

Smart Home Control
Android Subscriber Push Client – XMPP PushServer
Quelle: <http://sourceforge.net/projects/androidpn/>
Autor: <http://www.linkedin.com/in/sehnoh>
Modifiziert: Birgit Wendholt

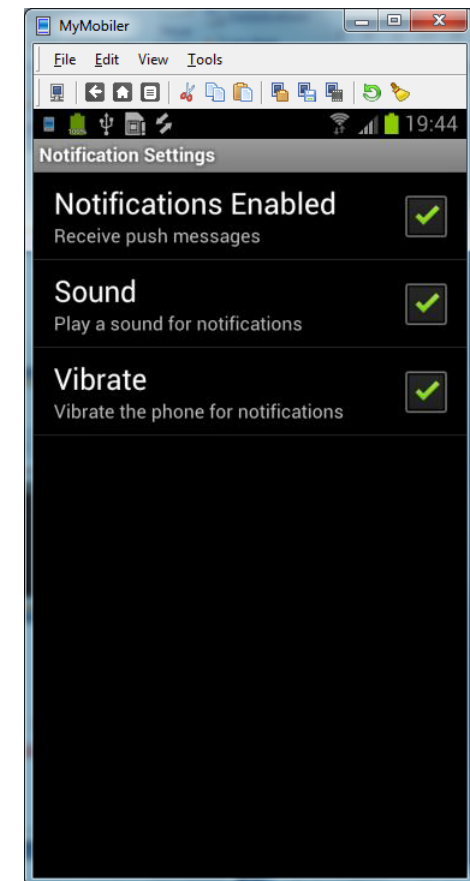
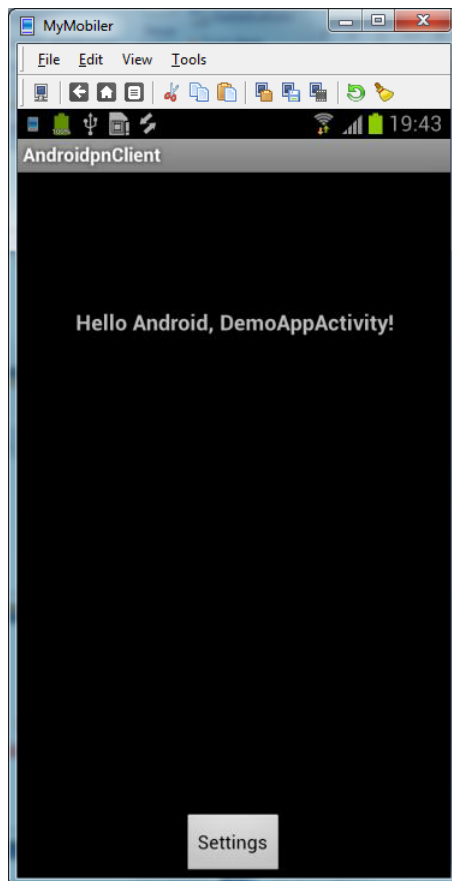
Prof.Dr.Ing.Birgit Wendholt

Einstellungen für den Clients

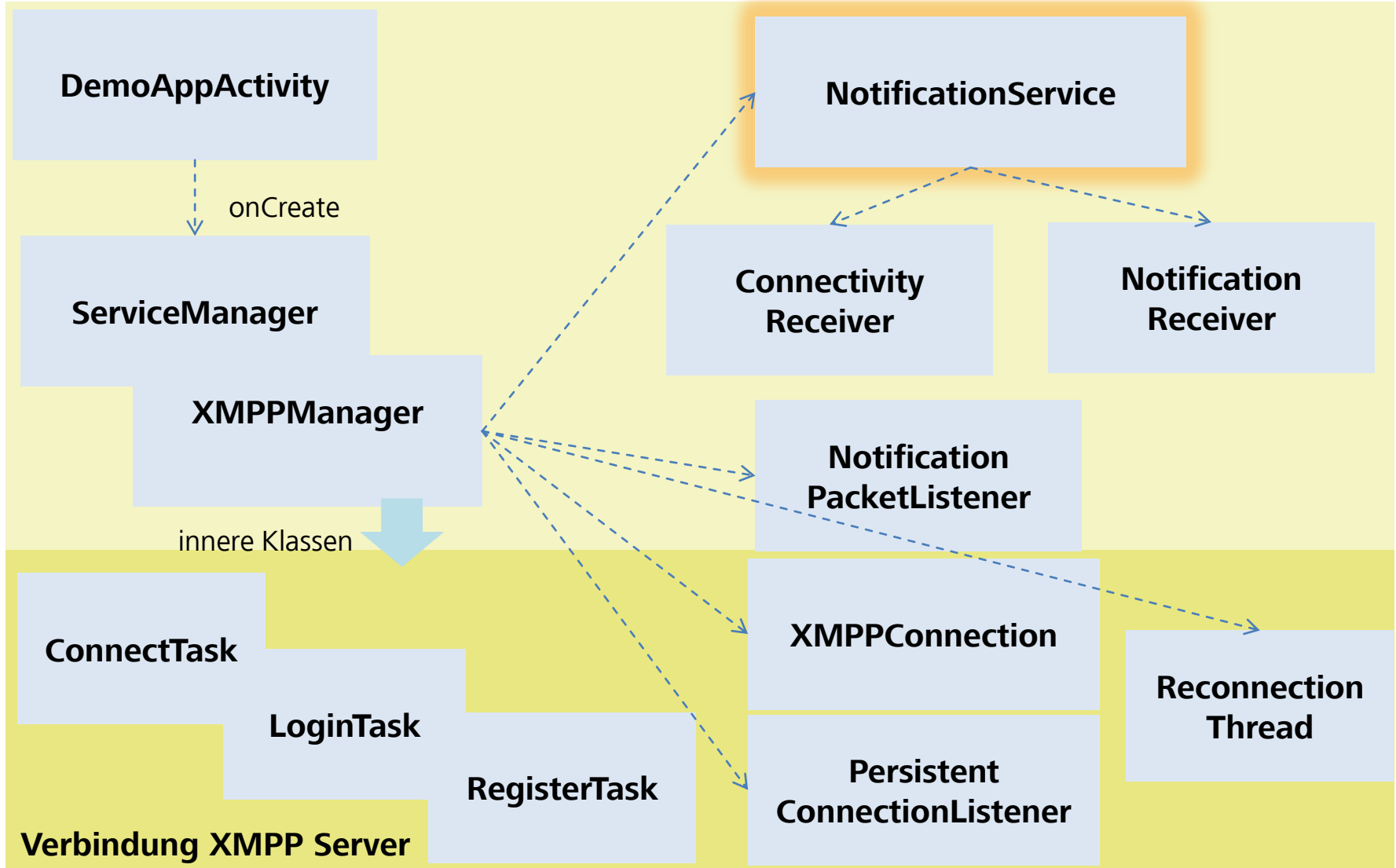
<<activity>>
DemoAppActivity

viewNotificationSettings

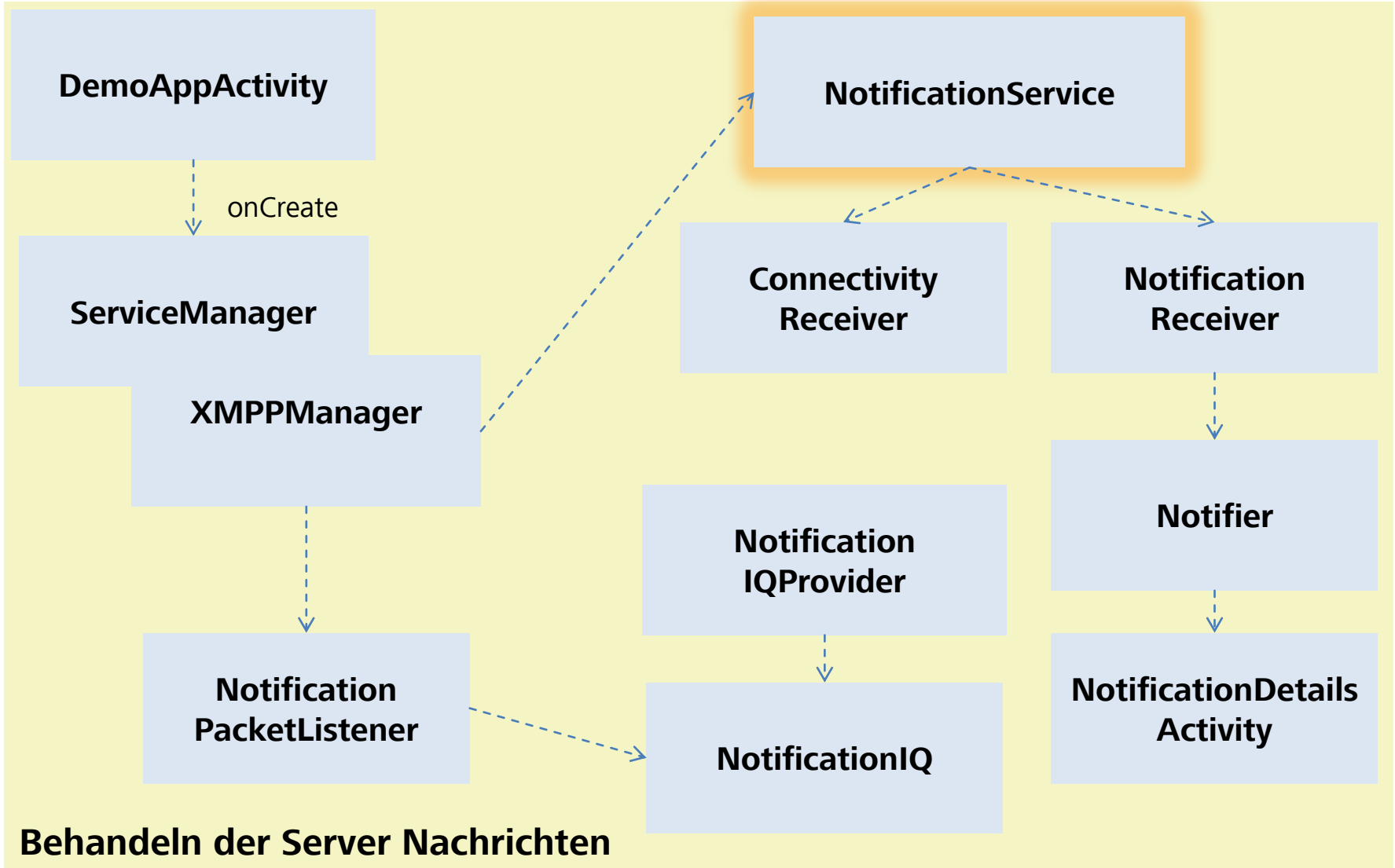
<<class>>
ServiceManager



Übersicht über die Client Klassen (1)



Übersicht über die Client Klassen (2)



Überblick über die Klassen und deren Aufgaben

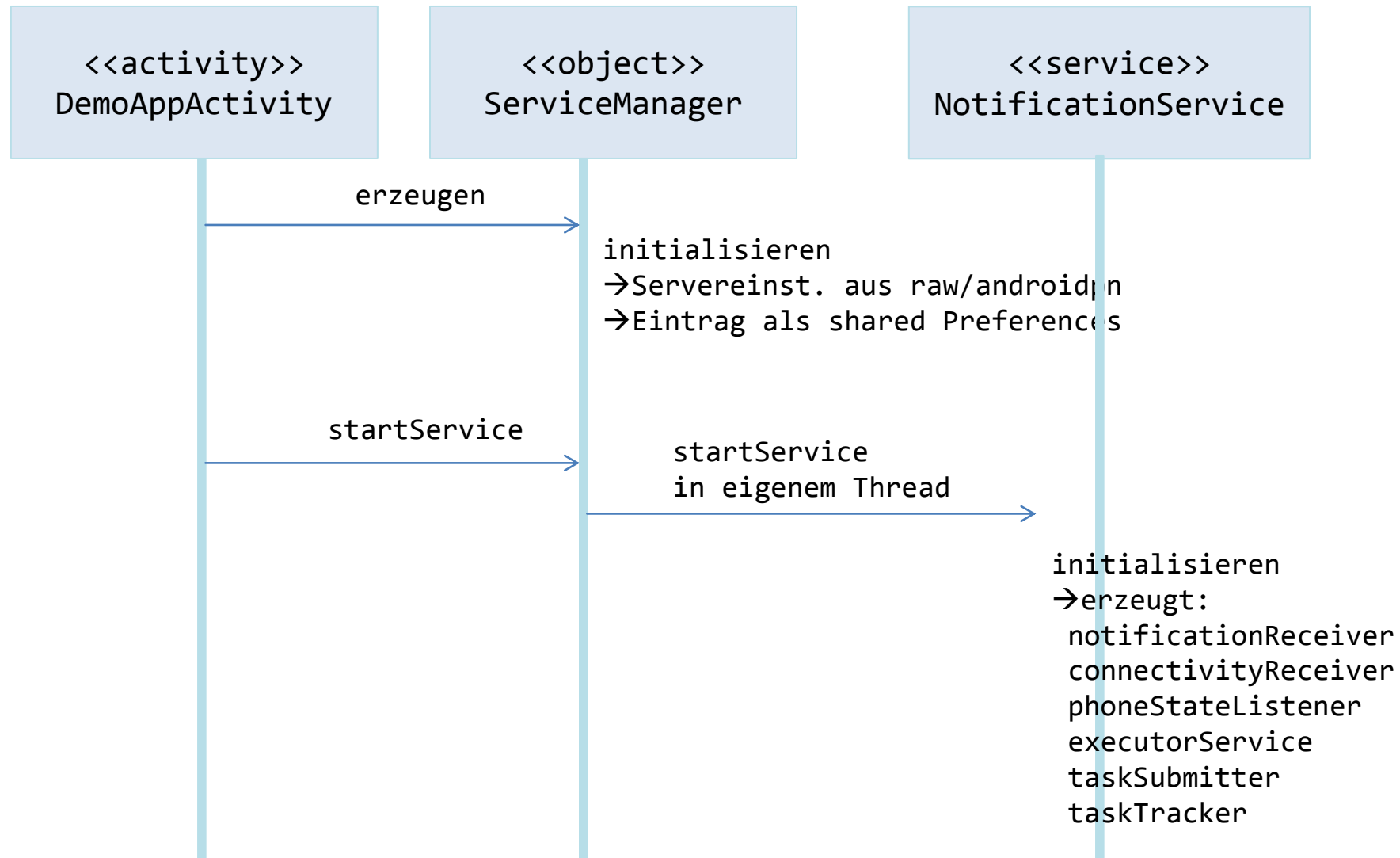
- TaskSubmitter (innere Klasse von NotificationService)
 - führt Tasks des XMPPManagers in separaten Threads aus
- TaskTracker ((innere Klasse von NotificationService))
 - beobachtet und zählt die aktiven Tasks
- NotificationReceiver (BroadcastReceiver): mit speziellem Intent
 - reagiert auf NotificationBroadcasts des Android Kerns
 - erzeugt Notifikationen für Nachrichten des XMPPServers
- NotificationIQProvider / NotificationIQ
 - Parsen von Servernachrichten
- NotificationPacketListener / Notifier
 - Triggern der NotificationReceiver

Überblick über die Klassen und deren Aufgaben

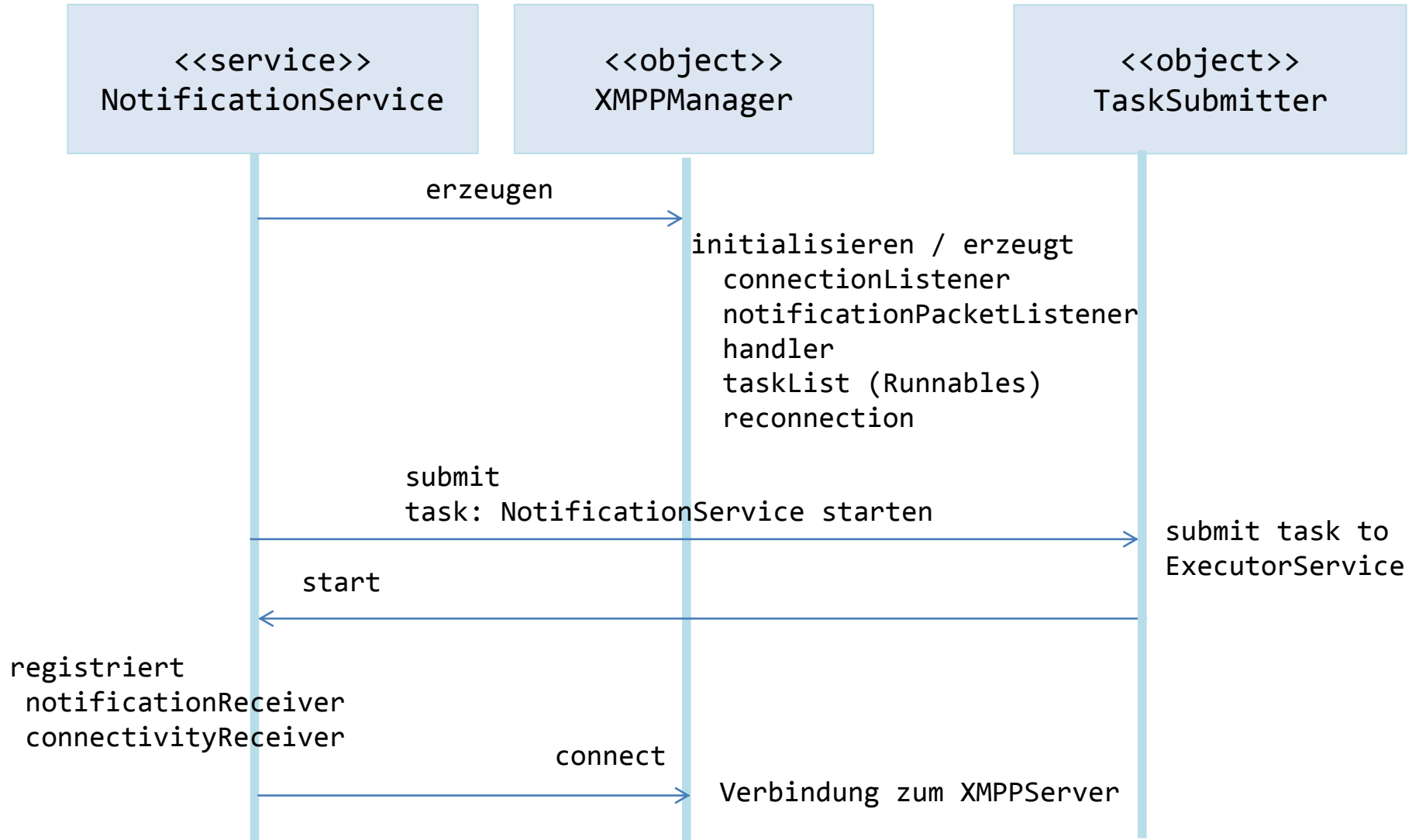
- XMPPManager Schnittstelle zum XMPPServer (PushServer)
 - Verbinden und Registrieren des PushClients als IQProvider (ConnectTask)
 - Registrieren des Accounts (RegisterTask)
 - Authentifizieren (LoginTask)
- PhoneStateChangeListener
 - stellt beim Wechsel der Verbindungsart die Verbindung zwischen NotificationService und XMPPServer wieder her
- ConnectivityReceiver (BroadcastReceiver)
 - reagiert auf Netzwerkaktivität
 - connect / disconnect des NotificationService
- PersistentConnectionListener / ReconnectionThread
 - bauen Verbindungen auf / ab
 - bauen Verbindungen nach Verbindungsverlust wieder auf

Initialisieren des PushClients

PushClient Starten des NotificationService

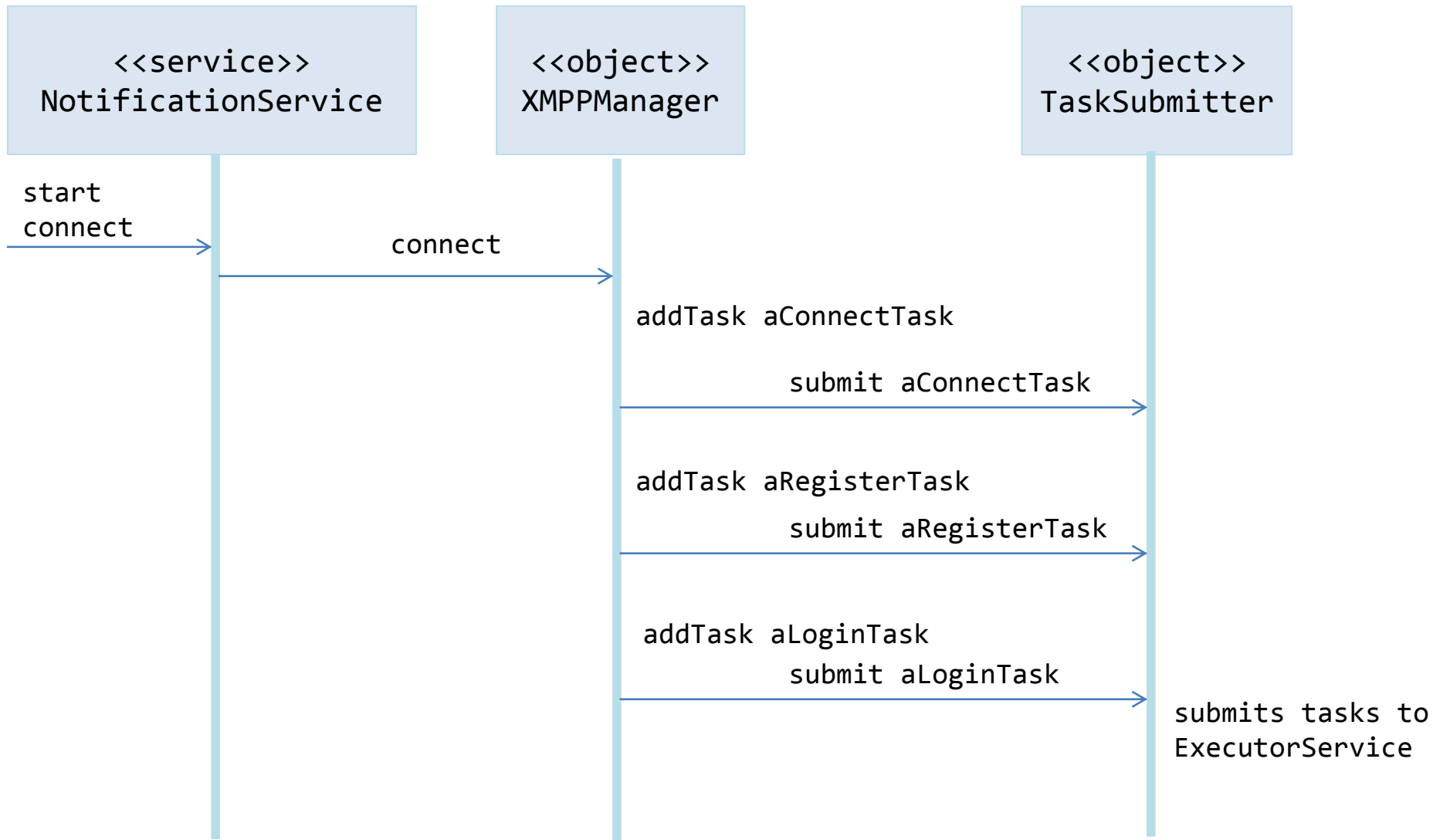


NotificationService onCreate



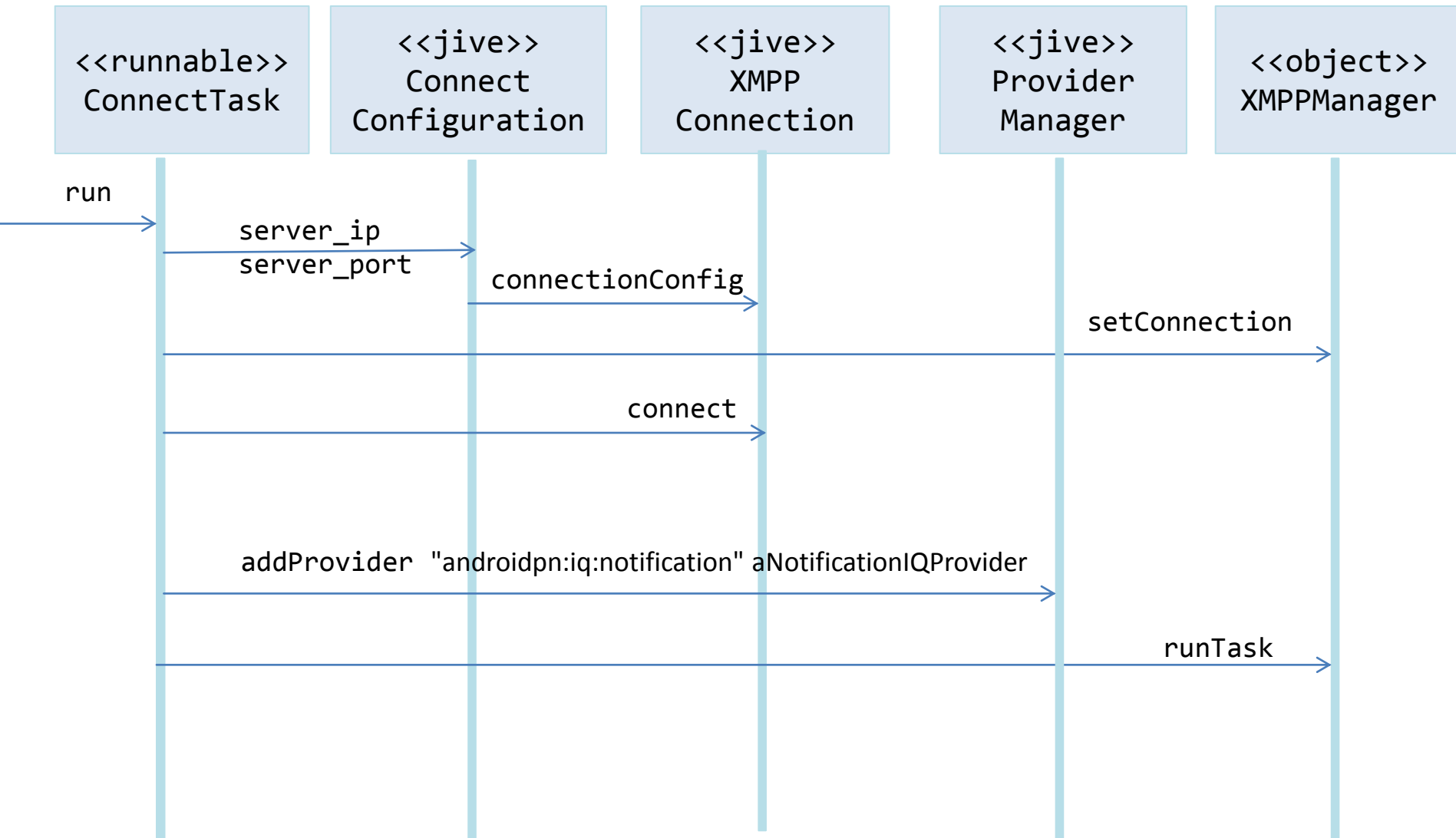
Connect

NotificationService / XMPPManager zum XMPPServer

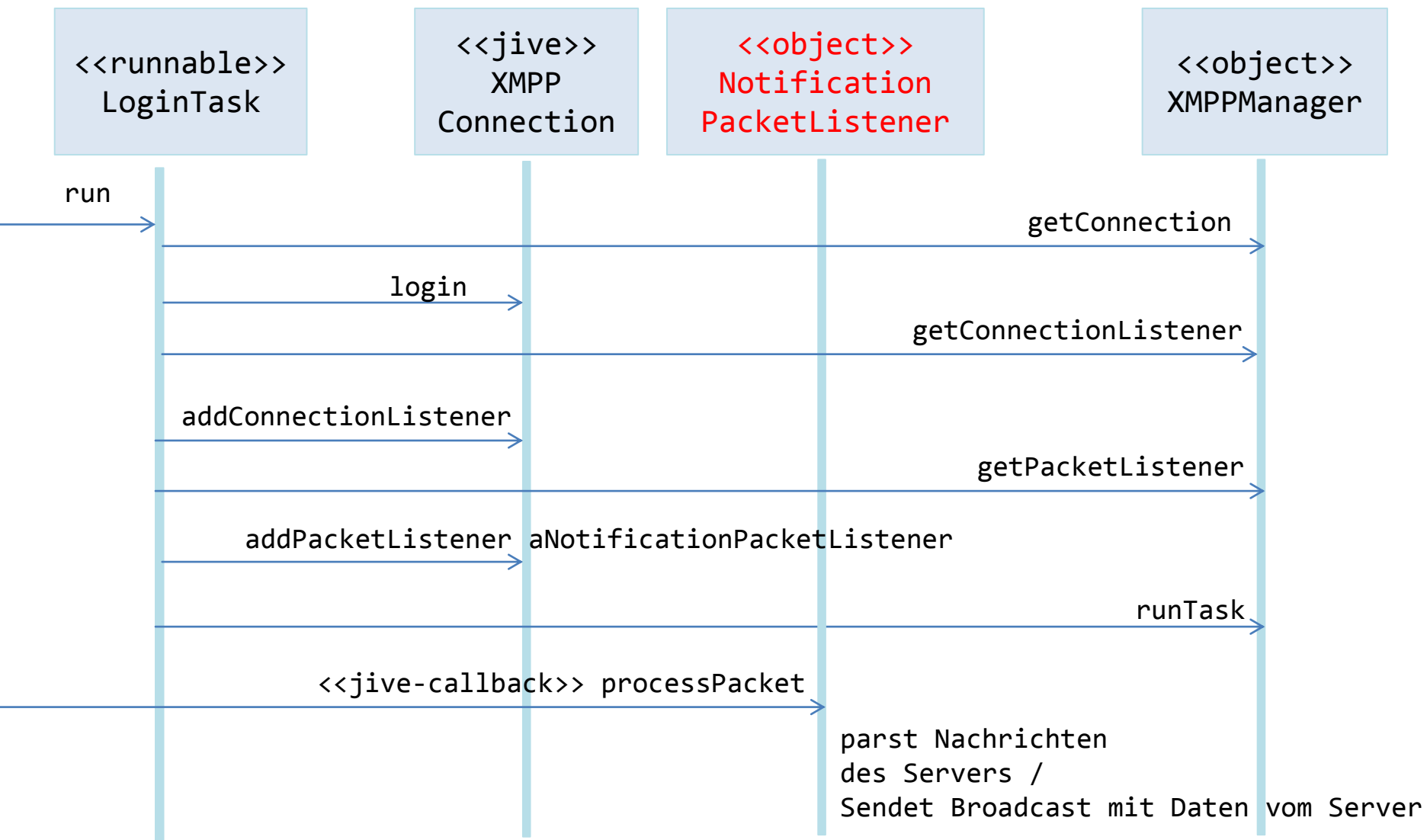


ConnectTask

Verbindung zum XMPPServer / Registrieren des IQProviders



Registrieren des PushClients beim XMPPServer

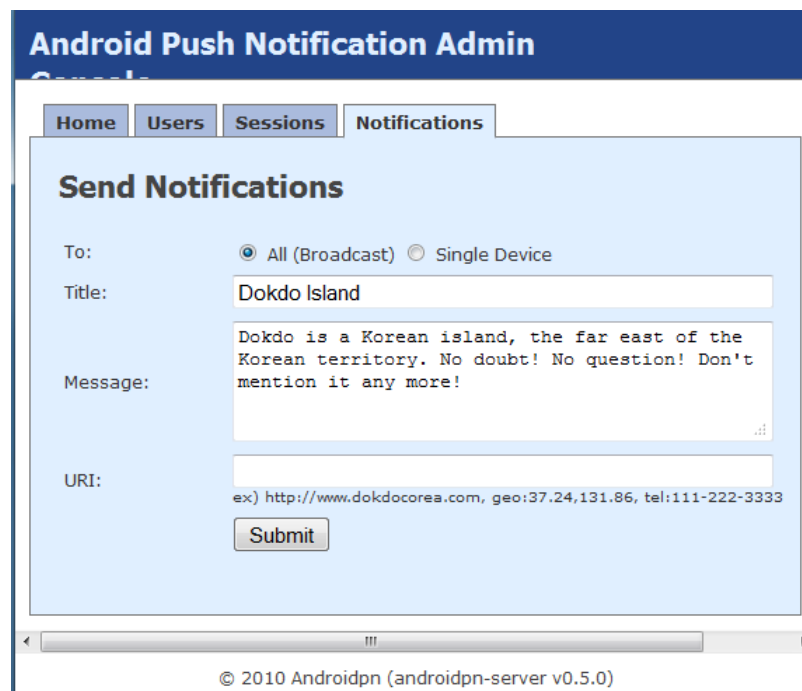
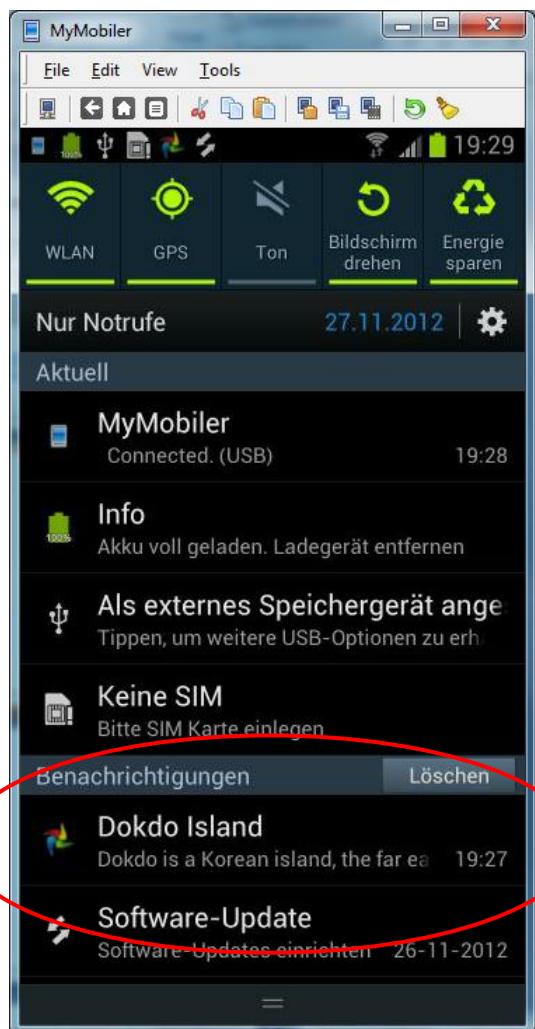


Empfangen und Verarbeiten der Server Nachrichten

Empfangen und Verarbeiten der Nachrichten des XMPPServers

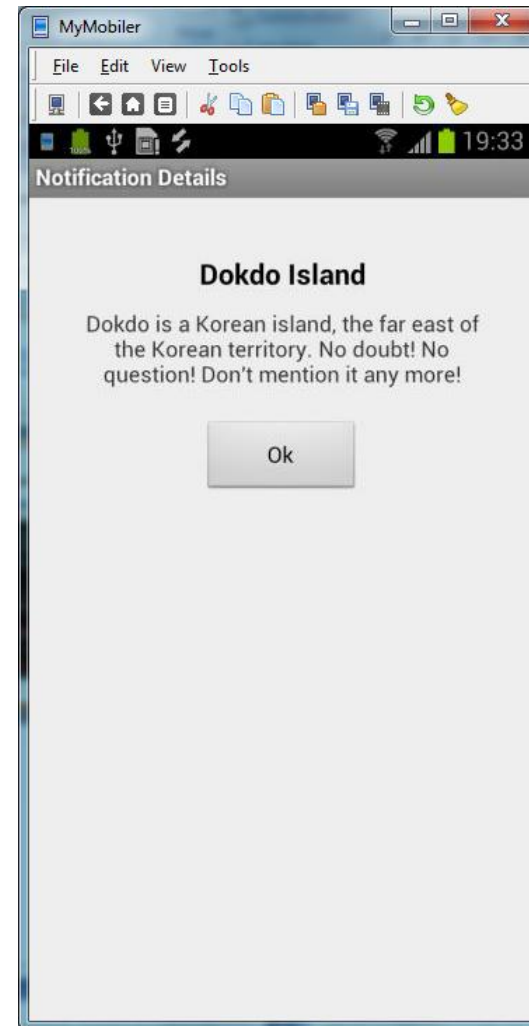
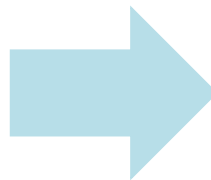
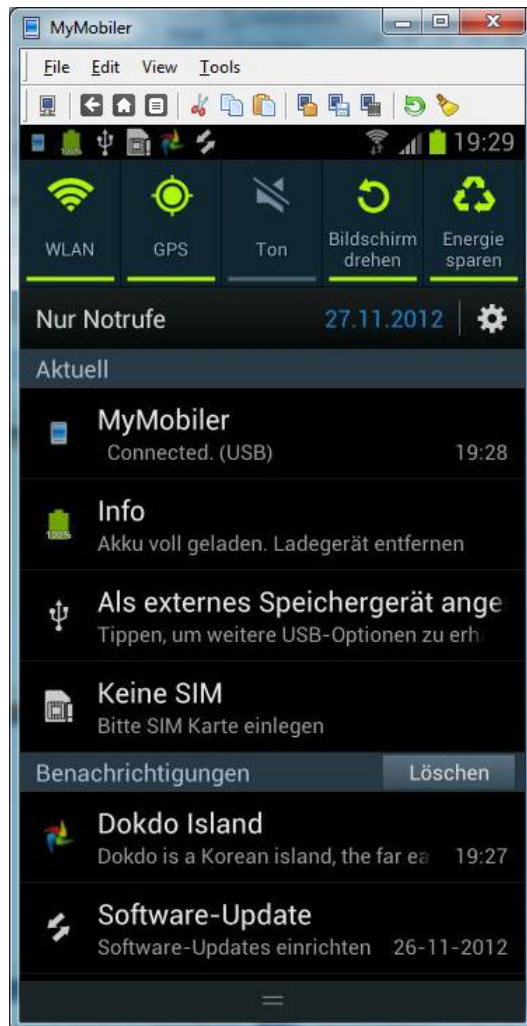
- Empfangen / Parsen / Weiterleiten:
 - **NotificationIQProvider** wird vom **ConnectTask** als Provider für Server Nachrichten registriert
 - parsed XML Nachrichten im XMPP Format in IQ Objekte → **NotificationIQ** (Subtyp von Packet) in der Callback Methode **parseIQ**
 - **NotificationPacketListener**
 - im **LoginTask** als **PacketListener** bei der **ServerConnection** registriert
 - interpretiert **NotificationIQ** Pakete im **processPacket** Callback
 - sendet XMPPServer Daten mittels Broadcast an das Android System
- Verarbeiten:
 - **NotificationService** registriert (in der Methode **start**) einen **BroadcastReceiver** für die Aktion **ACTION_SHOW_NOTIFICATION** (→ **NotificationReceiver**)
 - **NotificationReceiver** löst im **onReceive** Callback eine Statusbar Notifikation aus
 - Android Broadcasts → Folien zu Broadcasts

Ergebnis des erfolgreichen Empfangens



Nachricht versendet über die
Admin Konsole des Webapplikation

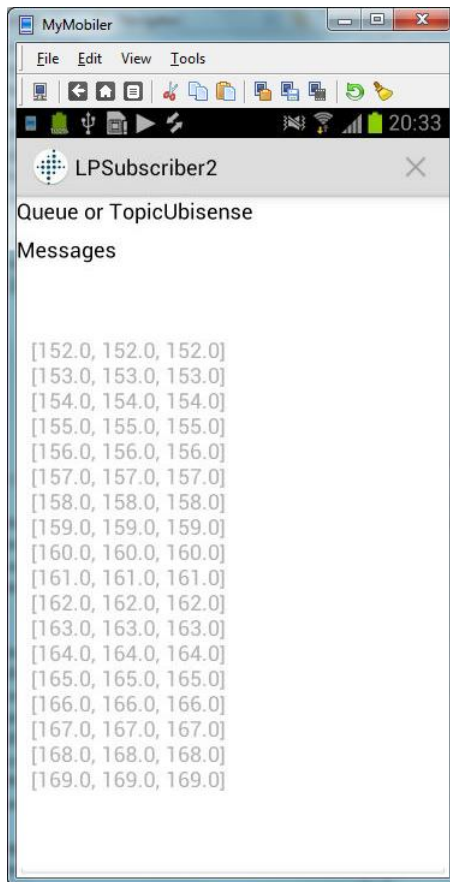
Vollständige Nachricht



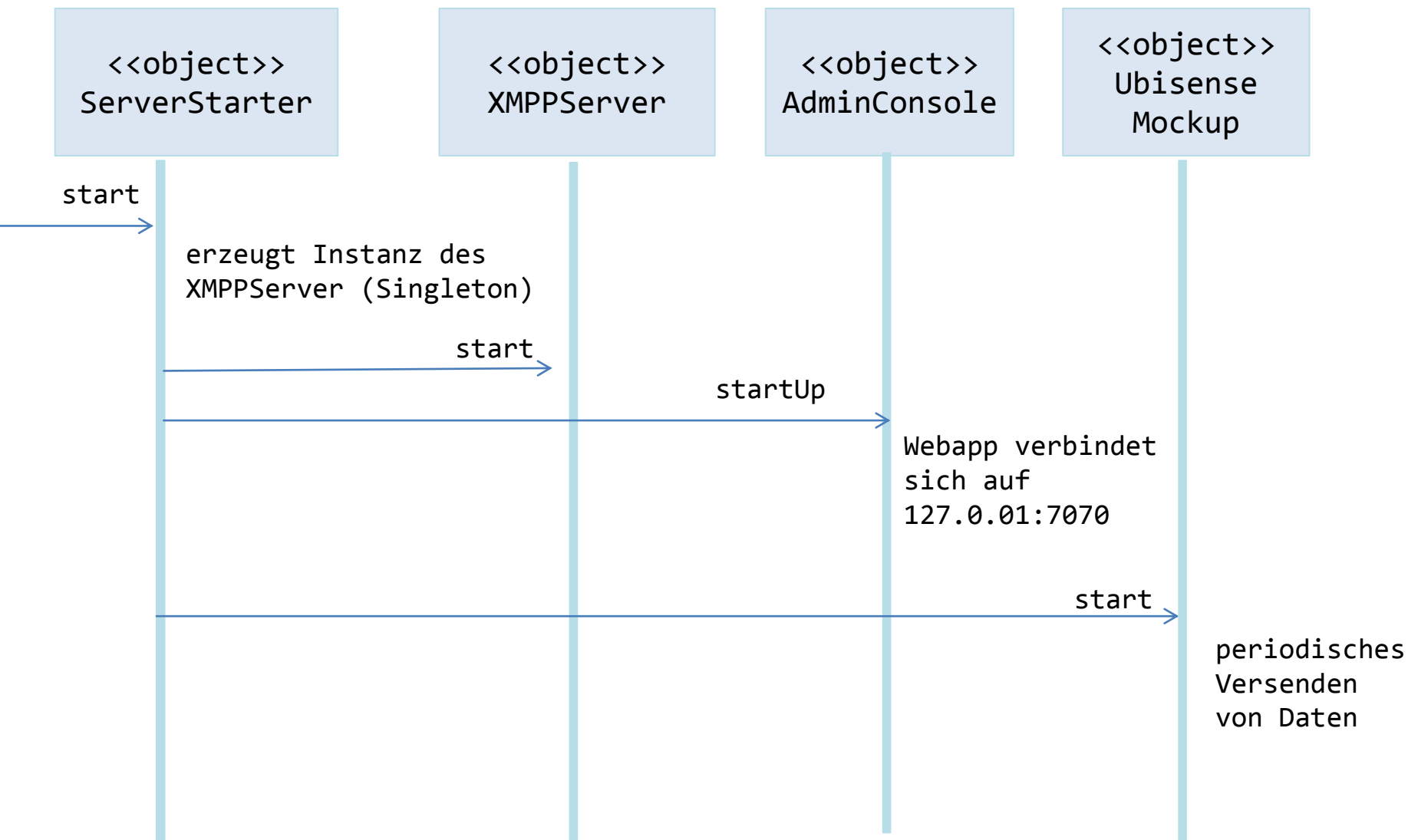
Modifikationen für das Empfangen und Verarbeiten von Nachrichten

- **Activity**
 - Anzeige periodische versendeter Servernachrichten
 - Registrieren des **NotificationListener**
- **NotificationListener**
 - triggert keine TaskBar Notifikation (alle damit zusammenhängenden Klassen wurden gelöscht)
 - liefert die Inhalte an die Aktivität aus
- **NotificationPacketListener**
 - anpassen, wenn sich die Inhalte und Quellen der Nachrichten ändern (z.B. für UbiSense)
- **NotificationIQProvider**
 - anpassen, wenn sich die Inhalte und Quellen der Nachrichten ändern (z.B. für UbiSense)
- **NotificationService**
 - Registrieren des **NotificationListener** entfernen

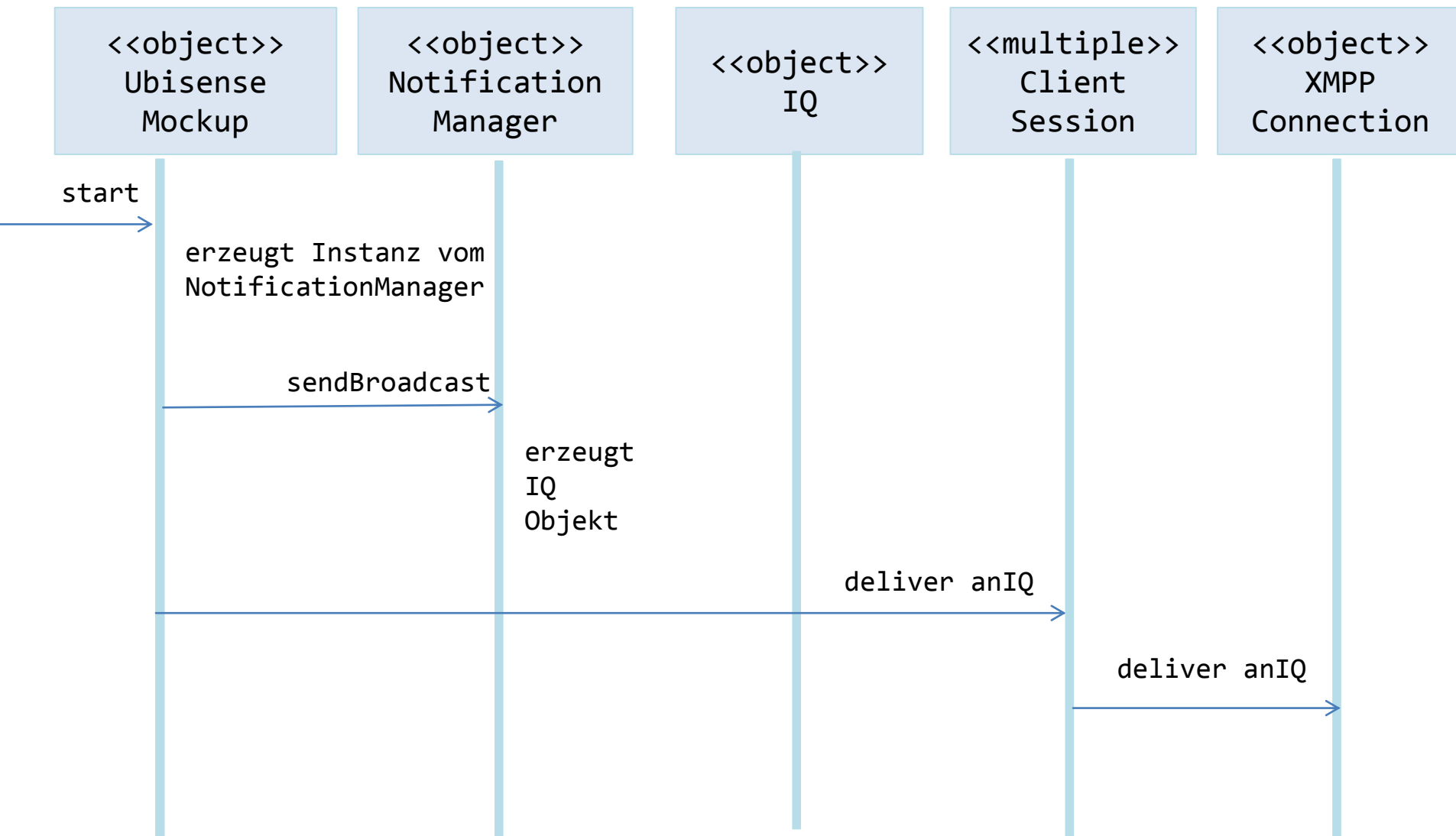
Mockup für das Empfangen periodischer Daten vom XMPP Server



PushServer modifiziert



PushServer modifiziert



XMPP PushServer

- Anpassungen beim PushServer der Schnittstellen zum Client:
 - Package **container** → neue Klassen als Publisher einrichten (z.B. UbisensePublisher)
 - Klassen im **XMPPServer** erzeugen und starten
 - evtl. die Methode **sendBroadcast** NotificationManager
 - evtl die **createNotificationIQ** in NotificationManager
 - Umstellen der Publisher auf Unicast Kommunikation