

# OrangeLabs RCS-e stack



## Provisioning management

Edition 1.0

Date: 12/10/2011

Author: Orange Labs, ASC Devices



Orange Labs

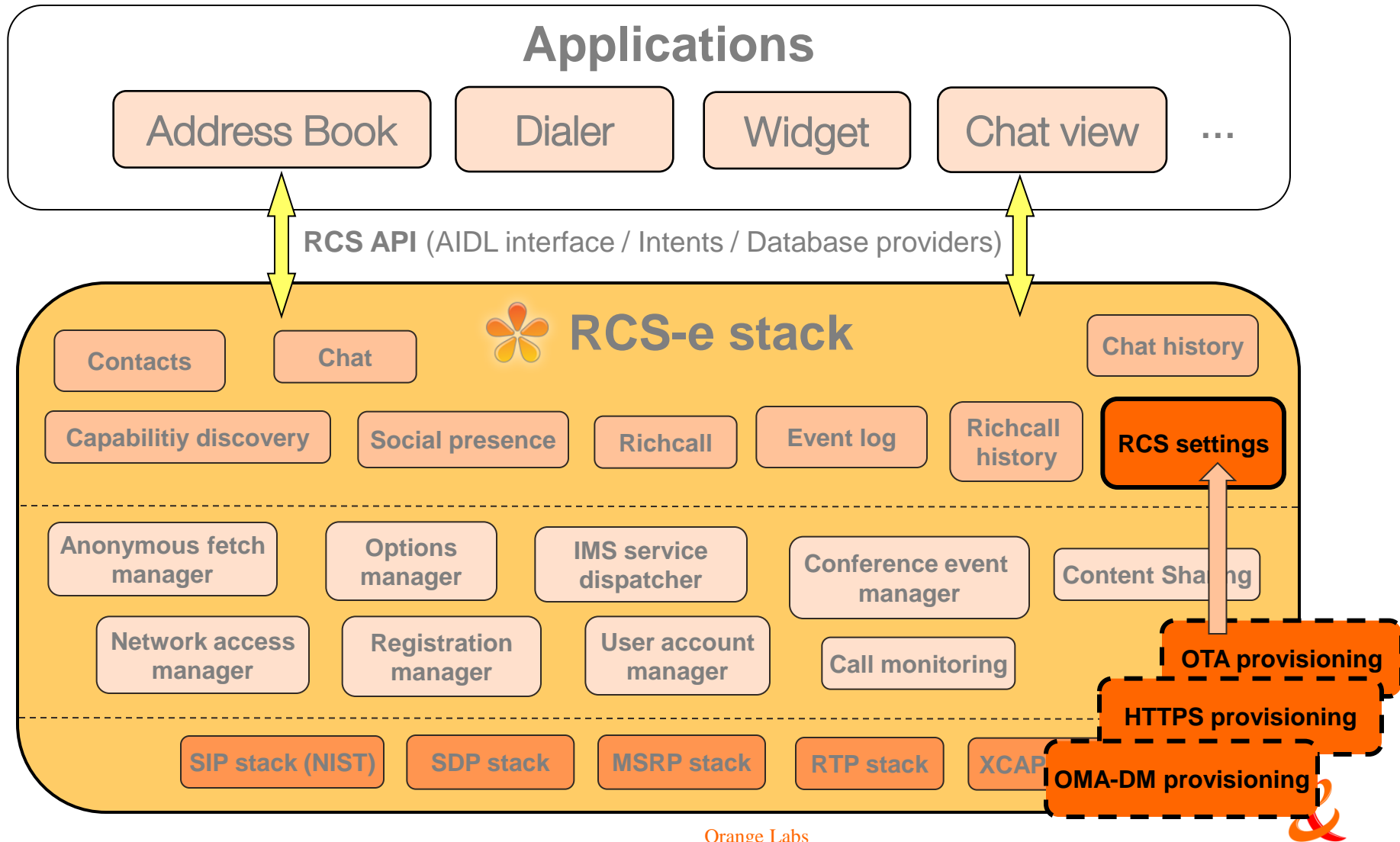


# What is already implemented in the stack today?

- A database for RCS settings which may be shared with other applications installed on the device (stack itself, local provisioning tool, OTA client, .etc).
- An XML configuration file parser which permits to parse the RCS data model specified in Annex A of the RCS-e 1.2 standard.
- A sample of source code showing how to integrate a provisioning mechanism in the stack: here it's a WAP Push event + HTTP download of the config file, then parsing of the config file and update database settings, and finally execute an auto reboot of the stack without any interaction with the end user. See source code at: <http://android-rcs-ims-stack.googlecode.com/svn/trunk/ota> (only available from the v2.3.3 release).



# Stack architecture



# Settings supported by the stack

- Some settings are used by the stack itself, some settings are offered by the stack and used by the UI level (e.g. max IM length).
- See the list of supported settings:



# How to integrate a new provisioning mechanism in the stack?

- Just implement a new module which permits to modify the RCS stack settings and to restart the RCS stack on the device.
- This may be implemented independently from the RCS stack by using its RCS API:
  - Use the Settings API to update the database settings (see class `RcsSettings`).
  - Use the Client API to restart the stack (see methods `stopRcsService` and `startRcsService` of the class `ClientApi`).



# Existing open source code

- Existing open source code may be reused in order to boost the implementation of an OMA-DM client.
- Open source projects:
  - **Android:** <http://source.android.com/source/>. See package « ota » which is deprecated since several releases.
  - **FUNAMBOL:** <https://core.forge.funambol.org/>.

