# Mobillion - architecture

Mobillion currently handles three types of input channels

1. Mobile devices (smartphones)
2. External systems - (ie. issue trackers, monitoring tools, custom applications...)
3. Web application - End users using browsers

Each type of channel is connected to Mobillion Core via corresponding communication module responsible for talking to specific type of external subject (mobile client app, specific type of external system).
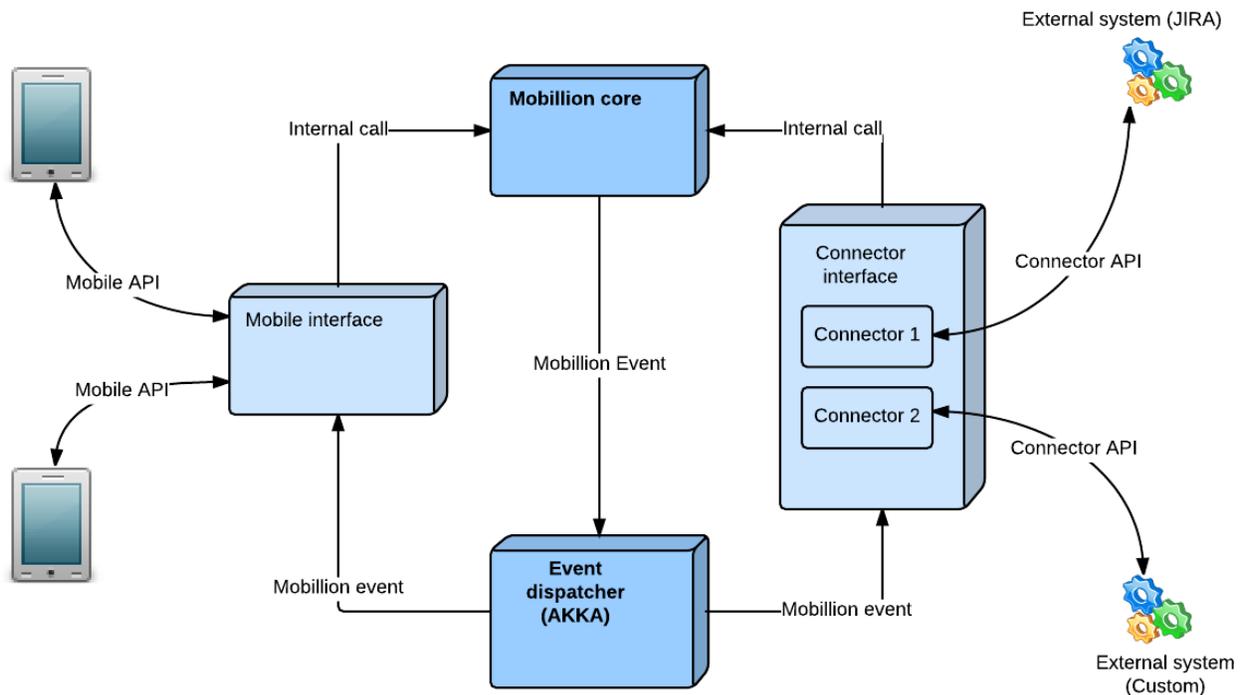
Communication modules receive messages/commands from external subjects and invoke appropriate internal calls on Mobillion Core. Messages from external sources usually notify Mobillion about new alarms , issues or whatever external source business logic deals with. Mobillion Core then decides what needs to be done internally (persist data, update state...). After executing internal logic, it creates Mobillion Events in order to notify all interested subjects about changes that just happened. Core module doesn't care who will receive notifications about these events, it just sends it to event dispatcher.

Dispatcher module is responsible for delivering Mobillion events to all registered event listeners (communication modules) in efficient and safe way. In order to achieve that, it is implemented as Akka Actor. This means that once the core delivers an event to event dispatcher it is free to continue with processing other commands, and delivering events job is handled by dispatcher in separate thread. Some event listeners are registered at application start-up (Mobile interface module, existing (db persisted) connectors) and others can be registered at run-time (connectors created for external sources added via admin web console) .

All events are delivered to all registered event listeners, and listener implementations are responsible for analyzing events to see if they are relevant for them. For example, Mobile interface module examines received event type and data in order to decide which mobile users need to be notified about that event, then sends push message to users mobile device. Connector modules also examine event type and data as well as specific connector configuration settings to decide if external source should be notified about that Mobillion event.

The goal of this architectures is to decouple processes of receiving input commands , handling internal logic and delivering output messages to connected subjects.

Author: Igor Buzatović

Mobillion product description - on Inovatrend corporate web site.