5. AN OVERVIEW OF THE TELSTRA NETWORK

5.1. Technologies and Features

Mobile networks evolve rapidly. Developers need to ensure they future proof their design in such a way to take advantage of new features as they become available.

The following table presents the technologies and features available on the Telstra Network (as of June 2020) along with future technology trends. Included alongside are relevant considerations for designing your wireless application.

Technology/Feature	Present	Future	Considerations
3G Frequency Bands Uses HSPA and HSPA+ technologies	• 850 MHz (B5)	• 3G currently has a large geographical footprint, no network expansion is planned. • Telstra will be switching off 3G in June 2024.	• No significant 3G design feature development is planned.
4G Frequency Bands Uses LTE technology	• 1800 MHz (B3) for coverage • 700 MHz (B28) for coverage • 2600 MHz (B7) for capacity and in-building coverage • 2100 MHz (B1) for capacity		• Note that the 700 MHz bands (B12, 13, 14, 17) used in the US are not compatible with the Australian 700 MHz band (B28) • Lower frequency bands work better in rural areas. • See link below for coverage map *
5G Frequency Bands	3.6GHz (n78)	• Low bands e.g. 850MHz (n5) & mmWave 26GHz (n258)	
LTE Carrier Aggregation (CA) Combines 2 or more carriers together to allow greater throughput	Support 2, 3, 4, 5 and 6 CA combinations of bands 1, 3, 3, 7, 7 and 28		• CA offers higher data rates that are suitable for large file downloads and video streaming applications • CA is limited to certain areas of the network
MIMO Multiple In Multiple Out. Using multiple antennas for the up and downlink of radio transmission can increase throughput or received signal quality	In downlink network supports: • 2x2 on all bands • 4x2 on B1, B3 and B7 • 4x4 on B1, B3 and B7		
LTE-M/NB-IoT Low throughput and low power cellular technology for IoT solutions	• 700 MHz (B28) for LTE-M, NB1 and NB2	• 1800 MHz (B3) may be considered for LTE-M	

When a device initiates PSM with the network, the device negotiates with the network how long the PSM period will be and the network retains state information and a reattach procedure is not required, even if the device awakes and sends data before the expiration of the time interval it agreed with the network.

As an example for a monitoring application, the device might be configured by an application to enable PSM, negotiate a 24 hour time interval with the network and provide a daily status update to a centralised monitoring point. If the device's monitoring application were to detect an alarm condition, irrespective of any agreed sleep interval, the application could wake the radio module instantly and send vital information without the need for a reattach procedure.

In a similar manner to a radio module that has been powered off, a radio module with PSM enabled cannot be contacted by the network whilst it is asleep. The inability to be contacted whilst asleep may preclude the use of PSM for some applications.

5.3.3. 3GPP Extended Discontinuous Reception (eDRX)

Extended Discontinuous Reception is an extension of an existing LTE feature which can be used by IoT devices to reduce power consumption.

Today, many smartphones use discontinuous reception (DRX) to extend battery life. By switching off the receive section of the radio module for a fraction of a second, the smartphone is able to save power. The phone cannot be contacted by the network while it is not listening but if the period of time is kept brief, the phone user will not experience degradation of service. E.g. If called, the phone might ring a fraction of a second later than if DRX was not enabled.

eDRX allows the time interval during which a device is not listening to the network to be greatly extended. For an IoT application it might be acceptable for the device to not be reachable for a few seconds or longer. Whilst not providing the same levels of power reduction as PSM for some applications eDRX may provide mechanism to deliver device reachability and power consumption.

5.3.4. 3GPP Enhanced Coverage

Some IoT applications require devices to be positioned in very poor radio conditions where the signal is extremely weak. For example, underground parking garages and in ground pits. The Enhanced Coverage feature has the potential to increase the depth of radio coverage to enable IOT devices to be placed and operate in locations that would otherwise not be possible.

The Enhanced Coverage feature, also called Coverage Extension, increases the power levels of signalling channels together with the ability to repeat transmissions. Through repeated transmission the ability of receivers to correctly resolve the message sent is improved. The trade-off is that repeating signal transmissions consumes additional power and the time between battery recharge or replacement may be reduced.

5.3.5. IoT Typical Usage by LTE Category

The following table is intended to help developers determine the most suitable device category for their IoT solution. You should choose a suitable device from the most appropriate device category to support the characteristics of your specific application.

LTE Category	> = 13	> = 1	1, M1	1, M1, NB1, NB2	M1, NB1, NB2
Device Data Usage	> 10 MB	> 1 MB	0.1 – 1 MB	< 0.1 MB (100 kB)	< 0.01 MB (10 kB)

6. CONSIDERATIONS FOR DEVICES INTEGRATING MODULES

6.1. Appropriate Technology Choices

6.1.1. Description

When selecting an embedded module for an integrated IoT/M2M device, a developer should take into consideration the module's supported cellular features and capabilities and the proposed application.

6.1.2. Methods

6.1.2.1. Appropriate Cellular Technology Choice

LTE only modules are available as well as multimode LTE + 3G modules. Telstra will not certify 3G only solutions.

6.1.2.2. Coverage / Radio Network Technology Support / Frequency Band Support

It is important to ensure that LTE coverage is available over the entirety of the expected usage areas. Telstra's LTE coverage is constantly expanding so developers should refer to our coverage maps for the latest information at: https://www.telstra.com.au/coverage-networks/our-coverage

Refer to the table in section 5.1 of this document for information regarding Telstra's current and future frequency band support for both 3G, 4G and 5G technologies.

6.1.2.3. Throughput performance

Choose a module based on your IoT/M2M solution's throughput and data usage requirements. See table in section 5.3.5 for guidance on choosing the appropriate LTE category based on typical IoT use cases.

As a rule, the higher the data throughput required, the faster device category should be selected. This will reduce the time connected to the network, which is the highest device power consuming state, and improves end user experience.

6.1.2.4. Choose devices /modules certified for use on Telstra's network

Certified modules have been tested by Telstra for compatibility with Telstra's network.

Approved modules and devices will typically have better longevity due to greater compatibility with our networks - features, technology and frequency bands.

For non-approved modules, and any devices integrating them, there can be no guarantee that they will work with our network currently or into the future.

For IoT/M2M integrated device approval, there is a streamlined process for devices integrating previously approved modules

6.1.2.5. Data only or Voice and Data

Note when selecting an embedded module - some modules only support data and others support both voice and data. If the application doesn't require voice, then a data only module is recommended as it will be cheaper and less complex.

6.1.2.6. FOTA (Firmware over the air)

FOTA is a requirement for new modules to be certified for use on our network.

Having the ability to address bugs and update devices remotely saves customers time and energy down the track. This is especially true for an IoT solution that may feature hundreds or even thousands of deployed devices as it would not be practical to physically attend to each device individually to install a software patch or update.

6.1.2.7. Module Radio Network Features

Choose modules that support important radio network features such as:

- **CDRX** (Connected Discontinuous Reception) – this is a device feature that allows the device to have micro sleeps. This feature allows the device to reduce battery consumption while minimizing impact to latency. CDRX is only invoked after 100ms of inactivity.
- **eDRX** (extended Discontinuous Reception) – this is an IoT device feature that requires network support. It allows the time interval during which a device is not listening to the network to be greatly extended, reducing battery consumption.
- **PSM** (Power Save Mode) – this is an IoT device feature that requires network support. The device will inform the network that it will be going into power save mode (using close to no power in this state) along with information about when it will 'wake up' for a short period to receive any messages that may be waiting.
- **RAI** (Release Assistance Indicator) – this is an IoT device feature that requires network support. It allows the device to indicate to the network that it is not expecting to receive/transmit any more uplink/downlink data. Upon receiving this signal the network moves the device into an idle state immediately rather than relying on a (10 seconds long) inactivity timer, reducing battery consumption.

6.2. Regulatory Considerations

When integrating a module into a device, regulatory requirements need to be met.

These are captured by the ACMA RCM - Regulatory Compliance Mark for the product (and embedded module).

The RCM indicates a device's compliance with applicable ACMA technical standards — that is, for telecommunications, radio communications, EMC and EME.

Refer: https://www.acma.gov.au/Industry/Suppliers/Product-supply-and-compliance/Steps-to-compliance/product-labelling

6.3. Antennas

- 3GPP defines how many antennas each LTE category shall support, and devices shall comply to these requirements.
  Reducing the number of antennas has an negative impact on the received signal which impacts performance and customer experience
- Antennas should support all frequencies bands supported by both the module and network
- Antennas should be optimized to suit the frequency bands to be used by the device.
  - For 3G devices they should be optimized for band 5 (850 MHz)
  - For 4G devices they should be optimized for all the bands they support and particularly bands 3 (1800 MHz) & 28 (700MHz)
  - For 5G devices they should be optimized for all the bands they support and particularly bands n78 (3.6GHz)

6.6. eSIM

An embedded SIM (eSIM) is a hardware secure element which holds the subscription profile of a mobile network operator that can be embedded/soldered into a device.

eSIMs are reprogrammable, enabling remote provisioning and management of services over the air.

eSIMs are supported by the Telstra Network. Contact Telstra by email at TelstraWirelessM2MHardware@team.telstra.com to discuss deployment and use of eSIMs in your device.

6.7. Ruggedness

- Ensure device is appropriate hardened / rugged against the elements for remote field deployment as appropriate.
- Ensure device has sufficient protection to prevent theft of UICC (SIM). Device should have a sealable, tamper proof enclosure

7. APPLICATION DEVELOPMENT TECHNIQUES

7.1. Best Practices for Development

Best practices that Telstra recommends when developing applications for use on our network are that applications should be designed to:

- Have interoperability / compatibility with the Telstra Network
- Minimize unnecessary data transfers
-
- Optimize any necessary data transfers
- Minimise unnecessary signalling overhead
- Be resilient to changing network conditions
- Be responsive
- Be secure
- Comply with industry and regulatory requirements
- Be serviceable
- Be lifecycle managed
- Conserve power

A series of techniques referencing industry standards and guidelines that can assist developers in implementing best practices outlined above are described in the following sections.

7.2. Fundamental Methods

7.2.1. Description

The GSMA Developer Guidelines outline a number of techniques to optimise the performance of smart phone applications with mobile connectivity, these techniques are equally applicable to IoT/M2M applications.

The guidelines recommend use of the methods listed below for developing the ideal mobile application, addressing many of the best practices listed in Section 7.1.

7.2.2. Methods

7.2.2.1. Asynchrony

To maximize user satisfaction, applications should be designed to be responsive which can be achieved using asynchronous logic for the main code block.

- Make use of separate parallel threads for independent network requests
- The main application thread handling the user interface should not be blocked by outstanding responses to network requests.
- Progressively load and present network response/data as it arrives to the user. Do not wait for all responses to return successfully before providing an update to the user.

7.2.2.2. Connection Loss and Error Handling

- **Request types**: Categorise network requests as user initiated (primary), non-user/system initiated and secondary (spawning from primary requests) to determine appropriate actions in event of network issues.
- **Cancellation**: Allow users ability to cancel primary requests. Cancellation of primary requests should result in cancellation of secondary requests.
- **Error handling**: Make use of notifications upon failure of primary requests. After attempting some limited number of retries, suspend the request and present the option to resume the request manually. See Section 0 for guidance on appropriate use of retry mechanisms.
- **Download resumption**: Divide large download files into chunks to make use of download resumption in event of network errors. This is an important mechanism to recover from interrupted file transfers rather than simply trying to download the entire file again.

7.2.2.3. Efficient Traffic Usage

- **Caching**: Keep a copy of the portion of data that has already been downloaded, in case it is needed again. Caching can reduce the need to reload images, web pages, style sheets, etc. which results in fewer data transfers, reducing network signalling and make apps appear faster and more responsive.
- **Cloud based transformations**: Avoid aggregation and processing of data from multiple data sources on the mobile application client. Instead perform these operations on an application server and expose its functionality as a web service via APIs to minimise the number of network connections and data transfer to the client.
- **Media transcoding**: Content optimization can minimize data usage and reduce download times. Developers should utilize the OS platform APIs/User Agent information to determine the device capabilities with regard to screen display resolution and streaming capabilities, and serve media accordingly. The lowest resolution/frame rate/codec rate that gives a good user experience should be used. Application Server should also have media content encoded in a variety of bit rates and the app should choose the media rate that suits the radio network being used.
- **Presence**: To minimise unnecessary traffic from presence based services information on presence or availability of users should be bundled before being published instead of sending/requesting an update per user separately. Make use of partial publication to only update information which has changed since the last state.
- **Email**: Consider imposing maximum attachment and message size limits to reduce the amount of data transfer. Provide users with a choice to download large attachments/messages instead of doing so automatically. Make use of Push notifications from server to device instead of polling to update messages.
- **Push notification**: Many applications attempt to deliver real time news, notifications and other data to devices by periodically polling the network which is wasteful if there is no new information on the server and causes unnecessary network signalling and drain on device battery. Instead push data to the device when there is actually new and relevant information available.
- **Compression**: Data compression where possible can be used to minimize data transferred over the network and reduce costs for the user. Applications that are text based and use HTTP protocols such as news aggregators lend themselves well to compression techniques, which can reduce text data size by 80%.
- **Data batching**: It takes time, power and network signalling to switch between device RRC (Radio Resource Control) states. When a device switches from an idle to dedicated channel to send data it consumes 60-100 times the amount of power it does in the idle state. By

7.2.3. References

Smarter Apps for Smarter Phones v4.0: http://gsmaterminals.github.io/Developer-Guidelines-Public

Implementation of these methods often requires use of platform specific APIs. Guides for Android and iOS platforms are linked below:

Google Android:	Connection management https://developer.android.com/guide/topics/connectivity Push notification https://firebase.google.com/docs/cloud-messaging/ Battery State http://developer.android.com/reference/android/os/BatteryManager.html
Apple iOS:	Connection management https://developer.apple.com/library/archive/documentation/NetworkingInternetWeb/Conceptual/NetworkingOverview/Introduction/Introduction.html Push notification https://developer.apple.com/library/archive/documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/APNSOverview.html#//apple_ref/doc/uid/TP40008194-CH8-SW1 Battery State http://developer.apple.com/library/ios/#documentation/uikit/reference/UIDevice_Class/Reference/UIDevice.html

time. These devices should not all try to reconnect to the network at the same time. Stagger the network activity of all the IoT/M2M devices so as not to contribute to network congestion.

Note Telstra's terms and conditions also mandate conditions around synchronized access to Telstra's network for multiple IoT/M2M modem devices and contains the following clause "If your Wireless M2M application employs more than 50,000 modem devices, you must provide a facility to control data transmission intervals in real time. We may require you to increase data transmission intervals during periods of network congestion".

Link to Testra's "Our Customer terms"
https://www.telstra.com.au/content/dam/tcom/personal/consumer-advice/pdf/business-b/dataservices-m2m.pdf

Developers should not schedule periodic reboots / power cycles of their devices, that is a behaviour that imposes strain on the network as it generates a large volume of network signalling, and the more devices in use the greater the load on the network. If device resets are required, then as described for software/firmware updates the resets should be randomized over a long period , a week or a month and they should be performed at off peak times such as between Midnight and 6am.

7.3.3. References

See Section 7 – Connection Efficient Requirements from GSMA IoT Device Connection Efficiency Guidelines v5.0:

https://www.gsma.com/iot/wp-content/uploads/2018/06/GSMA-IoT-Device-Connection-Efficiency-Guidelines_TS.34_v5.0.pdf

Refer *"Use of multiple modem devices"* in Telstra's *Our Customer Terms*.

https://www.telstra.com.au/content/dam/tcom/personal/consumer-advice/pdf/business-b/dataservices-m2m.pdf

See Appendix B – WIRELESS TECHNOLOGY INFORMATION for more details on the relationship between device state transitions, network signalling events and power consumption.

7.4. Firmware over the Air Updates

7.4.1. Description

Firmware over the Air (FOTA) upgrade is the ability of a device to have its firmware/operating system and RF chipset firmware upgraded using the cellular network ("over the air").

Note FOTA can be achieved by proprietary methods or standardized methods such as described by OMA (Open Mobile Alliance) Specifications body (http://openmobilealliance.org/) in which case it is using the OMA-DM (OMA Device Management) standard on FUMO (Firmware Update Management Object). Telstra has no preference for method used. Telstra expects the vendor/OEM or integrator to host the FOTA server for their devices.

Device vendors should keep up with latest OS and/or module firmware releases, rolling out updates as they become available to ensure new security vulnerabilities are addressed and known issues are minimised.

Note that Telstra reserves the right to insist on a firmware upgrade using this capability at any time should we find device issues causing network harm/other user harmful impacts.

IoT/M2M devices using a custom APN may be enabled to use IPv6 only (single stack) on a case by case basis. Telstra can be contacted by email at TelstraWirelessM2MHardware@team.telstra.com to discuss this if required. Telstra expects future large-scale deployments of IoT/M2M devices will be configured to use IPv6 only (single stack).

Telstra will be requiring module manufacturers to support IPv6 going forward.

IPv6 Support – Wireless Broadband Devices

Wireless broadband devices (USB Dongles, Wi-Fi hotspots, Gateways) will be required to support dual stack IPv4/IPv6 connections using the IPv4v6 PDP type.

7.5.2. Methods

Some basic app guidelines for IPv6 are:

- Don't hardwire IPv4 addresses in to your app / app code – use variables to represent IP addresses.
- Ensure when coding to use a variable for IP addresses that can hold an IPv6 address
- Ensure apps are both IPv4 and IPv6 compliant
- For IoT/M2M solution developers - use IPv6 capable modules in your application and particularly Dual Stack IPv4v6 capable embedded modules in your device. IPv6 capability will help future proof your device/application – increase its longevity and increase its security (if new security features of IPv6 are utilized).
- Developers using servers as part of their application should ensure that their servers are dual-stack or IPv6 enabled. The application should be designed to provide IPv4 End to End or IPv6 End to End, with IPv6 E2E being preferred.

7.5.3. References

IPv6 General Reference

Refer to Appendix C for an explanation of IPv6 terminology
https://www.internetsociety.org/deploy360/ipv6/faq/

XLAT464 References

http://tools.ietf.org/html/rfc6877
https://sites.google.com/site/tmoipv6/464xlat

Refer to RFC7849: An IPv6 Profile for 3GPP Mobile Devices for recommendations on connecting to IPv6 networks while also ensuring IPv4 service continuity https://tools.ietf.org/html/rfc7849

7.6. Follow Security Guidelines

7.6.1. Description

Developers need to consider security and privacy aspects of their application in order to protect their users and their data.

Security is also needed for Fraud prevention – given that these devices and their SIMS may be relatively accessible in high numbers.

Some specific security measures for IoT/M2M devices include:

- Firmware update capability (OTA) to allow device to be quickly patched should any security issues / vulnerabilities come to light.
- Ensure the physical security of the SIM in the device. For instance to avoid the oft-cited scenario where utility meters sim cards are stolen and used for data/call theft.
- Use IPv6 - due to its enhanced security features
- Utilize vendors FOTA to ensure you have the latest firmware for your device
- Review module and OS development platform security guidelines
- Ensure device has sufficient password protection / user authentication procedures to prevent against hacker access
- The physical security of devices when installing. Consider alarming device back to central server e.g. alarm if enclosure is opened
- Utilize external consulting/testing expertise against hacking/intrusion for critical IoT/M2M applications in utility and health monitoring areas.
- Consider hiding SSID for Wi-Fi connected devices. No need to broadcast.

7.6.3. References

OS Platform Security Guidelines
Each of the major mobile OS platforms has its own security guidelines for developers. These are a very good reference for developers.

Google Android:	http://developer.android.com/training/articles/security-tips.html
Apple iOS:	https://developer.apple.com/library/mac/documentation/Security/Conceptual/SecureCodingGuide/Introduction.html

Android Developer Security Tips (whilst specific to Android contains principals that are applicable to all platforms): http://developer.android.com/training/articles/security-tips.html

GSMA IoT Security Guidelines - Covers security considerations for IoT endpoints (devices), network elements and services ecosystem infrastructure: https://www.gsma.com/iot/iot-security/iot-security-guidelines/

7.7. Follow Privacy Guidelines

7.7.1. Description

Telstra respects user's privacy and it is important that developers ensure users privacy. There are legal penalties for not following the applicable laws.

7.7.2. Methods

- Ensure that no user information is sent to third parties without the user being clearly informed and opting in to the service (they must be clearly informed of where the data is going and how it will be used)
- GPS should not be used without the users consent to track / record user location
- Ensure security guidelines above are adhered to as these will help safeguard the users privacy
- Ensure your app complies with Australian Privacy Laws.

7.7.3. References

Australian Privacy Laws: https://www.oaic.gov.au/privacy/the-privacy-act/

http://www.futureofprivacy.org/wp-content/uploads/Best-Practices-for-Mobile-App-Developers_Final.pdf

7.8. Conclusion

In conclusion, developers should consider the guidelines mentioned throughout this section.

One thing to bear in mind is that the general principles apply to both data transfers and signalling that occurs due to network accesses and detaches.

There should be randomization, sensible back-off algorithms and non-infinite retries for all types of network use - both data transfer and network accesses. Appendix D – Network cause codes and device behaviour is highly relevant to this and details some common Network Cause codes and how devices should behave upon receiving them.

- Check that app performs as expected when there is a cabled connection to computer (tethered)
- Check interaction with peripherals such as SD card, Bluetooth accessories.
- Check that app performs as expected with above peripherals on/off/connected/not connected as appropriate.
- Check that app gives user appropriate indication of network connectivity, error conditions and ensure that there is a non-blocking UI. e.g. If no network connectivity the device doesn't flash up an error message and get stuck but rather allows the user to continue with activities within the app that don't require network activity.
- Ensure Security is tested. e.g. user authentication works as expected
- Does the app correctly handle interruptions e.g. from incoming calls.
- CPU usage of app is not excessive (most operating systems provide CPU monitoring tools)
- That the app performs as expected when the device is multitasking with other apps or when the app is background mode.

8.3. IoT/M2M Specific Testing Guidelines

Given the remote locations and rugged environments that IoT/M2M solutions can exist in testing must be all the more rigorous.

In addition to the testing described above the following additional testing is required for IoT/M2M:

- Test that FOTA solution works – test that both the application software can be updated and verify that firmware upgrade solution will work.
- Test that device is sufficiently rugged for planned deployment/usage. Is the device sufficiently hardened for the expected environment?
    - Test for heat, vibration, moisture, UV exposure, and generally adverse weather
- Test for failure conditions e.g. for power metering application, what happens if there is a power failure?
- Test that device over the air diagnostics work
- Verify that network reacquisition and retry algorithms function in a finite and non-aggressive way
- Test IPv6 functionality

8.4. References

http://www.appqualityalliance.org/online_testing_tool_and_best_practice_update

http://www.appqualityalliance.org/resources

9. APPENDIX A – TELSTRA WIRELESS AND IOT/M2M RELATED PRODUCT INFORMATION

9.1. Telstra Mobility Partners

Telstra has dedicated IoT/M2M product and solution teams to assist developers with integrating IoT/M2M solutions within our network. Email TelstraWirelessM2MHardware@team.telstra.com for assistance.

9.2. Telstra IoT Offerings

https://www.telstra.com.au/business-enterprise/products/internet-of-things

9.3. Telstra IoT Platform

https://www.telstra.com.au/business-enterprise/products/internet-of-things/capabilities/cumulocity

9.4. Telstra IoT Connection Management

https://www.telstra.com.au/business-enterprise/products/internet-of-things/capabilities/connection-management-platform

9.5. Telstra Mobile Assets and Workforce Enterprise Solutions

https://www.telstra.com.au/business-enterprise/solutions/mobility-solutions

9.6. Telstra Wireless Managed Data Networks – Wireless WAN

https://www.telstra.com.au/business-enterprise/download/document/business-mdn-wireless-wan.pdf

9.7. Telstra IP VPN information

https://www.telstra.com.au/business-enterprise/solutions/network-services/connectivity/vpn-service

9.8. Telstra Enterprise Support Contacts

https://enterprise-support.telstra.com.au/t5/tkb/communitypage

9.9. Telstra Mobile Phones

https://telstra.com.au/mobile-phones

9.10. Telstra Mobile Coverage

https://www.telstra.com.au/coverage-networks/our-coverage

10. APPENDIX B – WIRELESS TECHNOLOGY INFORMATION

10.1. RRC State Diagram

Network signalling is related largely to the RRC (Radio Resource Control) state transitions of the wireless device. It is important not to have unnecessary network signalling as this decreases the network's performance in terms of responsiveness and the amount of traffic it can handle for all users. It can therefore make an app appear slow or unresponsive and excessive state changes can quickly drain a devices battery.

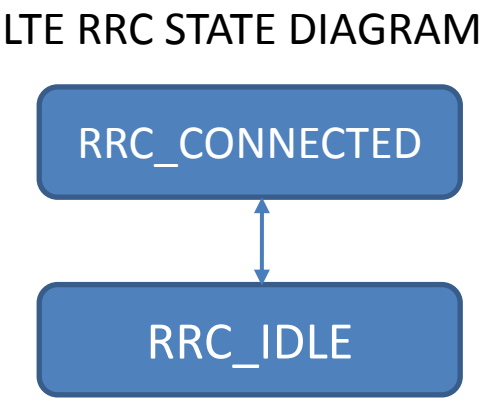
There are some app behaviours that cause unnecessary RRC state changes leading to too much signalling, which should be avoided, including –

- Having heart beats/pings to maintain always on activity
- Poor Network reattachment algorithms.
- Constant polling applications

RRC State Machine

The RRC state machine describes how a wireless device (4G) connects to the radio network in a variety of different states – which have different levels of connectivity, power consumption and throughputs.

This discussion will concentrate on the 4G network as this network has many advantages for users, developers and Telstra as an operator, over the older 3G technology and provides a better user experience.



As the device moves up the RRC states stack it consumes increasing amounts of energy, and the throughput available from the device increases. Note that these state changes require network signalling. There are also network controlled timers that control the minimum time before a state change can occur. It takes time to change state, and for small blocks of data the time to change state may actually exceed the time to transmit the required data. Therefore it is sensible to batch up small amounts of data into larger blocks.

LTE or 4G has two main states – RRC_CONNECTED and RRC_IDLE. An app that is in the connected state might typically consume a few watts whilst in IDLE state it will only consume tens of milli-Watts. The RRC_CONNECTED state is the state for data transfer between the network and mobile device. LTE is designed to allow the device to move between these two states very quickly especially when moving from IDLE to CONNECTED. Once in CONNECTED mode a network timer will again determine how quickly the device moves back to IDLE. So again to maximise device battery life it is important to reduce the amount of small data transfers to the network. The recommendation to buffer, collect and forward intermittently.

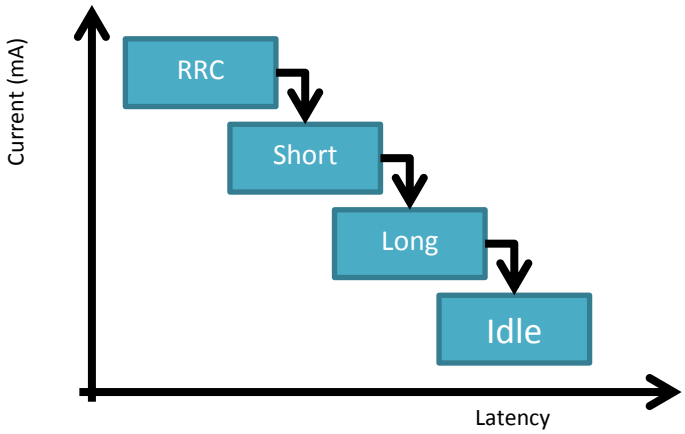
CDRX

CDRX stands for Connected Discontinuous Reception. It is an LTE device feature that allows the device to have micro sleeps and is controlled by the network. It helps to reduce network signalling load which benefits all users of the network.

It is a feature that reduces device battery consumption while minimizing impact on latency. In Telstra's network CDRX is only invoked after 100ms of inactivity.

The point to note is that the UE is still RRC connected while in CDRX sleep states.

The diagram below shows CDRX states for 4G



The point of all the above discussion is that it takes time, power and network signalling to switch between these RRC states and so it makes sense to minimize unnecessary state changes

10.2.Data Session Setup

Some brief information regarding data session on 4G/LTE.

LTE always has IP connectivity which is established by a **Default** Evolved Packet System (**EPS) bearer** by the Network Attachment procedure. This procedure attaches to the network and sets up a Default EPS bearer – and allocates IP address to the device.

When a device attaches to the LTE network for the first time it will be assigned a default bearer, which stays all the time and provides the always on IP connectivity. It has a nominal QOS.

If a specific QOS is required then this can be achieved by the network setting up a **Dedicated** Bearer. Currently Telstra does not yet have a specific QoS implementation for different applications / M2M so dedicated bearers should not be used for any purpose. UE initiated QoS is not supported and should not be attempted.

If an application requires a non-standard APN, then a new default bearer to the APN is established.

Developers should discuss any custom APN needs with Telstra via email address TelstraWirelessM2MHardware@team.telstra.com

A default bearer remains as long as the UE is attached to the LTE network. A UE can have additional default bearers. Each default bearer has its own IP address.

LTE dedicated bearers must be paired with a default bearer and they use the same IP address as the default bearer.

An LTE device can have up to 18 bearers in total – but one must be a default bearer. A default bearer is needed per APN (and another IP address).

IoT/M2M developers need to consider network connection setup procedure for 4G when developing their solution e.g. in the case of prompt for password applications on 4G where the device may connect to the network and fail if the correct password has not been previously entered.

Developers should consult with their vendor's documentation on relevant high level software commands or low level AT commands to perform these procedures as needed.

10.3.References

Refer 3GPP TS 36.331 for more information on RRC states of 4G devices:
https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2440

Refer 3GPP TS 24.301 for more information on EPS session management procedures:
https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=1072

11. APPENDIX C – IPV6 DESCRIPTION AND TERMINOLOGY

The drivers for the use of IPv6 and the description of IPv4 exhaustion have been well documented on the Internet, and will be assumed to be understood by device application developers – see the Wikipedia article http://en.wikipedia.org/wiki/IPv6 for an introduction to IPv6 if required.

IPv4 and IPv6 are distinct protocols that do not natively interoperate, but it can be assumed that both protocols will need to coexist in the Internet for many years to come. This means that any device will need to communicate with other devices that may themselves either speak only IPv6 or IPv4, either by itself natively speaking both protocols ("dual stack" configuration) or through a protocol translator or tunnel in the network or the device (e.g. NAT64, 464XLAT, etc.).

Single Stack

A single stack network or device as the name implies only supports a single IP protocol type. Single stack is often denoted simply as SS.

There are two possibilities:

1. IPv4 SS network, supporting only IPv4 traffic as most existing wireless networks are today
2. IPv6 SS network, supporting only IPv6 traffic

Dual Stack

A dual stack network is a network whose nodes are capable of processing IPv4 and IPv6 traffic simultaneously. It thus facilitates the transition to IPv6 while many devices and internet sites are still IPv4 by letting the two protocols co-exist.

A dual stack UE (user equipment aka mobile device), supports the following PDP types IPv4 (single stack), IPv6 (single stack) and IPv4v6 (dual stack or "DS" – that is an IPv4 and an IPv6 connection simultaneously).

A dual stack UE will request a dual stack bearer (IPv4v6) and the network will allocate the appropriate bearer which may be either IPv4 single stack, IPv6 single stack, or dual stack (both an IPv4 and IPv6 connection).

464XLAT

For handsets & tablets that are IPv6 single stack capable there will still be a need for many years to co-exist with the IPv4 ecosystem. E.g. to reach IPv4 sites, or for apps that are not yet IPv6 compatible (i.e. contain IPv4 literal addresses within their code).

RFC6877 464XLAT provides a solution to allow IPV4 services & applications to work over an IPV6 single stack network. The 464XLAT solution requires at a minimum a NAT64 in the network along with 464XLAT daemon/code running on the device.

464XLAT refers to architecture for both network and device to allow this to work and this is described in RFC6877. Refer http://tools.ietf.org/html/rfc6877. 464XLAT was first proposed by T-Mobile (Cameron Byrne) in partnership with NEC and JPIX in this RFC.

The 464XLAT code required by the device is open source code and is available to be used in any operating system.

12. APPENDIX D – NETWORK CAUSE CODES AND DEVICE BEHAVIOUR

The network has a variety of cause codes that it can send to the device in response to device requests that indicate a reason for failure of the request.

The device should pay attention to these cause codes and behave accordingly.

In a few cases the industry (3GPP) specifications and network timers specify the retry algorithm's behaviour but not always completely. Thus in these cases and where not specified at all, the behaviour is left to the device manufacturer/integrator. As discussed earlier, if retries are required, they should not be aggressive and infinite in nature.

Some cause codes indicate trouble with a user's service subscription and retries are pointless (once issue is confirmed as not a one off) – in this case the user/developer needs to confirm their service subscription status with Telstra. For 3G these cause codes are described in 3GPP specification 24.008 annexes.

Refer http://www.3gpp.org/DynaReport/24301.htm
There are three main categories of error codes. Those related to Mobility Management (MM), Call Control (mainly applicable to voice calling) and Data Session Management (SM).

Mobility Management (MM) includes causes related to:

- MS Identification,
- Subscription Options
- Network Failures/Congestion/Authentication Failures
- The nature of request
- Invalid messages
- GMM = related specifically to data

Call Control (CC) causes are grouped into:

- Normal Class
- Resource unavailable class
- Service / Option not available
- Service Not implemented class
- Invalid Message
- Protocol error
- Interworking issues

GPRS/Data Session Management (SM) [ESM used for LTE] are divided in to subgroups of causes related to:

- Nature of request
- Invalid Messages