# IOT SOLUTION FOR TOWER-X (Cell Tower management system)

### Tower management system (Telecom TMS)

IOT employs proactive approach to management by enabling increased efficiency, better forecasting, and higher levels of service. Apart from the prime job of maintaining uptime of telecom towers and supporting logistics, it is also now shifting their focus to curb inefficiencies and check rising operating costs.



**Existing Network** 

### **TOWER X- Our solution for TMS**

The bottom line of Tower Monitoring Solution (TMS) is that telecom service providers can gain visibility into tower performance, transforming the way in which they operate by reducing operating expenditure and keeping profits up. Sensors mounted anywhere from door, fuel tank, to camera collect and transmit real-time data to the central management system, giving operators a complete picture of their current operational condition.

With Tower X (TMS), we can do the following:-

- •Maintain Service Level Agreements(SLAs) with minimal capital expenditure
- Ensure smooth operations of asset-intensive remote sites
- Monitor fuel filling and avert fuel pilferage
- Cut operational expenditure (opex) to remain competitive
- Reduce environmental damage by cutting down carbon emissions

## **Elements of Tower X**

#### •Energy Management System

• Monitoring all electrical parameters like voltage, current ,energy, power factor, frequency, harmonics etc.

#### •HVAC Management:-

 Monitor Temperature and humidity to determine the optimal environmental settings

#### Surge Protection Device Monitoring:-

Many Parameters like Earth Line Current, Earth Line Peak Voltage on Surge event
Alarm/ Alerts Via Mobile/ Email/ Dashboard

#### Security Management:-

Surveillance of the whole tower site with the help of cameras and motion detection sensors
Biometric/ RFID based Access control for the site with proper log of Entry and Exit

#### •DG Monitoring:-

- •Fuel Level Sensors Helps us know the Fuel Levels
- •Viscosity Sensors helps us in Alerting if Fuel is replaced with some other Liquid

#### Power Source Monitoring:-

•Alerts over Events Triggered Via SMS/ EMail/ Dashboard etc.



copyright (c) elitiatech 2016 - All Rights Reserved

### List of Sensors used

#### SENSOR

- •Fuel sensor
- Motion detector
- Door sensor
- •Fire sensor
- Leakage sensor
- Temperature sensor
- Vibration sensor
- •AC/DC current sensor

#### CAPABILITY

Tracks fuel level and detects pilferage Identifies unauthorized movement in site Restricts authorized access into the cell tower site Alerts during smoke/ fire

Detects liquid or fuel leaks

Maintains cell tower site at optimum temperature Detects tampering of devices

Determines the quality of power generated



# **TOWER X PROCESS FLOW**



### **Process Flow Description**

#### Infrastructure worker role services

Setting up Environment:

Create:

-> Message Queue

-> Mongo dB Collection

A persistent object store maps between objects in your application and records in a persistent store

-> Blob storage

Blob storage is a service that stores file data in the cloud. Blob storage can store any type of text or binary data, such as a document, media file Open:

-> Sql server

#### **Role entry services**

Entry point that establish (Bidirectional) communication link between (Server and Client) running on the same network that listen the Data packet Message (string  $\rightarrow$  collection of bytes) -> Perform Checking whether Data packet Message IS (GPRS Packet Type OR SMS Packet Type)

#### Cache provider

A cache provides high throughput, low-latency access to commonly accessed application data, by storing the data in memory. For a cloud app the most useful type of cache is distributed cache, which means the data is not stored on the individual web server's memory but on other cloud resources, and the cached data is made available to all of an application's web servers.

#### Message queue

**Messaging** is a way of *asynchronous* communication of applications running on same or different machines with reliable delivery. Programs communicate by sending packets of data called messages to each other. A message may be a string, a byte array, an object... etc. Typically, a **sender (producer)** program creates a message and pushes it to a message queue and a **receiver (consumer)** program gets the message from the queue and processes it.



#### WORKER ROLE AND WEB ROLE SERVICES

A **Web Role** (consisting of 1 or more instances) that hosts a website or an API and is accessed by the application's users. The web role can perform simple tasks synchronously, but whenever any complex processing is required it will create a message and drop it on a Queue.

->One or more Queues, which support asynchronous communication between the Web Role and the Worker Role.

->A **Worker Role** (consisting of 1 or more instances) which pull messages of the Queue and perform slow or complex processing tasks.

->Storage (such as Tables, Blobs or SQL Database) that stores the system's state and typically includes the result of the Worker Role's processing tasks.

## Why HANA for IOT

SAP HANA native + Java /.NET + HTML5/SAPUI5 apps on standards-based development Environment.

Powerful and standards-based services to securely:-

- 1. Leverage the speed of HANA
- 2. Connect cloud apps with on-premise & other cloud systems
- 3. Build scalable & attractive sites on SAP HANA/SAPUI5
- 4. Integrate on-premise & cloud with SAP HANA Cloud Integration
- 5. Use and build extensions to other SAP and non-SAP software
- 6. Build collaborative applications
- 7. Connect with IoT devices

# Front ending with SAPUI5

#### Dashboard

Dashboards help the users to monitor the Site's health and Performance by displaying visual information about all the sites Three different dashboards which can be incorporated are:-



Some of the information which can be displayed using SAPUI5 are:-

- Uptime and Outage duration details
- Power and Fuel Cost details
- Faults trend
- Manpower mapping
- •Fuel Consumption details
- •Preventive Maintenance tickets details
- •Total run hours of all the sites and many more details