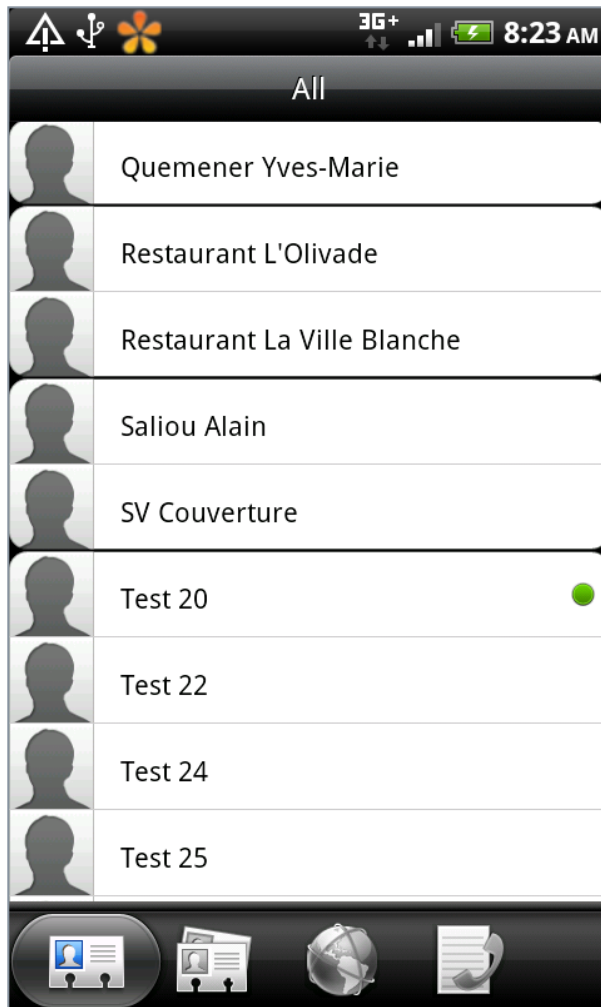


# RCS settings



# Native address book integration 1/2

**Native  
address  
book  
from HTC**



# Local provisioning application

Only for debug

Provisioning

Profile Stack Service Logger

IMS authentication procedure:  
DIGEST

IMS username:  
+34890890022

IMS display name:  
+34890890022

IMS private URI:  
+34890890022@sip.osp.com

IMS password:  
nsnims2008

IMS home domain:

Provisioning

Profile Stack Service Logger

Is-composing timeout for chat (seconds):  
15

Session refresh expire period (seconds):  
3600

Capability refresh timeout (seconds):  
86400

Capability polling period (seconds):  
3600

Capability discovery mode:  
options

☒ Rich call mode

Provisioning


Profile Stack Service Logger

☒ Trace activation

☒ SIP trace activation

☒ Media trace activation

Trace level:  
DEBUG

 Save

