IoT Solution Guidelines, Version 2.0 - No Harm to Network

Introduction

This document summarizes No Harm to Network requirements from Deutsche Telekom originating in GSMA TS.34, *IoT Device Connection Efficiency Guidelines* [1], use-cases or features not covered yet within GSMA TS.34, as well as lessons learned gained from IoT commercial deployments. Requirements including the words "SHALL" or "SHALL NOT" in their descriptions are mandatory; all guidelines with "SHOULD" or "SHOULD NOT" are recommended.

This document is divided into three sections, reflecting how IoT Service Providers must implement No Harm to Network considerations and best-practice design in the different IoT Solution Layers (refer to Figure 1).

- Definitions
- IoT Service Provider Guidelines (IoT Server Application and Network Service Enablement Layer)
- IoT Device Guidelines (IoT Device Application and Device Service Enablement Layer)

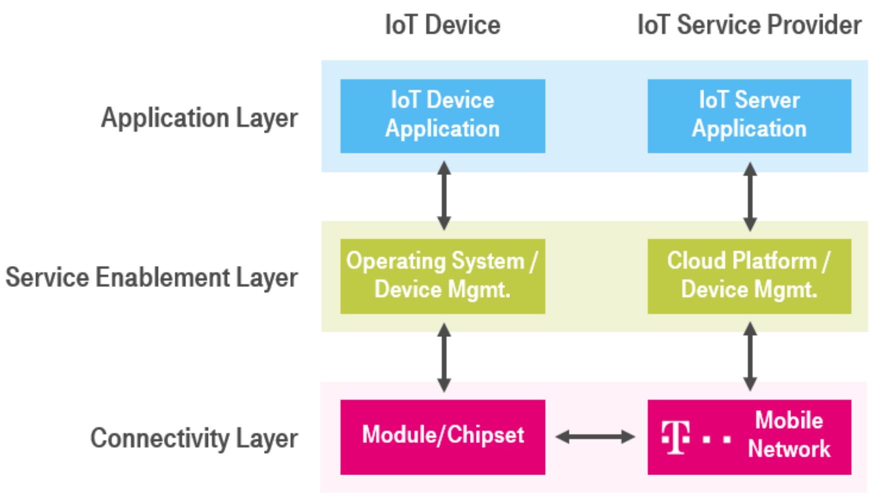

Figure 1: IoT Solution Layers

I. Definitions

IoT Service Provider

Companies offering IoT services to end consumers or enterprises via Deutsche Telekom's Connectivity Layer (3GPP mobile networks). The IoT Service Provider hosts their IoT Service on an own Application and Service Layer (application server or cloud platform), or may choose to integrate Deutsche Telekom's IoT services: Device Management, Storage, Analytics, Web Apps, etc.). The provider may be a Mobile (Virtual) Network Operator, the provider of Service Enablement, or the developer of IoT Device and Server Applications.

IoT Service Platform

Application server or cloud platform used by the IoT Service Provider to host IoT services, manage IoT Devices and exchange data with their IoT Device Applications over Deutsche Telekom's Connectivity Layer.

IoT Device

Sensors, actuators, or other deployed Machine to Machine (M2M) hardware exchanging data bidirectionally and managed by the IoT Service Provider over the Application, Service Enablement and Connectivity Layers. The communication between IoT Device and IoT Service Provider is referred to as the IoT Service.

[1] https://www.gsma.com/iot/gsma-iot-device-connection-efficiency-guidelines/

IoT Device Application

The application logic running on the IoT Device's microcontroller (MCU) and exchanging data with the IoT Service Platform. It sends AT commands to the IoT Device's integrated communication module/chipset in order to access Deutsche Telekom's Connectivity Layer. It may include the IoT Service Provider's Service Layer client.

II. IoT Service Provider Guidelines

Avoidance of Synchronized Behavior

Any IoT Service Platform which communicates to multiple IoT Devices SHALL avoid synchronized behavior and employ a randomized pattern for accessing IoT Devices registered to the platform's domain. The triggering of data transmissions, the rebooting of the IoT Device hardware or sub-components (such as the communication module/chipset), or the issuing device management commands (including, but not limited to (re-) registrations and firmware updates) SHALL NOT be synchronized.

Behavior when IoT Service Platform Temporarily Offline

If the IoT Service Platform is temporarily offline, it SHALL NOT request all IoT Devices to synchronize all at once when it comes back online.

Triggering Devices only when Attached

The IoT Service Platform SHALL be aware of the IoT Device's state and only send "wake up" triggers whenever the IoT Device is known to be attached to the mobile network.

Behavior when IoT Device does not Respond to SMS Triggers

If the IoT Service Platform uses SMS triggers to "wake up" IoT Devices, the IoT Service Platform SHALL avoid sending multiple SMS triggers when no response is received within a certain time period.

Communication over a 3GPP NB-IoT access bearer SHALL NOT use SMS on Deutsche Telekom's mobile network.

Behavior when SIM Subscription is Inactive

If the SIM subscription associated with an IoT Device is to be placed in a temporarily inactive state (i.e. for a fixed period of time), the IoT Service Provider SHALL first ensure that the IoT Device's communication module/chipset is temporarily disabled to restrict it from trying to register to the mobile network once the SIM is disabled.

Behavior when SIM Subscription is Permanently Disabled

Before the SIM subscription associated with an IoT Device is to be placed in a permanently terminated state, the IoT Service Provider SHALL first ensure that the IoT Device's communication module/chipset is permanently disabled to restrict it from trying to register to the mobile network once the SIM is disabled.

The IoT Service Provider SHOULD consider avoiding mechanisms for the permanent termination of IoT Devices that are not easily serviceable, as it may require manual intervention (i.e. a service call) to re-enable the IoT Devices.

Frequency and Prioritization of Data Transmissions

Whenever there is a need to transmit data over the mobile network, the IoT Service Platform SHOULD classify the priority of each communication. The IoT Service Platform distinguishes between high-priority data requiring instantaneous transmission, versus delay-tolerant or lower-priority data which can be aggregated and sent during non-peak hours.

IoT Server Applications communicating with IoT Devices over 3GPP Low Power Wide Area (LPWA) access bearers, such as NB-IoT and LTE-M, SHALL optimize their application reporting period to never exceed Deutsche Telekom affiliate tariff's daily maximum number of messages.

Data Aggregation, Compression and Transcoding

The IoT Server Application SHALL minimize the number of parallel mobile network connections and overall frequency of connections to IoT Devices over the mobile network. Data is aggregated by the IoT Server Application into an application report before being compressed and sent over the mobile network. Data transcoding and compression techniques are used, as per the IoT Service's intended Quality of Service, to reduce connection attempts and data volumes.

IoT Server Application using 3GPP Low Power Wide Area (LPWA) access bearers, such as NB-IoT and LTE-M, SHALL optimize their payload sizes to comply with Deutsche Telekom affiliate monthly volume limits.

III. IoT Device Guidelines

Avoidance of Synchronized Behavior

The IoT Device Application SHALL avoid synchronized behavior with other IoT Devices or events, employing a randomized pattern (e.g. over a period of time, from a few seconds to several hours, or days) to request a mobile network connection over the Connectivity Layer. The triggering of data transmissions, the rebooting of the IoT Device hardware or sub-components (such as the communication module/chipset), or execution of device management commands (including, but not limited to (re-) registrations and firmware updates) SHALL NOT be synchronized.

Use of "Always-on" Connectivity

If the IoT Device Application sends data very frequently (i.e. inactivity periods shorter than two hours), it SHALL use a persistent PDP/PDN connection with the mobile network instead of activating and deactivating said connectivity.

Handling of "Keep Alive" Messages on Home Network

If the communication between the IoT Devices and mobile network is IP-based, it may require the use of TCP / UDP "keep alive" messages. In such cases, the IoT Device Application SHALL automatically detect the server-specific timers and/or mobile network firewall timers, such TCP_IDLE value or UDP_IDLE value (NAT timers as defined by Deutsche Telekom for consumer APN, or by business enterprise for own-administered NAT, in the case of private APN), when using push services. This is achieved by increasing the polling interval dynamically until a mobile network timeout occurs, and then operating just below the timeout value. Fixed polling intervals SHALL NOT be used by the IoT Device Application, as polling interval values change between Deutsche Telekom affiliates, and may dynamically adapt with mobile network loading.

IoT Device Applications communicating with the IoT Server Application over 3GPP Low Power Wide Area (LPWA) access bearers, such as NB-IoT and LTE-M, SHOULD NOT implement TCP / UDP "keep alive" messages on the home network.

Handling of "Keep Alive" Messages on Roaming Network

To minimize the possibility of IP connectivity being lost when camping for extended periods (two hours, or more) on a roaming network – i.e. due to expiration of firewall timers, the IoT Device Application SHALL periodically (period less than two hours) transmit small amounts of data to the IoT Service Platform via the visited network. A randomized timer triggers this mechanism, ensuring that the simultaneous transmission of data from a large number of IoT Devices via the visited network is avoided.

IoT Device Applications communicating with the IoT Server Application over 3GPP Low Power Wide Area (LPWA) access bearers, such as NB-IoT and LTE-M, SHOULD NOT implement TCP / UDP "keep alive" messages on the roaming network.

IoT Service Coordination

If the IoT Device Application communicates with several IoT Server Applications using the same communication module/chipset, the IoT Device Application SHOULD coordinate the payload transmission of each IoT Service in a way which makes efficient use of the mobile network.

Data Aggregation, Compression and Transcoding

The IoT Device Application SHALL minimize the number of parallel mobile network connections and overall frequency of connections between the IoT Device and the mobile network. Data is aggregated by the IoT Device Application into an application report before being compressed and sent over the mobile network. Data transcoding and compression techniques are used, as per the IoT Service's intended Quality of Service, to reduce connection attempts and data volumes.

The IoT Device Application SHOULD monitor the volume of data it sends and receives over a set period of time. If the volume of data will soon exceed a maximum value defined by the IoT Service Provider, the IoT Device Application sends a report to the IoT Service Platform and stops the regular sending of data until the necessary time period has expired.

IoT Devices Applications communicating with IoT Server Applications over 3GPP Low Power Wide Area (LPWA) access bearers, such as NB-IoT and LTE-M, SHALL optimize their payload sizes to comply with Deutsche Telekom affiliate monthly volume limits.

Frequency and Prioritization of Data Transmissions

Whenever there is a need to transmit data over the mobile network, the IoT Device Application SHOULD classify the priority of each communication. The IoT Device Application distinguishes between high-priority data requiring instantaneous transmission, versus delay-tolerant or lower-priority data which can be aggregated and sent during non-peak hours.

The IoT Device Application SHOULD monitor the number of network connections it attempts over a set period of time. If the number of connection attempts exceeds a maximum value set by the IoT Service Provider, the IoT Device Application sends a report to the IoT Service Platform and stops requesting mobile network connectivity until the necessary time period has expired.

IoT Devices Applications communicating with IoT Server Applications over 3GPP Low Power Wide Area (LPWA) access bearers, such as NB-IoT and LTE-M, SHALL optimize their application reporting period to never exceed the Deutsche Telekom affiliate tariff's daily maximum number of messages.

Off-Peak Communication

If allowed by the IoT Service, the IoT Device Application SHOULD avoid concentrating communication over the mobile network during periods of high utilization (i.e. transmissions during early morning hours are preferred).

Adaption to Mobile Network Capabilities, Data Speed and Latency

The IoT Device Application SHALL be capable of adapting to changes in mobile network feature capability and service exposure. Furthermore, it is designed to cope with variations in mobile network data speed and latency, considering the differences in available throughput, data speed and latency when switching between different 3GPP access bearers (i.e. 2G, 3G, LTE and LPWA).

Low Power Mode

If the IoT Device Application does not need to exchange any data with the IoT Service Platform for a period greater than 24 hours, and the IoT Service can tolerate some latency, the IoT Device SHOULD implement a power-saving mode where the device's communication module/chipset is effectively powered down between data transmissions. This will reduce the IoT Device's power consumption and reduce mobile network signaling.

IoT Device Applications communicating over 3GPP Low Power Wide Area (LPWA) access bearers, such as NB-IoT and LTE-M, SHOULD NOT power down their communication module/chipset. The 3GPP power saving features SHOULD be used instead, thus avoiding power-draining, system selection scanning procedures.

Behavior when IoT Service Platform is temporarily Unreachable or Offline

If the IoT Service Platform is temporarily offline, the IoT Device Application SHALL first diagnose if the communication issues to the server are caused by higher layer communications (TCP/IP, UDP, ATM...). Higher layers mechanisms SHALL then try to re-establish the connection with the server. This is done by assessing (and if necessary, attempting to re-establish) connectivity in a step-wise approach, top-down.

The IoT Device Application SHALL NOT frequently initiate an application-driven reboot of the communication module/chipset. The IoT Devices SHALL retry connection requests to the IoT Service Platform with an increasing back-off period.

If the IoT Device detects that the IoT Service Platform is back online, it SHALL employ a randomized timer to trigger communication requests to the mobile network.

Behavior when Coverage Lost (GPS, GLONASS, LAN, WAN)

When GPS, GLONASS coverage is lost, the IoT Device Application SHALL NOT reboot the communication module/chipset. The IoT Device Application SHOULD perform diagnostics, reboot the affected hardware element and send an alert to the IoT Server Application.

When LAN or WAN coverage is lost, the IoT Device SHALL NOT reboot the communication module/chipset. The IoT Device Application SHALL retry scanning to acquire mobile network connectivity with an increasing back-off period.

Behavior when Sensors / Actuators Malfunction

When in-built sensors or actuators malfunction, the IoT Device Application SHALL NOT reboot the communication module/chipset. The IoT Device Application SHOULD perform diagnostics, reboot the affected hardware element and send an alert to the IoT Server Application.

Behavior when Sensor Alarms / Actuators Triggered

When in-built sensors or actuators are triggered, the IoT Device Application SHALL NOT reboot the communication module/chipset. The IoT Device Application SHOULD instead send an alert to the IoT Server Application.

Behavior when Device Memory Full

When the IoT Device's memory is full, for example due to the amount of collected data or an unwanted memory leak, the IoT Device Application SHALL NOT reboot the communication module/chipset. The IoT Device Application SHOULD perform diagnostics, reboot the affected hardware element and send an alert to the IoT Server Application.

Behavior when Communication Requests Fail

The IoT Device Application SHALL always handle situations when communication requests fail in a way that does not harm the mobile network. The mobile network may reject communication requests from the IoT Device with a 3GPP error cause code (refer to GSMA TS.34). When the IoT Device Application detects that its requests are rejected, it SHALL retry connection requests to the mobile network with an increasing back-off period. The IoT Device Application SHALL NOT start an application-driven reboot of the communication module/chipset, in an attempt to ignore or override the mobile network's decision.

Communication requests from the IoT Device Application SHALL NOT be retried indefinitely – all requests must eventually time-out and be abandoned by the IoT Device Application.

Behavior when Device-Originated SMS are Barred

When the IoT Device Application detects that its subscription for MO-SMS is barred by the mobile network, the IoT Device Application SHALL retry connection requests to the mobile network with an increasing back-off period. The IoT Device Application SHALL NOT start an application-driven reboot of the communication module/chipset.

Reselection of Radio Access Technology Bearers

If the IoT Device supports more than one family of access technology (for example 3GPP, WLAN) the IoT Device Application SHALL employ a randomized delay before switching to a different family of access technology.

The IoT Device Application SHALL implement a protection mechanism to prevent frequent "ping-pong" between these different technologies. This is done by limiting the frequency of reselection actions, with appropriate hysteresis mechanisms.

Handling Loss of Service on Roaming Network

The IoT Device Application SHALL always be prepared to recover lost end-to-end connectivity while camping on a roaming network. This is implemented with a top-down, staged recovery algorithm diagnosing each protocol layer. In case of failing to re-establish one layer, the algorithm initiates the recovery procedure on the following protocol level below. This may be done, for example, as follows:

- Step 1. Re-establishment of higher layer connectivity, e.g. VPN tunnels, SSH sessions, etc.,
- Step 2. Re-establishment of the PDN connectivity or PDP context,
- Step 3. Re-attach (data) to the network,
- Step 4. Re-triggering of a plain network selection,
- Step 5. Complete reboot of the device.

All re-establishment procedures SHALL be implemented in a reasonable way avoid excessive signaling to the network. This may include usage of randomized triggers and incremental, back-off retry mechanisms. Threshold and timer values may depend on the IoT Service's requirements.

IPv4/v6 Dual Stack Support

The IoT Device Application SHALL support IPv4/v6 dual stack (PDN Type = IPv4v6) so that it can properly roam onto mobile networks having support for either IPv4 only or IPv6 only or dual stack only.

Release History

Publication Date	Version	Author
17.08.2018	2.0	Miguel Rodriguez (ITS-IVA)
02.05.2019	2.1	Miguel Rodriguez (ITS-IVA)