Telecom Regulatory Authority of India

Public Open Wi-Fi framework

Architecture & Specification (Version 0.5)

12th July, 2017

Mahanagar Door Sanchar Bhawan, Jawahar Lal Nehru Marg,
New Delhi – 110002

Table of Contents

Introduction

The Internet is the single most self-empowering infrastructure available for a citizen in the 21st century. The World Bank observed that a 10% increase in internet penetration leads to a 1.4% increase in GDP. Access to the Internet is considered a basic human right by many countries globally, including Estonia, Finland and France. In India, access to data is still limited due to poor coverage of fiber/telecom and prohibitive pricing of cellular data.

WiFi is a complementary, not competing technology to LTE. Public hotspots hold an important place in the last-mile delivery of broadband to users. WiFi is much easier to scale than adding new LTE towers. It bolsters connectivity inside buildings, airports, etc. where LTE penetration is inherently limited. It allows for offloading from telecom networks to ease congestion, and will be crucial when the next billion IoT devices come online. Yet, there are only 31,000 public WiFi hotspots in India, compared to 13 million in France, and 10 million in the United States of America.

It is not enough to only install more routers. TRAI aims to offer a seamless experience to end users, both residents and international travelers. To provide a simplified, consistent experience across hotspots from various providers means unbundling authentication, payment and accounting from hardware and software running on the Access Point. This will allow small entrepreneurs such as tea shops, to set up and maintain Access Points. Whereas, device manufacturers, payment companies, ISPs/Telcos and Consumer Internet companies can provide the remaining pieces to set up Public Data Offices (PDOs).

The unbundling is also important from the point of view of scale. PDOs will be akin to the PCOs that connected all of India, even when tele-density was less than 7 telephones per 100 people. It is also suggested that the Public WiFi Hotspots store community interest data locally, and allow access to it through negligible costs. Overall, the introduction of public WiFi network, should encourage the PDOs to become bustling centers of economic activity.

TRAI has conducted multiple consultations regarding this which began in July 2016 and has released papers and notes regarding this. TRAI has also initiated a pilot in July 2017 to conduct field trials. All related documents are available on TRAI website.

Project Mission

The vision of this initiative is to establish an Open Architecture based **WiFi Access Network Interface** (WANI), such that;

1. Any entity (company, proprietorship, societies, non-profits, etc.) should easily be able to setup a paid public WiFi Access Point.
2. Users should be able to easily discover WANI compliant SSIDs, do one click authentication and payment, and connect one or more devices in single session.
3. The Experience for a small entrepreneur to purchase, self-register, set-up and operate a PDO must be simple, low-touch and maintenance-free.
4. The products available for consumption should begin from "sachet-sized", i.e. low denominations ranging from INR 2 to INR 20, etc.
5. Providers (PDO provider, Access Point hardware/software, user authentication and KYC provider, and payment provider) are unbundled to eliminate silos and closed systems. This allows multiple parties in the ecosystem to come together and enable large scale adoption.

Document Objectives

This document intents to provide detailed technology specifications for various providers to ensure full WANI system interoperability. All providers must ensure compliance with this specifications to be part of this initiative. This is a technical document and does not fully cover detailed policy aspects and enabling framework.

TRAI believes that through unbundling of services, multi-provider ecosystem, and easy regulatory process, millions of WiFi access points can be enabled across the country that allows users to connect via single-click authentication and use it with ease.

NOTE: This is a draft specification which may undergo changes before becoming final specifications based on feedback from ecosystem during pilot.

Glossary of Terms

PDO	Public Data Office
PDOA	Public Data Office Aggregator
APP	Application – mobile app provisioned as frontend for users to access and connect to the available WiFi hotspots
AP	Access points distributed across the city
IP	Internet protocol address assigned to all the elements in the architecture
JSON	JavaScript Object Notation
URI/URL	Uniform Resource Identifier/Locator
CP	WiFi Captive Portal
OTP	One Time Password
SSID	Service Set Identifier
MAC	Media Access Control – A globally unique ID/address given to physical network devices.
ACCESS POINT	Wireless hardware device that allows other devices to connect over WiFi to a network/Internet.
HOTSPOT	A physical location where WiFi Access Point is available for people to connect to Internet.

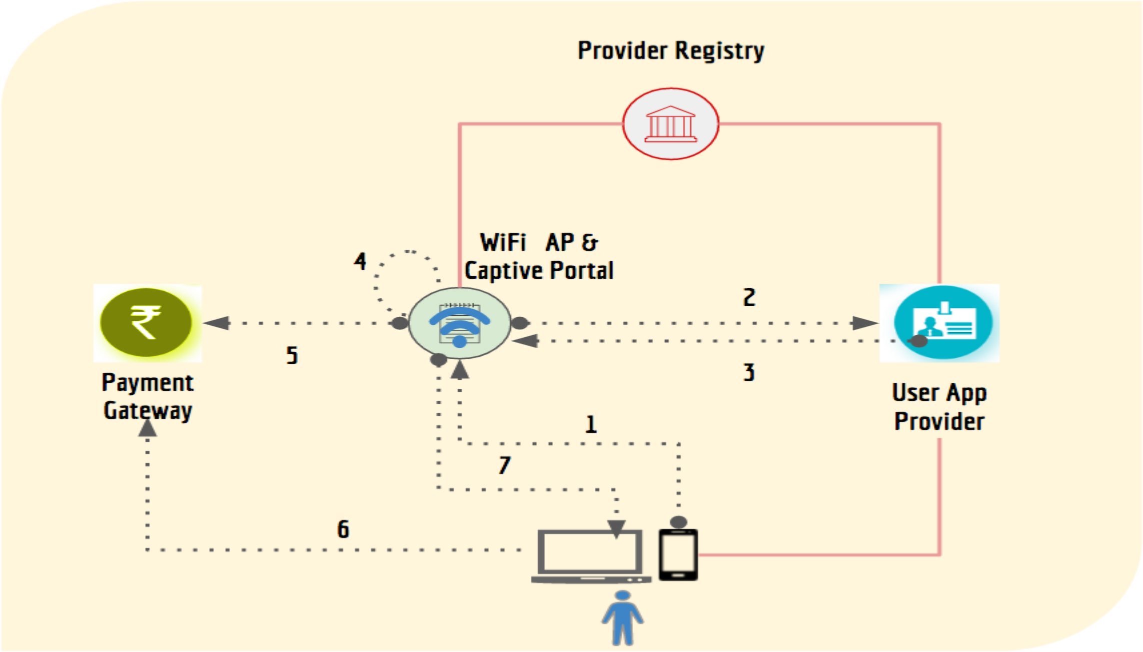
Detail Specifications

High Level Architecture

Players in the ecosystem

- **PDO/PDOA**: Any Indian entity (companies, associations, small merchants, etc.) having a PAN number wanting to provide one or more WANI compliant WiFi hotspots to public using either free or paid model. They conform to the governing rules laid out by TRAI under this framework.
- **Hotspot Hardware/Software/Service Provider**: Any software or service provider who is providing necessary software, hardware, services, and/or support for PDOs to setup WANI compliant WiFi hotspot. These can be any software/service provider, either Indian or global. It is expected that these providers will offer a WiFi-in-a-box solution for PDOs. Their software will need to be compliant to specifications laid out in this document. They will also integrate with a bank or a payment gateway for collecting payment from user.
- **User App Provider**: Any company providing a software application and backend authentication infrastructure for users to signup, discover WANI compliant WiFi hotspots, and do single-click connect from within the app. This app allow users to create a profile, do their KYC (mobile verification), and allow setting up preferences for MAC-IDs for various accessing devices and payment methods. This app should allow users to discover WANI compliant hotspots and connect to it. In addition, App Provider must offer a backend user authentication service that is called by WiFi Captive Portal software whenever user connects to obtain a signed user profile.
- **Central Registry of Providers (**or simply **Provider Registry)**: A central registry managed by DoT/TRAI or an entity approved by DoT/TRAI containing information about the PDOs/PDOAs, and User App providers in a digitally signed XML format. This is a relatively static registry where approved providers are allowed to manage their profiles. Actual specification of the registry is provided later in this document.

High Level Flows



One Time Flow

One time flows are depicted in red lines in above diagram.

- PDO/PDOA completes Self-Registration with Provider Registry using their public certificate (for signature validation). They also register their WiFi Access Points, SSIDs, and locations.
- User App provider is also registered with Provider Registry along with their authentication URL and public certificate (to validate their digital signature).
- User completes one time KYC with App Provider through their App. User App caches trusted SSIDs from Provider Registry from time to time.

Usage Flow

Usage flows are depicted in dotted lines in above diagram. Bullet number below corresponds to the number depicted within the diagram above.

1. User opens the App in which user has already registered and allows discovery and connection to WANI compliant WiFi access points. Within the app, user browses for nearby WANI compliant SSIDs and then chooses one SSID to connect to.

2. WiFi Captive Portal of the PDO initiates user authentication with App provider backend using the token passed from the app.
3. App provider backend returns a signed user profile token back to WiFi Captive Portal.
4. WiFi Captive Portal displays data packs available with their charges. User selects desired data sachet, click to confirm the terms.
5. WiFi Captive Portal sends request for payment through their payment gateway.
6. User completes payment.
7. PDO activates all device MAC-IDs that were part of the signed profile and allows them to connect to the session without additional authentication. Pack is activated and user can begin browsing.

Specifications

Following sections describe the technical specifications for Provider Registry, user signup, user authentication, and usage. Providers must ensure they comply with these specifications for ensuring interoperability across the country.

Provider Registry

Provider registry is maintained by DoT/TRAI for ensuring all authorized providers are identified, discovered, and trusted by the ecosystem. Providers will be given an account on the site where registry is maintained for managing their profile, public keys, and other details.

Currently the Provider Registry XML will be made available on the following URL:

```
https://trai.gov.in/wani/registry/wani_providers.xml
```

This registry XML will be updated whenever data changes in provider database. Applications reading this and caching the registry should respect the “`ttl`” (Time to Live) parameter and ensure it is refreshed to get latest data. It is also critical to ensure sub-registries linked via the main registries also need to be downloaded based on the need.

Schema (XSD will be made available separately) for wani_providers.xml is:

```
<WaniRegistry lastUpdated="" ttl="">
      <PDOAs>
            <PDOA id="" name="" phone="" email="" apUrl="" status="" rating="">
                  <Keys>
```

```
                                <Key exp="">base-64 encoded public key</Key>
                        <Keys>
                </PDOA>
        </PDOAs>
        <AppProviders>
                <AppProvider id="" name="" phone="" email="" authUrl="" status=""
        rating="">
                        <Keys>
                                <Key exp="">base-64 encoded public key</Key>
                        <Keys>
                </AppProvider>
        </AppProviders>
        </Signature>
</WaniRegistry>
```

Element/Attribute	**Description**
WaniRegistry	Root element of the registry
WaniRegostry→lastUpdated	Timestamp in YYYYMMDDhhmmss format providing when the registry XML was last updated. Useful for cache refresh.
WaniRegistry→ttl	Time To Live in hours suggesting how long this data should be cached before checking for change. Default is "24".
WaniProvider→PDOAs	Parent element for listing of all PDOAs.
PDOAs→PDOA	Repeating element providing one entry per PDOA.
PDOA→id	Unique provider ID within the registry.
PDOA→name	Name of the provider entity.
PDOA→phone	Contact number of the provider entity.
PDOA→email	Email of the provider entity.
PDOA→apUrl	URL to the signed XML where all WiFi Access Points of this providers along with MAC-IDs, and location is listed. This list is grouped by location to make it easier for applications to cache parts of this. At a later point, when the number of entries are in millions, this list itself may be further split with URLs pointing to sub-lists. Applications should start from this main registry and use the URLs within these XMLs to auto navigate the complete registry.
PDOA→status	Current status of the provider. Valid values are INPROCESS, TEMPORARY, ACTIVE, INACTIVE, SUSPENDED, BLACKLISTED.
PDOA→rating	User rating of the provider. This is a decimal value between 0 and 5. This is meant for future use.
PDOA→Keys	Parent element where public keys are listed.

Element/Attribute	Description
`Keys→Key`	Individual public keys to validate the signature of the PDOA. When integrating via APIs across ecosystem partners, it is necessary to sign the API requests and responses to establish trust. This element will contain the base-64 encoded certificate in X509 V3 format. Currently SHA256withRSA (2048 bit key) is the supported signing algorithm.
`Key→exp`	Expiry of the key in YYYYMMDD format. This is provided to support co-existence of multiple keys and is required for key rotations.
`Waniprovider →AppProviders`	Parent element for listing of all user application providers.
`AppProviders →AppProvider`	Element representing individual app provider.
`AppProvider→id`	Unique id of the app provider within the registry.
`AppProvider→name`	Name of the app provider.
`AppProvider→phone`	Contact number of the app provider.
`AppProvider→email`	Email of the app provider.
`Appprovider→authUrl`	Authentication URL (API endpoint) of the app provider against which WiFi Captive Portal will call to authenticate and obtain the signed user profile. This must be an HTTPS URL into which authentication input data can be sent in the body.
`AppProvider→status`	Current status of the provider. Valid values are INPROCESS, TEMPORARY, ACTIVE, INACTIVE, SUSPENDED, BLACKLISTED.
`AppProvider→rating`	User rating of the provider. This is a decimal value between 0 and 5. This is meant for future use.
`AppProvider→Keys`	Parent element where public keys are listed.
`Keys→Key`	Individual public keys to validate the signature of the AppProvider. When integrating via APIs across ecosystem partners, it is necessary to sign the API requests and responses to establish trust. This element will contain the base-64 encoded certificate in X509 V3 format. Currently SHA256withRSA (2048 bit key) is the supported signing algorithm.
`Key→exp`	Expiry of the key in YYYYMMDD format. This is provided to support co-existence of multiple keys and is required for key rotations.

WiFi Access Points (pointed by `"apUrl"` parameter of the PDOA) XML format:

```
<WaniAPList lastUpdated="" ttl="" providerId="">
      <!-- location element repeats -->
      <Location type="DISTRICT" name="" state="">
            <AP macid="" ssid="" status="" rating="" geoLoc="">
                  <Tag name="OPENBETWEEN" value=""/>
                  <Tag name="AVGSPEED" value=""/>
```

```
                    <Tag name="FREEBAND" value=""/>
                    <Tag name="PAYMENTMODES" value=""/>
             </AP>
       </Location>
       </Signature>
</WaniAPList>
```

Element/Attribute	Description
WaniAPList	Root element of the registry where all WANI compliant WiFi Access Points are listed for a provider.
WaniAPList→ lastUpdated	Timestamp in YYYYMMDDhhmmss format providing when the registry XML was last updated. Useful for cache refresh.
WaniAPList→ttl	Time To Live in hours suggesting how long this data should be cached before checking for change. Default is “24”.
WaniAPList→providerId	Id of the provider. This is same id as the WaniRegistry XML.
WaniAPList→Location	Repeating element organized by location of the Access Point.
Location→type	Type of location used for grouping. Currently it will be grouped by DISTRICT. In future further grouping may be supported.
Location→name	Name of the location which is used for AP grouping. Currently this will be name of the district.
Location→state	Name of the State in which this Access Point is located.
Location→AP	Element depicting one Access Point. This element repeats.
AP→macid	MAC-ID of the Access Point.
AP→ssid	SSID of the Access Point.
AP→status	General status of the AP. Valid values are ACTIVE, INACTIVE.
AP→rating	User rating of the provider. This is a decimal value between 0 and 5.
AP→Tag	Various tags describing the AP features. • OPENBETWEEN – value should be in the format hh-hh where hh represents time between 00 and 24. E.g., 09-17 (depicting 9 am to 5 pm) or 00-24 (depicting 24 hr availability). • AVGSPEED – Average speed in Mbps of the AP. It should be a positive integer. E.g., 2 meaning 2 Mbps. • FREEBAND – If this AP offers any free band in minutes. E.g., a value 10 depicts 10 free minutes. A Special value -1 depicts ALWAYS free. • PAYMENTMODES – Allowed payment modes. Values can be CASH, COUPON, CREDITCARD, DEBITCARD, NETBANKING, UPI, and WALLET. More enumerations may be added based on RBI approved payment schemes in India.

User Signup and Profile Management

Users are expected to use some software application (mobile/desktop/etc.) provided by the "App Provider" for user signup, KYC, and profile management. User App should provide the following key features during user signup and profile management:

1. Users install an app from the App Provider.
2. App MUST capture user mobile number and does a mobile number verification (via OTP or GSM Mobile Connect or any other mechanisms).
3. App also allows creation of mandatory "username" which is unique within the App Provider system. This is shared with WiFi provider during authentication and used for audit and traceability.
4. App should allow user to setup profile with additional **optional** attributes:
    a. Email – user should be able to optionally setup email for getting alerts, etc.
    b. Preferred payment address – This is ONLY for capturing UPI or Wallet address in the form **`upi://vpa/token`** (VPA is Virtual Payment Address for UPI collect transaction) or **`wallet://acc-no@ppi/token`**. App provider MUST NOT capture or store ANY sensitive information such as credit card number. All other types of payment will be directly handled by WiFi Captive Portal.
    c. If the User App is also a payment app (like UPI/Wallet app), then additional optional `token` string can be used to provide auto-deduct/offline/other additional payment functionalities.
5. App MUST also allow users to easily add/remove devices (MAC-ID and a name) which they want to connect to various Wi-Fi hotspots.
    a. This allows the user to have more than one device to be connected to WiFi hotspots within same session.
    b. This is critical to allow IoT devices used by user to also connect using the common app and authentication. For example, by connecting the mobile phone to the WiFi network, user may also connect his/her laptops or connected cars or other future devices.
    c. Optionally app may also provide "device group" profiles to allow users to define named group of devices so that they can choose one group vs another during connecting.

Access Point Discovery

1. User App should allow users to discover nearby WANI compliant Access Points by detecting nearby SSIDs and verifying the MAC-IDs against the SSID Registry.
2. In addition, optionally user App can provide location specific searches and allow users to discover "nearby" WiFi hotspots without being the WiFi range. SSID registry can be cached locally by app smartly for doing location level searches.
3. App should also optionally allow users to save "favorites", "most recent", etc. for easy selection of regular connections.
4. In addition, ideally App may also provide easy sorting and selection of access points based on the "`Tag`" attributes such as when AP is available, average speed, rating, etc. This allows users to select best AP within available selections.
5. App must provide a mechanism for users to rate the access points and providers.

Connecting to Access Point and Usage

1. Whenever users want to connect to public Wi-Fi hotspot using this scheme, they can open their App, browse WANI compliant Wi-Fi hotspots (see section on discovery above), and click connect.
2. App creates a token **`waniapptoken`** which needs to be passed to WiFi Captive Portal. This token is created as below:

   ```
   waniapptoken = <app-provider-id>|<enc-token>
   ```

   `enc-token` MUST NOT be a fixed value to ensure it is not can be reused beyond a session. It MUST be encrypted using App provider public key so that only App provider backend can decrypt tis token. It is created as below:

   ```
   enc-token = base-64(RSA-Encrypt(token))

   token = {
       "ver": "1.0", // version of the token structure
       "timestamp": "YYYYMMDDhhmmss",
       "username": "", // username of user
       "appId": "", // App id to handle multiple apps from same provider
       "appVer": "", // version of app to handle multiple app versions
       "totp": "", // TOTP generated by the app. This is essential to
              ensure App provider server can trust origin of this token
       "custData": {} // any custom JSON data structure needed by the app
   ```

```
}
```

3. App base-64 encodes the token and passes it on to captive portal using parameter name **`waniapptoken`** (can be passed as part of GET parameter). E.g.,

   ```
   http://portal.com/?waniapptoken=
   FG23A|ZDM3MzQxM2RlYjc0NGIyNGM2MjI2MTM2MTY0MGVmN2Q3MGI4YjcxZjlmMTMyOTQ4NzdmNmY5O
   WViZjFlNTk3Yg==
   ```

4. WiFi provider's Captive Portal should look for `waniapptoken` parameter and process it as below:

   a. Extract the App Provider ID from the token prefix (string until the "|" delimiter within the token).

   b. Verify the App Provider ID against the locally cached WANI Registry (`WaniRegistry→Appproviders→Appprovider[id={id}]`) and obtain `authUrl` for that App provider.

   c. Encrypts the `waniapptoken` using PDOA private key as below to create a new token **`wanipdoatoken`**

   ```
   wanipdoatoken = <PDOA-Id>|<key-Exp>|<base-64(RSA-Encrypt(waniapptoken))>
   ```

   d. Calls the `authUrl` by passing the signed token `wanipdoatoken` as part of the URL parameter. This MUST BE an https call.

   ```
   https://auth.app-provider.com/?wanipdoatoken=
   12GF34|MjAxODA4MTV8NTlGMDIyOTM5NTdBRTI4N0Q3RDdBOTFEMEU1OEU2RTQ3OUU4NDAzRk
   IyMDQwM0U5N0ZGQzQ1RTE1RDRBMjcwMw==
   ```

5. App Provider backend server should do the following on their server:

   a. Extract the PDOA-Id from the parameter (token prefix).

   b. Verify the PDOA ID against the locally cached WANI Registry (`WaniRegistry→PDOAs→PDOA[id={id}]`).

   c. Once verified, take the public key of the PDOA corresponding to the key-exp parameter from WANI registry (`WaniRegistry→PDOAs→PDOA[id={id}]→Keys→Key[exp={key-exp}]`)-

   d. Decrypt using the `waniapptoken` from the `wanipdoatoken` using the public key of the PDOA.

   e. Decrypt `waniapptoken` using their own private key and verify the token structure, TOTP, etc.

6. After validation of the waniapptoken, App Provider should return the following structure back to WiFi Captive Portal:

```
{
   "ver": "1.0", // version of the profile format
   "app-provider-id": "", // mandatory - ID from WaniRegistry
   "app-provider-name":"", // name from WaniRegistry
   "timestamp": "YYYYMMDDhhmmss", // current timestamp
   "Username":"", // mandatory
   "payment-address":"", // upi://vpa/token or wallet://ac-no@ppi/token
   "Devices":[], // device MAC-IDs array if any for current session
   "signature":"", // computed for this structure (see below)
   "key-exp": "" // Key→exp value of the key pair used for signature
}
```

   Signature is computed as below:

```
signature = base-64(RSA-Encrypt(hash))
hash = SHA-256(timestamp+username+payment-address+devices[0]+…+devices[i])
```

7. Once Wi-Fi hotspot provider obtains response, it needs to do the following verification:
   a. Decrypting the hash from signature using the public key of the App provider (that corresponds to Key→exp value from registry).
   b. Calculate the hash and verify if the hash is matching.
   c. If matching, proceed with next steps. If not, show error and allow user to disconnect and connect again (try again).
8. After verification, Captive Portal should show the user available packages. Once the user chooses a package, user should be directed to make payment on the portal.
9. If user profile had preferred payment address, then it should be defaulted and allow user to do payment without any data entry on the portal.
10. WiFi provider will have to allow user to make payment during which time user must be given temporary Internet access to payment provider's server.
11. Once payment is confirmed, WiFi Access Point should now allow all devices (MAC-IDs within user profile) to be connected to same session and share the package. This is critical for single-click access to Internet for multiple devices that users typically carry around these days without each devices having to go through same process. This is a MANDATORY compliance requirements for WANI PDO/PDOAs.
12. When the session is about to expire, hotspot provider can prompt the user and requests extension of the session and charge additional amount ONLY WITH explicit user consent without user having to go through all steps again.

a. Note that users who connect to their favorite WiFi hotspots can "pre-authorize" payment through Wallet or UPI e-mandate (part of UPI 2.0) which makes even payment a single click seamless experience. This also allows WiFi providers to easily extend user sessions with single user click "extend my session (charge Rs.xx)" without any further steps to make payment.

IMPORTANT NOTE: With single click user authentication through authorized Apps and payment pre-authorization via e-mandates, connecting and using public WiFi will be a seamless, friction free experience for users.

Compliance Aspects

WiFi Provider

1. Captive portal must allow standard connection and authentication as per this specification.
2. WiFi Provider must provide choice to user to select a package with clear details of the package.
3. Captive portal should respect and handle preferred payment scheme for users and allow seamless collection of payment once the package is selected.
4. WiFi provider must comply and be certified with regulatory and security rules for payment transactions, auditing, and storage/handling of any sensitive payment information.

App Provider

1. App provider must provide an App to user (for any device/OS based on market needs) and comply with user sign up, profile management, and authentication specifications as per this document.
2. App provider must ensure user data is strongly protected to ensure user privacy and data security is ensured.
3. App provider must have a mechanism to allow regular app update and improvements.

4. App is encouraged to provide good user interface for consumers to easily discover, search, find best access points, and connect to it with single-click.

CONCLUSION

Telecom industry is seeing rapid transformation through drop in data prices, increased speed, and increased consumption of data packs. India is also creating a slew of digital platforms to help its citizens with better access to various services. According to reports, Indians consumed more cellular data than China, and as much as the USA in the current cellular data pricing regime. TRAI believes that by adopting an Open Architecture approach, emphasis on innovation and consumer experience is placed as the winning criteria.

This document provide technical architecture specifications for an interoperable ecosystem. The WiFi Access Network Interface (WANI) represents an exciting opportunity to do for data what PCOs did for Long Distance Calling. It will bring a new generation of users and entrepreneurs into the market to bridge the need of last mile connectivity. The opportunities created are immense and will benefit 100's of millions of users in India waiting to get affordable access to Internet.

----- END -----