UNIONE EUROPEA
Fondo Sociale Europeo
Fondo Europeo di Sviluppo Regionale

Ministero dell'Istruzione, dell'Università e della Ricerca

PON RICERCA E INNOVAZIONE 2014 - 2020

Avviso 1735 del 13.07.2017 MIUR

Progetti di Ricerca Industriale e Sviluppo Sperimentale nelle 12 Aree di Specializzazione individuate dal PNR 2015-2020

Definizione degli Standard e dei Protocolli di Comunicazione

Rapporto Tecnico D5.5

Avviso	Avviso 1735 del 13.07.2017 MIUR
Codice progetto	ARS01_01259
Nome del progetto	Community Energy Storage Gestione Aggregata di Sistemi di Accumulo dell'Energia in Power Cloud
Acronimo	ComESto
Documento	D5.5
Tipologia	Rapporto Tecnico
Data di Rilascio	22/04/2021
Obiettivo Realizzativo	OR5
Attività Realizzativa	A 5.5
Soggetti Beneficiari Proponenti	UNICAL, TIM, SPINTEL, E-DISTRIBUZIONE
Elaborato (Nome, Cognome – Soggetto Beneficiario)	Floriano De Rango, Nicola Sorrentino–UNICAL; Caterina Cippone, Giacomo Morello – TIM; Pasquale Cucunato, Francesco Noto – SPINTEL; Maurizio Cinus, Roberto Infantino – E-DISTRIBUZIONE
Verificato (Nome, Cognome – Soggetto Beneficiario)	Floriano De Rango, Nicola Sorrentino – UNICAL
Approvato (Nome, Cognome – Soggetto Beneficiario)	Membri del PEB

Indice

Indice delle figure

Indice delle tabelle

Abbreviazioni ed acronimi

Abbreviazione/Acronimo	Testo Esteso
AGM	Absorbent Glass Mat
AMI	Advances Metering Infrastructure
ANIE	Federazione Nazionale Imprese Elettrotecniche
APAC	Asia Pacific
API	Interfaccia di Programmazione
ARERA	Autorità di Regolazione per Energia Reti e Ambiente
ASE	Altri Sistemi Esistenti
BEV	Battery Electric Vehicle
BNEF	Bloomberg New Energy Finance
CAGR	Compound Annual Growth Rate
CAPEX	CAPitalEXpenditure
CATI	Computer Assisted Telephone Interview
CAWI	Computer Assisted Web Interviewing
CAES	Compressed Air Energy Storage
CE	Consumer Empowerment
CES	Community Energy Storage
CoAP	Constrained Application Protocol
CONOE	Consorzio Nazionale raccolta Oli Esausti
CORE	Constrained RESTful Environment
CPS	Cyber Physical Systems
CPU	Central Processing Unit
CPV	Concentrated Photo Voltaic
CSP	Concentrated Solar Power
CT	Current Transducer
DB	Digital Business
DoD	Depth of Discharge
DR	Demand Response
DRP	Demand Response Program
DSL	Digital Subscriber Lines
DSM	Demand Side Management
E&SG	Energy and Strategy Energy Group
EC2	Elastic Computer Cloud
EDA	EventDriven Architecture
eDRX	extended Discontinuous Reception
EHPA	Associazione Europea Pompe di Calore
FA	Fattori Abilitanti
FER	Fonti Energetiche Rinnovabili
FF	Fonti energetiche di origine Fossile
FV	Fotovoltaico
GMI	Global Market Insights
GSE	Gestore dei Servizi Energetici
H_2	Idrogeno

HMI	Human Machine Interface
HTTP	Hypertext Transfer Protocol
IaaS	Infrastructure as a Service
ICN	Information Centric Networking
ICT	Information and Communications Technology
IEA	International Energy Agency
IED	Intelligent Electronic Devices
IETF	Internet Engineering Task Force
ILUC	Indirect Land Use Change
IoT	Internet of Things
IPCC	Intergornmental Panel on Climate Change
ISPRA	Istituto Superiore per la Protezione e la Ricerca Ambientale
LCOS	Levelized Cost Of Storage
LEC	Local Energy Community
LHTES	Latent Heat Thermal Energy storage
Li-ion	Batterie Ioni di Litio
LPWAN	Low-Power Wide Area Network
m-CHP	Micro Cogeneration Heat and Power
M2M	Machine-To-Machine
MENA	Middle East and North Africa
MISE	MinisterodelloSviluppoEconomico
MQTT	Message Queue Telemetry Transport
MSD	Mercato dei Servizi di Dispacciamento
MSD_D	Mercato dei servizi di Dispacciamento specifico per le reti di Distribuzione
MSH	Multiple Site Heating
NFER	Nuove Fonti Energetiche Rinnovabili
NHA	National HydropowerAssociation
NIST	National Institute of Standard and Technology
OFC	Optical-Fiber Communication
OVE	Olio Vegetale Esausto
PaaS	Platform as a Service
PEM	Proton Exchange Membrane
PEMFC	Proton Exchange Membrane Fuel Cell
PGD	Paradigma di Generazione Distribuita
PHEV	Plug In Electric Vehicle
PHS	Pumped Hydro Storage
PLC	Power Line Communication
PNIEC	Piano Nazionale Integrato Energia e Clima
PoliMi	Politecnico di Milano
P2G	Power to Gas
P2H	Power to Heat
P2P	Power to Power
QoS	Quality of Service
RDEE	Riconversione Diretta Energia Elettrica
RER	Renewable Energy Resource
REST	Representational State Transfer

RdM	Regioni del Meridione
SaaS	Software as a Service
SAE	Sistemi di Accumulo Energetico
SAP	Sistemi di Autoproduzione
SAT	Sistemi di Accumulo Termico
SCADA	Supervisory Control And Data Acquisition
SdA	Sistemi di Accumulo
SDC	Sistemi di Distribuzione Chiusi
SEN	Strategia Energetica Nazionale
S&MG	Smart and Micro Grid
SEESEU	Sistemi Esistenti Equivalenti ai Sistemi Efficienti di Utenza
SEU	Sistemi Efficienti di Utenza
SG	Smart Grid
SHD	Solar District Heating
SHTES	Sensible Heat Thermal Energy Storage
SLAs	Service Level Agreements
SMES	SuperconductingMagnetic Energy Storage
SOAP	Simple Object Access Protocol
SOEC	Solid OxideElectrolyser Cell
SOFC	Solid OxideFuel Cell
SSH	Single Site Heating
SSP	Sistemi in regime di Scambi sul Posto
SSPC	Sistemi Semplici di Produzione e Consumo
SV	Sampled Value
S3	Simple Storage Service
TA	Tecnologie Abilitanti
TcES	Thermochemical Energy Storage
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
URI	Uniform Resource Identifier
UVAC	Unità Virtuali Abilitate di Consumo
UVAP	Unità Virtuali Abilitate di Produzione
UVAM	Unità Virtuali Abilitate Miste
UVAN	Unità Virtuali Abilitate Nodali
VE	Veicoli Elettrici
VT	Voltage Transducer
V2G	Vehicle to Grid
WAN	Wide Area Network
WCT	Wired Communication Technology
WMN	Wireless Mesh Network
WSN	Wireless Sensor Network
XML	Extensible Markup Language

EXECUTIVE SUMMARY

Lo scopo dell'OR 5 è la progettazione della piattaforma per la gestione della Community Energy Storage. In particolare, realizzare una piattaforma basata sul paradigma del Cloud (di seguito verrà indicata come piattaforma ComESto) che renda disponibile ai gestori della Community Energy Storage per l'espletamento delle funzioni Energy Provider (EP), un sistema in grado di gestire unitariamente le risorse distribuite in modo da raggiungere un'ottimizzazione nella gestione energetica dell'aggregazione ed alla partecipazione ai "mercati per i servizi di dispacciamento specifico per le reti di distribuzione" (MSD_D) per l'erogazione di servizi di bilanciamento/dispacciamento al DSO.

La piattaforma ComESto per erogare i servizi di supporto allo svolgimento delle attività di trading dell'EP sarà implementata su modelli di gestione e controllo dei sistemi di accumulo, sia convenzionale che non, ed in generale della nanogrid su diverse scale temporali che vanno dalla fase di pianificazione (dimensionamento), alla programmazione (su base mensile, giornaliera) fino al tempo reale e che siano in grado di tenere conto sia condizioni ottimali di ogni singolo membro dell'aggregazione che del "welfare" dell'intera comunità, in termini di comfort, efficienza energetica e costi.

Per favorire la scalabilità, l'efficienza della piattaforma si dovrà indagare su quale sia la tecnologia di comunicazione migliore e più efficiente da utilizzare e, in ottica di realizzare degli scambi tra i membri della piattaforma con un adeguato livello di sicurezza e di condividere informazioni e dati simultaneamente tra i membri dell'aggregazione in maniera sicura si dovrà anche valutare quale siano i meccanismi migliori da utilizzare all'interno del progetto.

1 INTRODUZIONE E OBIETTIVO DI PROGETTO

La ricerca di Fonti Energetiche Rinnovabili è uno degli obiettivi di ricerca principali degli ultimi anni. La possibilità di poter creare un sistema energetico "green", cioè a emissioni zero, è diventato oramai un obiettivo prioritario a livello mondiale.
Insieme alla rete energetica nazionale si stanno affiancando una serie di sistemi ibridi capaci di fornire energia in maniera alternativa con particolare attenzione agli aspetti legati a consumi e emissioni.
In tutto questo contesto si sta affermando una nuova figura nel panorama energetico mondiale, la figura del prosumer ovvero di un utente finale capace non solo di assorbire energia ma anche di produrla diventando un elemento attivo del sistema complessivo.
In tale contesto risulta estremamente importante definire degli standard e dei protocolli di comunicazione tra gli attori del nuovo sistema energetico che prevede elementi capaci non solo di consumare ma anche di produrre energia e fornirla al sistema.

L'obiettivo principale delle attività di ricerca condotte nell'OR 5.5 ha riguardato lo studio e l'analisi dei principali standard e protocolli di comunicazione da poter utilizzare nel contesto delle smartgrid. La ricerca prevede la definizione e l'individuazione dei protocolli da utilizzare nella piattaforma ComESTO affinché le nanogrid possano scambiare informazioni tra di loro ed interfacciarsi con l'EP, nonché ricevere i comandi di dispacciamento da parte del DSO.
Partendo dalle tecnologie di comunicazione wireless nate a supporto dell'Internet delle cose (IoT) nel presente deliverable si sono riassunte le principali caratteristiche che contraddistinguono ogni tecnologia partendo dalla rete 4G, passando per LTE, 5G, NB-IoT, WiFi, fino ad arrivare ai nuovi standard per comunicazioni long range note con il nome di LoRA e la nuova generazione di rete cellulare: il 6G.

La ricerca condotta in tale attività si è soffermata su un'attenta analisi delle diverse tecnologie wireless abilitanti per l'IoT. Un sistema di smartgrid è composto da una serie di dispositivi/sensori che hanno la necessità di scambiarsi grosse quantità di informazioni per cui l'IoT risulta essere una tecnologia fondamentale. Ed in particolare, risulta di grande importanza lo studio dei principali protocolli di comunicazione che si possono adoperare in tale ambito perché rappresentano un punto cruciale in tutto il contesto delle smartgrid.

La ricerca in ambito smartgrid per ciò che riguarda i possibili meccanismi di comunicazione nelle smartgrid non può prescindere dallo studio dello standard IEC 61850. Tale standard è adoperato per la progettazione dei sistemi di automazione per le sottostazioni elettriche e prevede l'utilizzo di un certo numero di protocolli che girano su reti TCP/IP con switch ethernet molto performanti. Quindi la comunicazione tra i vari attori del sistema è un aspetto cruciale da tenere in considerazione e deve rappresentare uno standard unificato a livello internazionale in grado di seguire i progressi tecnologici che sono stati portati avanti nel contesto energetico.

Sono stati studiati e analizzati i protocolli machine to machine a supporto delle smartgrid capaci di garantire una corretta comunicazione tra i dispositivi/sensori presenti nel sistema. Si sono analizzati nel dettaglio i due principali protocolli di comnunicazione utilizzati nel contesto dell'M2M, il protocollo CoAP e il protocollo MQTT. Di ognuno dei due si sono approfonditi i meccanismi di inoltro, il formato del messaggio le varie caratteristiche del protocollo utile ai fini del suo impiego nel contesto dei sistemi di energia. Inoltre, si è implementato un caso d'uso con entrambi i protocolli al fine di valutare la bontà del loro impiego nei meccanismi di comunicazione tra i dispositivi/sensori nella piattaforma ComESTO. Infine, si è mostrato una valutazione delle prestazioni confrontando CoAP e MQTT e arrivando ad indicare il protocollo MQTT come candidato principale da utilizzare all'interno del dimostratore ComESTO.

2 ANALISI RETI DI TELECOMUNICAZIONI E STANDARD DI MERCATO

Nelle prossime sezioni sarà presentato un riassunto delle tecnologie nate per supportare i servizi IoT [1], con un focus particolare sul Narrowband Internet of Things (NB-IoT). Verranno analizzati i differenti scenari che saranno resi possibili nei prossimi mesi grazie all'introduzione della nuova tecnologia NB-IoT, per poi concludere presentando le soluzioni impiegate oggi in TIM nel mondo delle utilities e dello smart metering in particolare.

2.1 Le diverse tecnologie Wireless abilitanti per l'IoT

Il panorama delle tecnologie di comunicazione wireless disponibili sul mercato è estremamente variegato, esattamente come la tipologia di servizi e dei relativi requisiti per i quali tali tecnologie vengono adottate. Ogni tipologia di applicazione IoT porta con sé una serie di requisiti di comunicazione (latenza, consumi, distanza, banda, costi) che rende ad oggi praticamente impossibile individuare un'unica tecnologia in grado di soddisfare i requisiti di ogni applicazione.

In Figura 1 è evidenziata una possibile classificazione delle tecnologie radiomobili sulla base della copertura che possono offrire e le relative velocità di trasmissione che possono essere raggiunte. Nel mondo dei dispositivi short-range, che normalmente operano su bande non licenziate (in genere per l'Europa 868MHz e 2,4GHz) si sono affermate tecnologie per dominio di applicazione (es. ZigBee per il contesto home, Wireless MBus per il metering, Low Power Bluetooth per il wearable e l'healthcare, NFC per il payment, reti mesh basate su 802.15.4 per applicazioni smart city, etc.).

Figura 1 - Range e bitrate di alcune tecnologie di comunicazione per IoT

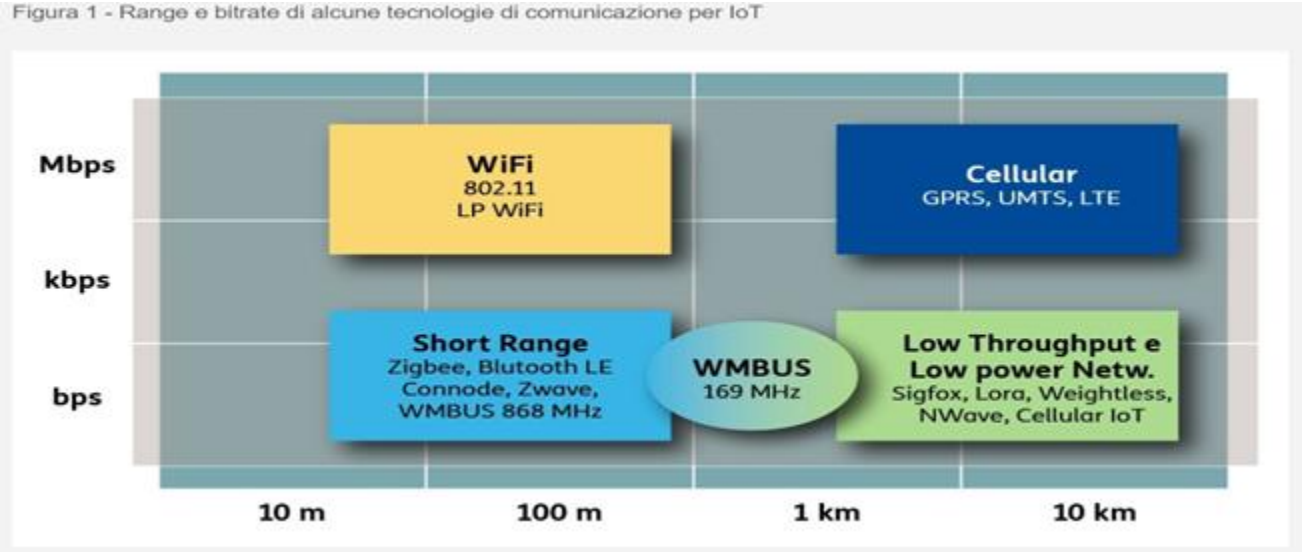


Figura 1. Classificazione delle Tecnologie Radiomobili

Per rispondere alle particolari esigenze delle numerose applicazioni che richiedono una copertura più ampia ma non necessitano di elevati bit-rate, il mercato ha visto nascere in prima battuta alcune soluzioni proprietarie di LPWAN (Low Power Wide Area Networks): Sigfox e Lora sono in tal senso gli esempi più noti. Ultimamente tuttavia, anche l'ente di standardizzazione 3GPP ha reso disponibile dei nuovi profili di accesso radio (Cellular-IoT o Machine-Type-Communication) con l'obiettivo di aumentare le coperture rispetto alle attuali reti mobili, ridurne i consumi ed avere un costo paragonabile agli attuali moduli GPRS, spesso utilizzati come soluzione alternativa in questi particolari contesti; il tutto garantendo la consolidata affidabilità e diffusione delle soluzioni standardizzate operanti su bande licenziate.

Soluzioni IoT nello standard 3GPP

Nell'ambito della Release 13 del 3GPP [2] sono state specificate tre tecnologie abilitanti per l'IoT in ambito cellulare, a seconda dei requisiti da soddisfare e dei mercati a cui si rivolgono:
NB-IoT (*NarrowBand Internet of Things*), rappresenta una soluzione basata su una nuova interfaccia radio, che può essere utilizzata sia in una porzione della banda del segnale LTE (o nella sua banda di guardia), o ancora in modo autonomo in porzioni di spettro rese disponibili dal rilascio di frequenze (ad esempio nel caso del refarming di una banda GSM).
LTE-M (*Long Term Evolution - Machine-Type Communications*), rappresenta un'evoluzione di quanto iniziato a definire nell'ambito della Release 12 del 3GPP, in termini di MTC (*Machine-Type Communications*) in una rete LTE, con l'introduzione di una categoria specifica per i terminali, denominata Cat-0. LTE-M è pertanto noto anche con l'acronimo eMTC (*enhanced-MTC*) e per i terminali è stata introdotta una nuova categoria, denominata Cat-M1.
EC-GSM-IoT (*Extended Coverage GSM IoT*), rappresenta la soluzione compatibile con una rete GSM/EDGE, di cui riutilizza una porzione della banda e che richiede la disponibilità dell'EGPRS in rete (ossia della componente a pacchetto di EDGE).

La Figura 2 mostra il percorso evolutivo di definizione delle tre tecnologie specifiche per l'IoT a partire dalle tecnologie già disponibili in ambito 2G (EGPRS) e 4G (LTE).

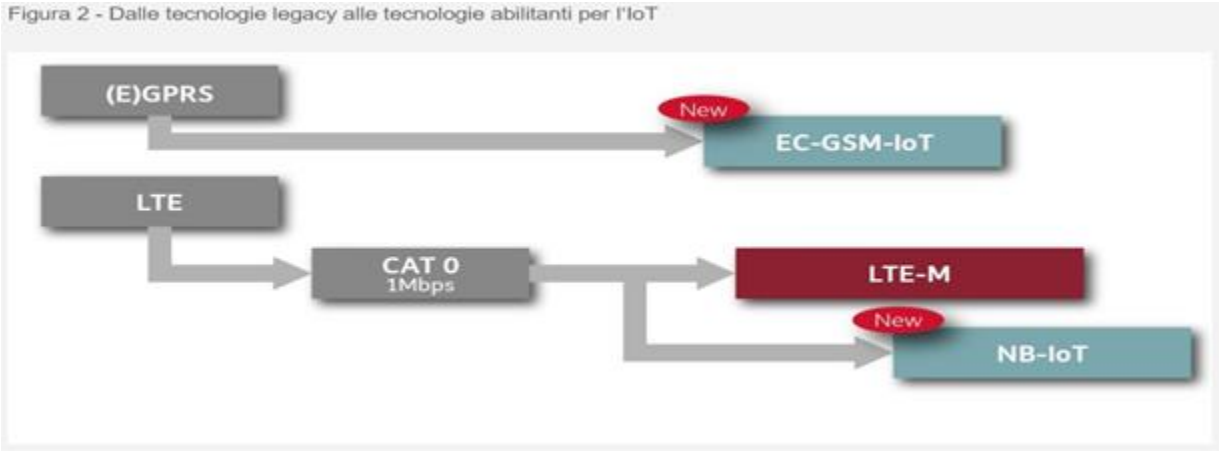


Figura 2. Percorso evolutivo di definizione delle tre tecnologie specifiche per l'IoT

NB-IoT.
NB-IoT nasce come una nuova tecnologia radio, tuttavia utilizzabile in una rete LTE nella sua banda utile di dispiegamento (il cosiddetto "*in-band deployment*", mediante l'utilizzo di una o più porzioni di spettro da 180 kHz, dette PRB, allocate nella banda utile di LTE) oppure nella sua banda di guardia ("*guard-band deployment*", mediante l'utilizzo di uno o più PRB di 180 kHz allocati nella banda di guardia di LTE) o ancora in porzioni di spettro rese comunque disponibili, fossero anche una o più portanti GSM di una rete GSM ancora dispiegata in campo ("*stand-alone deployment*", mediante l'utilizzo di uno o più canali di 200 kHz nominali, 180 kHz effettivi).

La Figura 3 riporta le 3 suddette modalità di dispiegamento di NB-IoT.

Figura 3 - Modalità di dispiegamento della tecnologia NB-IoT

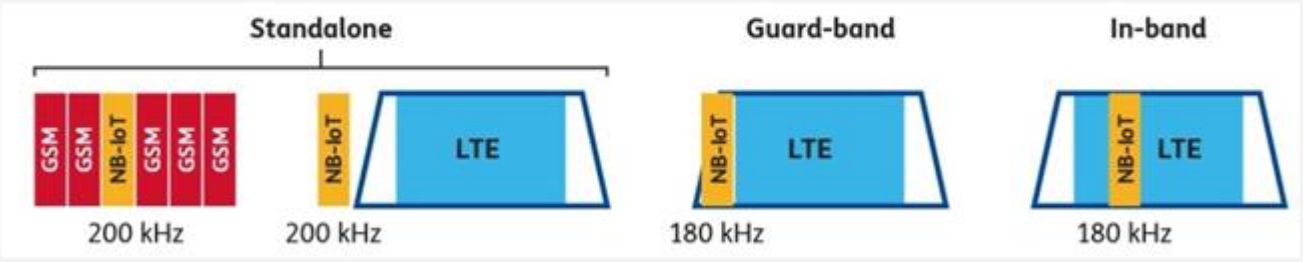


Figura 3. Tecnologie radio in NB-IoT

Il sistema è pertanto autoconsistente, con i suoi propri canali di broadcast e segnali di sincronizzazione, motivo per cui non può essere dispiegato nei 6 PRB centrali di un'allocazione LTE su cui sono trasmessi i suddetti canali e segnali per LTE, al fine di evitare interferenza reciproca tra NB-IoT ed LTE. È proprio tale caratteristica che consente inoltre a NB-IoT di essere dispiegato in modalità "*guard-band*" o "*stand-alone*", dal momento che ai fini della segnalazione broadcast e della sincronizzazione non dipende da un sistema legacy esistente.

I principali requisiti soddisfatti da NB-IoT sono così riassumibili:

- dispiegamento in una banda estremamente ridotta (180 kHz) e facilmente scalabile al crescere del traffico IoT (con allocazioni multiple di canali da 180 kHz);
- consistente estensione della copertura radioelettrica rispetto a quella fornita da una rete legacy GPRS, ossia di 20 dB superiore, corrispondente ad un MCL [nota 1] di 164 dB, per coprire gli scenari in cui i dispositivi sono collocati in luoghi non agevolmente accessibili, ad esempio negli scantinati, e/o sono protetti in contenitori metallici;
- potenza di trasmissione del terminale impostata a 23 dBm oppure a 20 dBm, valori tali da consentire l'integrazione dell'amplificatore di potenza nel SoC (*System-on-Chip*); a titolo comparativo, un terminale GPRS trasmette a 33 dBm, ossia ad un valore di almeno 10 dB superiore, pur raggiungendo una copertura radioelettrica di 20 dB inferiore rispetto a NB-IoT;
- durata della batteria del terminale superiore ai 10 anni, nel caso di un modello di traffico che contempli l'invio, da parte del terminale stesso, di una quantità di dati sino a 200 byte al giorno;
- data rate ridotto, dell'ordine di alcune decine di kbps sia in UL sia in DL, con valori di picco pari a 250 kbps in UL ed a 170/226.7 kbps in DL in *in-band/stand-alone deployment* (e valori mediati nominali di 62.5 kbps in UL e di 21.25 kbps in DL);
- assenza di requisiti stringenti in termini di latenza, con ritardi comunque non superiori ai 10 secondi nel caso di applicazioni che richiedano l'invio di allarmi da parte di dispositivi collocati anche in luoghi tali da richiedere la massima estensione di copertura radioelettrica di 20 dB; il ritardo è valutato tra l'istante in cui si verifica l'evento che determina la segnalazione d'allarme e l'istante in cui tale segnalazione è disponibile alla stazione base per essere inviata alla core network;
- complessità estremamente ridotta e presumibilmente costo estremamente contenuto dei terminali (comunque inferiore a quello dei dispositivi legacy GPRS-only di Release 97);
- supporto di un elevato numero di terminali (maggiore di 50.000) in ogni singolo settore di una cella tri-settoriale, con l'allocazione di un PRB per settore.

I campi di applicazione di NB-IoT includono quelli per i quali i dispositivi sono collocati in luoghi tali per cui è necessario garantire una consistente estensione della copertura radioelettrica e la durata della batteria è un fattore estremamente importante, in quanto non risulta agevole, e neanche economicamente conveniente, intervenire sugli stessi dispositivi per sostituirne la batteria; in questi casi il ciclo di vita dei dispositivi corrisponde di fatto alla durata stessa della loro batteria.
Al contempo, la mole di dati da trasferire e da ricevere da parte di tali dispositivi è molto contenuta (nell'ordine di alcune decine di byte al giorno, come media), per cui il NB-IoT risulta una soluzione ottimizzata per applicazioni quali lo smartmetering.
L'estensione della copertura radioelettrica, sino ad ottenere un MCL di 164 dB, è ottenuta tramite funzionalità radio quali:
una concentrazione della potenza trasmessa su una banda estremamente ridotta, ossia un incremento della PSD (*Power Spectral Density*): la canalizzazione di 180 kHz utilizzata sia in DL sia in UL corrisponde all'occupazione spettrale di un PRB in LTE, con 12 sotto-portanti da 15 kHz ciascuna; in UL è inoltre prevista la possibilità di utilizzare anche una singola sottoportante con la scelta tra 2 canalizzazioni (3,75 kHz e 15 kHz, in grado di garantire un'estensione della copertura radioelettrica rispettivamente di 17 dB ed 11 dB rispetto a quella raggiungibile con la canalizzazione convenzionale di 180 kHz), oltre all'impiego di 3 o 6 sotto-portanti da 15 kHz ciascuna;
un elevato numero di ripetizioni dei TB (*Transport Block*): si considerano sino a 2048 ripetizioni in DL e sino a 128 ripetizioni in UL, valori selezionati in modo da poter raggiungere, insieme alle altre funzionalità, i 20 dB di guadagno attesi sulla massima estensione della copertura radioelettrica rispetto al sistema GPRS;
l'impiego di schemi di modulazione efficienti che permettono di ridurre il PAPR (*Peak-to-Average Power Ratio*), consentendo di utilizzare l'amplificatore in un punto di lavoro prossimo alla saturazione, senza introdurre distorsioni, riducendo pertanto il back-off che è necessario introdurre per garantirne il funzionamento in linearità, con conseguente incremento della potenza effettivamente trasmessa, che diventa prossima alla potenza nominale dell'amplificatore.

L'estensione della durata della batteria, sino a 10 anni, è ottenuta tramite funzionalità radio quali:

- eDRX *(extended Discontinuous Reception)*, che consente di ridurre la frequenza con cui sono monitorati i canali di controllo in DL da parte del terminale ed il numero di report di misura inviati dallo stesso terminale alla rete;
- PSM *(Power Saving Mode)*, che consente di minimizzare il consumo di potenza da parte del terminale che si trova in tale modalità, in cui il terminale risulta comunque irraggiungibile (non sono infatti neanche monitorati i canali di controllo da parte del terminale), pur essendo ancora registrato alla rete; il terminale esce da tale modalità con procedure periodiche originate dallo stesso terminale, quali il TAU *(Tracking Area Update)*.

EC-GSM-IoT

EC-GSM-IoT si basa su funzionalità aggiuntive a partire da EGPRS che, insieme al PSM, consentono ad una rete GSM/EDGE di essere predisposta per fornire servizi IoT. Lo standard è stato pensato in particolare per quei Paesi, come quelli in via di sviluppo, dove una rete LTE non è ancora disponibile.

L'occupazione spettrale di ogni canale corrisponde a quello legacy GSM, ossia 200 kHz. Tuttavia, al fine di dispiegare EC-GSM-IoT, si richiede una banda utile di 2.4 MHz per permettere il *frequency hopping*, che, con l'aggiunta di 2 canali di guardia di 200 kHz ciascuno agli estremi della banda, porta l'occupazione di banda complessiva a 2.8 MHz. Nel caso in cui EC-GSM-IoT sia l'unico servizio dispiegato insieme all'EGPRS con cui condivide le risorse radio, ma in assenza del servizio voce GSM, sono sufficienti 600 kHz, riconducibili ad 1 MHz di banda necessaria con i suddetti canali di guardia.

La potenza di trasmissione del terminale è pari a 33 dBm (ossia quella di un terminale GSM convenzionale), al fine di raggiungere un'estensione della copertura radioelettrica corrispondente ad un MCL di 164 dB. Tale livello di potenza richiesto all'amplificatore non ne consente l'integrazione nel SoC, oltre ad avere implicazioni in termini di consumo e di conseguente durata della batteria. Nel caso in cui la potenza di trasmissione del terminale venga ridotta a 23 dBm (ossia al

valore comunque massimo previsto sia per LTE-M sia per NB-IoT), l'estensione della copertura radioelettrica si riduce conseguentemente di 10 dB, limitandosi ad un MCL di 154 dB. Il data rate di picco raggiungibile sia in DL sia in UL è di 491 kbps, mentre il valore mediato nominale è di 98 kbps sia in DL sia in UL. Al fine di soddisfare i requisiti di capacità (più di 50.000 terminali in ogni singolo settore di una cella tri-settoriale), si rende necessario utilizzare una tecnica overlay basata su CDMA, sia sui canali di traffico sia sui canali di segnalazione.

In Tabella 1 è riportata un'analisi comparativa delle tre tecnologie considerate, che ne riassume le principali caratteristiche radio.

Tabella 1 – Comparativa tra tecnologie radio

Tabella 1 - Principali caratteristiche radio di LTE-M, NB-IoT, EC-GSM-IoT

	LTE-M	NB-IoT	EC-GSM-IoT
Dispiegamento	In-band LTE	In-band LTE, Guard-band LTE, Stand-alone	In-band GSM
Copertura radioelettrica	155.7 dB (20 dBm)	164 dB (23 dBm)	154 dB (23 dBm) 164 dB (33 dBm)
Canalizzazione	1.08 MHz	180 KHz	200 kHz
Banda minima richiesta	1.4 MHz	180 kHz (In-band LTE, Guard-band LTE) 200 kHz (Stand-alone)	2.4 MHz (con servizi voce GSM) 600kHz (senza servizi voce GSM)
Duplexing	HD-FDD, FD-FDD, TDD	HD-FDD	HD-FDD
Data rate di picco	HD-FDD e FD-FDD 1 Mbps DL HD-FDD e FD-FDD 1 Mbps UL	In-band LTE 170 kbps DL Stand-alone 226.7 kbps DL 250 kbps UL	491 kbps DL 491 kbps UL
Data rate mediati nominali	FD-FDD 800 kbps DL FD-FDD 1 Mbps UL HD-FDD 300 kbps DL HD-FDD 375 kbps UL	21.25 kbps DL 62.5 kbps UL	98 kbps DL 98 kbps UL
Classi di potenza dei terminali	20 dBm / 23 dBm	20 dBm / 23 dBm	23 dBm / 33 dBm

2.2 Rete 4G-LTE

La rete ***LTE (Long Term Evolution)*** utilizza banda di frequenze a 1800 MHz, 800 MHz e 2600 MHz.
Il processo di standardizzazione del sistema LTE nel 3GPP (Third Generation Partnership Project) ha portato alla definizione dei requisiti che sono riportati di seguito:

- Peak data rate iniziali fino a 100 Mbps in DL e di 50 Mbps in UL. Il peak data rate di fatto dipende dalla banda del sistema, che in LTE è scalabile, e dalla categoria dei terminali LTE utilizzati. In futuro con la tecnica MIMO 4x4 e nuove categorie di terminali saranno ottenibili velocità di 300 Mbps il DL e 75 Mbps in UL.
- Miglioramento della latenza terminale-rete: sotto i 10 ms per lo User Plane
- Banda del sistema scalabile: LTE può operare con larghezze di banda particolarmente flessibili, da circa 1.4 a 20 MHz, con possibilità ad esempio di bande intermedie quali 5 e 15 MHz.

LTE offre prestazioni e livelli di servizio che migliorano l'utilizzo di contenuti multimediali e l'accesso al Cloud. Di seguito alcuni esempi di utilizzo con una sintesi dei vantaggi che offre la tecnologia LTE

- Navigazione Internet in mobilità fino a 100 Mbps in download e 50 Mbps in upload.
- Video streaming: I video non hanno interruzioni né tempi di attesa. È possibile guardare video di alta qualità/definizione.
- Download: Il download di file e video anche di grandi dimensioni richiede tempi di attesa minimi. LTE offre infatti velocità di download circa 5 volte superiori a quella offerta dalla tecnologia 14.4 Mbps, e più del doppio rispetto alla tecnologia HSPA 42 Mbps (commercializzata come Ultra Internet 42 Mega)
- Upload: Si possono caricare file, foto e video anche di grandi dimensioni (ad esempio su Cloud, forum, social network, etc.…) in pochi istanti. LTE offre infatti velocità di upload di circa 10 volte superiori rispetto agli standard precedenti.
-

2.3 LTE Cat NB1 (NB-IoT)

Nell'ambito delle reti ***LPWAN (Low Power Wide Area Network),*** l'**LTE** con le sue declinazioni tecnologiche *LTE Advanced e Pro Advanced, LTE Cat M1 (LTE M) e* ***LTE Cat NB1*** *(**NB IoT**)* si configurano essere le tecnologie di accesso più adatte per supportare le esigenze dell'***IoT***.
Il **N**arrow **B**and **IoT** (NB IoT) è una **tecnologia radio a banda stretta, standardizzata dal 3GPP** *(Third Generation Partnership Program), l'ente di standardizzazione mondiale per le telecomunicazioni mobili,* che opera su banda licenziata.
L'NB IoT indirizza nello specifico le applicazioni cosiddette «**Massive Machine Type Communication**» (***Massive IoT***) caratterizzate da ampia diffusione di oggetti da connettere, basso/medio throughput (da pochi KB a qualche MB), dalla necessità di disporre di device a basso costo e dalla elevata efficienza energetica che si traduce in un aumento della durata delle batterie che alimentano gli oggetti connessi. Applicazioni che necessitano anche di coperture radio decisamente più performanti rispetto a quelle oggi offerte dalle reti GSM/LTE sia in ambienti Outdoor che Indoor.

2.3.1 BENEFICI DELLA TECNOLOGIA NB IoT

Passiamo a elencare i principali benefici che, l'utilizzo della tecnologia NBIoT, consente di ottenere:

- Consistente Miglioramento della CoperturaRadioelettrica sia Outdoor che Indoor grazie alla capacità di fornire un MCL[(1)] massimo di 164Db, che si traduce in un guadagno di segnale che può arrivare fino a +20Db se confrontato con quello tipicamente garantito dalla rete legacy GPRS.
- Miglioramento dell'Efficienza Energetica che consente di allungare il ciclo di vita delle batterie dei dispositivi connessi fino a 10 anni di durata grazie ad un utilizzo intelligente del device e delle risorse di Rete con cui il device interagisce.
- Aumento della capacità servente dell'Operatore all'interno di una stessa cella, che si traduce nella capacità di garantire connettività dati ad un elevatissimo numero di oggetti in una stessa area: l'NB IoT ha la capacità di dare servizio fino a 100K dispositivi all'interno di una stessa cella.

(1)L' *MCL (o Maximum Coupling Loss) rappresenta la massima attenuazione che un sistema radiomobile è in grado di sopportare lungo il canale trasmissivo.*

Vantaggi della copertura di +20Db

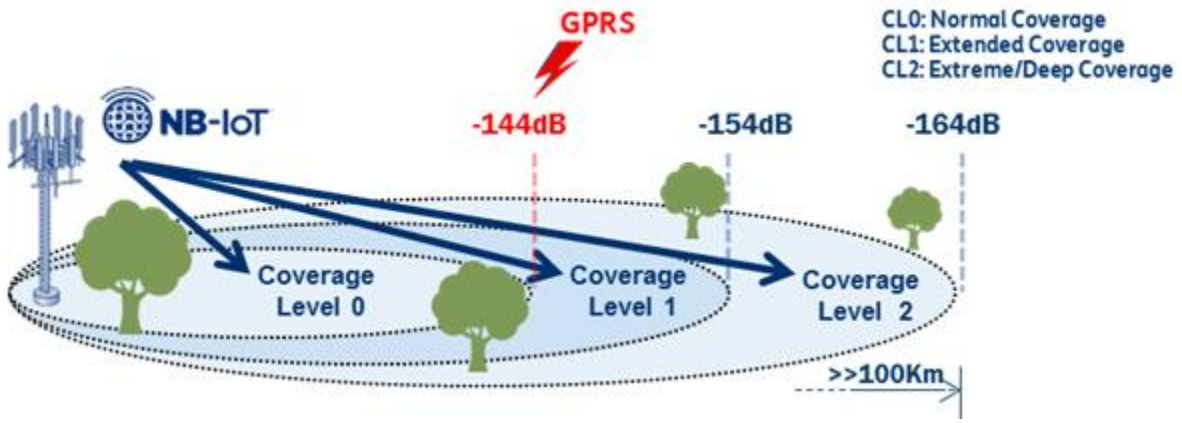


Figura 4. Esempio di copertura per NB-IoT

- In *Outdoor* significa che sul territorio la copertura della rete NB IoT è 7 volte superiore a quella del GSM (2G/3G)
- In Indoor significa che la copertura della rete NB IoT negli spazi interni è in grado di raggiungere:
    - Locali seminterrati
    - Locali interni
    - Zone protette da grate metalliche
    - Aree interrate (Tombini, Tubature, Canalizzazioni, Pozzetti, …..)

Più si è vicini all'antenna radio e migliore è il livello del segnale radio e la QoS percepita in termini di maggiore capacità/Data Rate e minori sono le ripetizioni di messaggi tra il dispositivo e la rete.

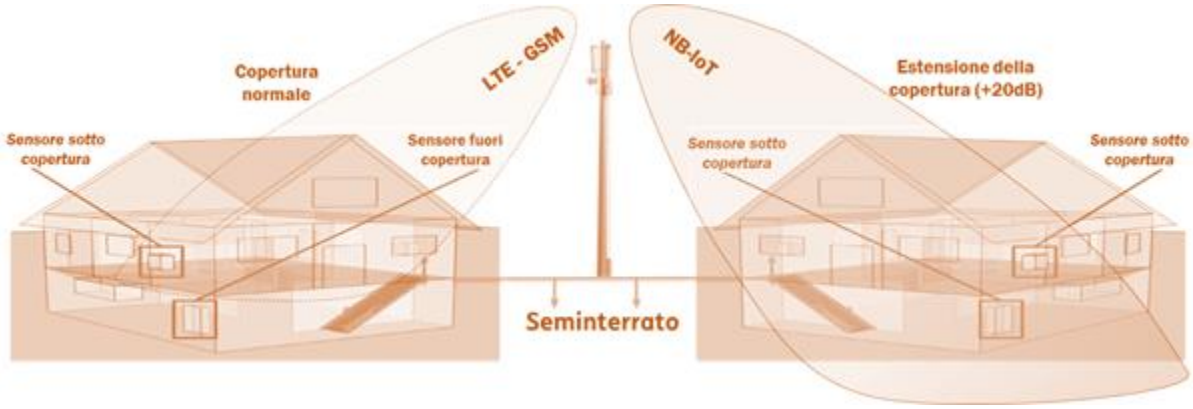


Figura 5. Esempio di connessione

2.3.2 Efficienza energetica dei dispositivi

L'NB IoT introduce il concetto di «massimizzazione dell'efficienza energetica» e questo grazie a:

- Disponibilità di dispositivi che utilizzano potenze di trasmissione di 10dB più basse rispetto a quelle dei dispositivi GPRS e che si attestano tra i 23 dBm e i 20 dBm. Minore potenza di trasmissione significa non solo minore consumo energetico ma anche minor costo potendo contare sulla semplificazione dell'Hw ottenibile con l'integrazione dell'amplificatore di segnale direttamente nel System on Chip e con l'adozione di batterie di minor capacità. Uso di tecniche di rete note come PSM ed eDRX che permettono l'ottimizzazione dell'interlavoro tra il chipset radio NB IoT del dispositivo e la Rete Radiomobile.
- Il Power Saving Mode (PSM) è una tecnica di gestione che consente di ottimizzare il consumo di potenza del dispositivo portandolo in uno stato di irraggiungibilità e al tempo stesso di stand by. Una volta entrato in PSM il dispositivo non risulta più raggiungibile. La novità introdotta dalla tecnologia NB IoT sta nel mantenere sempre viva la sua registrazione sulla rete radiomobile così da minimizzare al massimo le successive procedure di re-attache alla rete stessa che in questo caso non sono più necessarie. Il terminale nello stato PSM si sveglia saltuariamente per permettere solamente le consuete procedure Tracking Area Update.

L'**extended Discontinuous Reception** (**eDRX**) è una tecnica che consente di ridurre la frequenza di monitoraggio dei canali di controllo in Down Link (DL) da parte del terminale nonché il numero di report di misura che di norma vengono inviati dal terminale stesso alla Rete (c.d. procedure di Paging Occasions).

2.3.3 Le funzionalità offerte dall'NB-IoT:

- Copertura nazionale equivalente a quella dell'LTE con aumento delle performance in termini di rapporto Segale/Rumore di +20dB
- Potenza di trasmissione del terminale di 10 dB inferiori a quelli tipici della tecnologia GPRS (20dB vs 33dB) adatta anche ai wearables object.
- Durata delle batterie dei terminali superiore ai 10 anni.
- Data rate ridotto, tecnologia particolarmente adatta per applicazioni a basso throughput
- Bassa latenza: ritardo introdotto non è superiori ai 10 sec.
- complessità e costi ridotti dei terminali (costi < a 10€/modulo)
- Supporto ad un elevato numero di dispositivi (circa 100K oggetti gestiti) in ogni singolo settore di una cella tri-settoriale
- Elevata Sicurezza garantita dalla rete LTE

L'NB IoT di fatto anticipa, per particolari applicazioni, il 5G con il quale andrà ad integrarsi.

2.4 Tecnologia 5G

Il 5G è la nuova generazione dei sistemi mobili. Il suo campo di applicazione è molto più ampio rispetto al passato, rappresentando di fatto l'abilitatore tecnologico della società connessa e della Digital Life. I paradigmi di innovazione a cui si ispira sono basati su tecnologie e modelli di business in continua evoluzione, che richiedono agli Operatori Telco un percorso di profonda trasformazione. Analizzeremo brevemente le tecnologie abilitanti [1] il 5G oltre ad ipotizzare un percorso di dispiegamento che metta a valore i benefici congiunti delle tecnologie 5G e LTE ricordando comunque che allo stato la tecnologia 5G non è disponibile sulla zona d'interesse progettuale Rende-Cosenza.

Il 5G non è semplicemente un nuovo sistema costituito da una nuova interfaccia radio e una nuova rete core, come avvenuto in passato per le precedenti generazioni di sistemi radiomobili, ma rappresenta il punto di aggregazione di un insieme di stream di evoluzione tecnologica a driver di mercato, e si prefigge di diventare la piattaforma Telco su cui sviluppare l'ecosistema dei nuovi servizi digitali. Pertanto, diviene cruciale per gli Operatori Telco essere in grado di comporre, in fase di pianificazione e dispiegamento, un quadro di insieme che includa le migliori soluzioni tecnologiche, in modalità organiche agli investimenti, flessibile alla realizzazione di nuovi servizi e che minimizzi la frammentazione delle piattaforme. E', inoltre, sempre più importante disporre di tecnologie che consentano Time To Market a costi ridotti e, in modo corrispondente, decommissioning altrettanto semplificati con la possibilità di utilizzare con successo le stesse piattaforme per nuovi servizi. In questa prospettiva, un primo elemento essenziale è rappresentato dall'inclusione nel framework 5G dalle future evoluzioni di LTE-A (specifiche dello standard 3GPP dalla Release 15 in poi).

Un elemento rilevante è rappresentato dalla realizzazione di un'unica core network (NextGeneration Core - NGC), alla quale attestare, attraverso la stessa interfaccia logica, tutti gli accessi 5G (quindi evoluzione di LTE-A e NR, ed in futuro altri accessi, quali accessi fissi broadband), allo scopo non solo di garantire la necessaria continuità di servizio, ma soprattutto per focalizzare su una sola piattaforma tecnologica evoluta lo sviluppo dei nuovi servizi, senza dover ricorrere a dispiegamenti dedicati per tecnologia o servizio.

Tuttavia, affinché tali architetture risultino adeguatamente flessibili ed efficienti, è necessario che esse siano definite secondo un paradigma di virtualizzazione end to end (inclusi i sistemi di accesso in architettura Virtual RAN), che consenta il dispiegamento di funzioni di rete virtuali multi-vendor e di catene di servizio e2e indipendenti su una piattaforma HW comune, che possa essere condivisa anche con le funzioni IT, realizzando dunque un nuovo scenario di convergenza, non più solo fisso-mobile, ma anche tra le entità funzionali di rete e quelle dell'Information Technology.

Gli annunci di trial e lanci pre-commerciali 5G si susseguono, coinvolgendo player di tutti i mercati avanzati: negli Stati Uniti Verizon ha fondato il Verizon 5G Technical Forum (V5GTF) ed ha annunciato il dispiegamento di un pilot on field in 11 città orientato a servizi FWA a 28 GHz. Nell'area Asia-Pacifico i primi pilot e lanci pre-commerciali sono legati agli eventi vetrina delle Olimpiadi invernali in Corea del Sud e dei Giochi Olimpici in Giappone, con focus su servizi MBB. In Europa, la Commissione Europea ha lanciato il 5G Action Plan per il lancio coordinato del 5G negli Stati Membri a partire dal 2020, con un piano di trial pan-europei a partire dal 2018, finalizzati all'evento vetrina dei Campionati Europei UEFA. In tale scenario, un piano di dispiegamento del 5G coerente sia con le attività della Commissione Europea ed i requisiti dell'Agenda Digitale Europea, sia con la roadmap degli standard 3GPP e la disponibilità tecnologica, si può articolare in due Fasi (rappresentate in Figura 6).

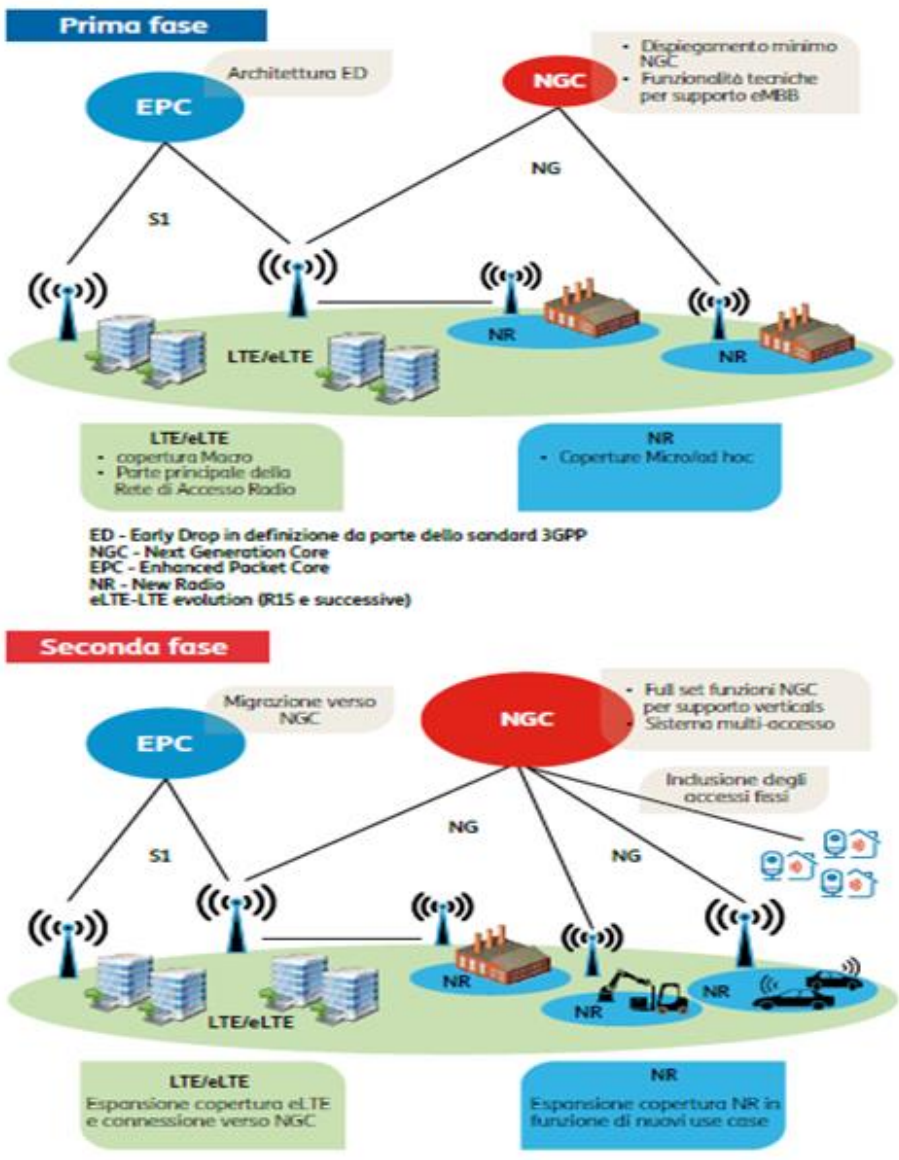


Figura 6. Fasi dispiegamento 5G

Fasi del dispiegamento 5G

A valle di una attività preliminare, dedicata a trial e PoC sia in laboratorio sia on field, si prevede una prima fase di dispiegamento commerciale in cui il 5G sarà lanciato in aree specifiche in funzione di determinati requisiti di servizio, su bande intermedie (ad esempio 3.5 GHz) o, se compatibile con i requisiti Regolatori, anche millimetriche, dove LTE-A rimane la tecnologia per i dispiegamenti macro, a cui si appoggiano coperture ad hoc NR, principalmente in dual connectivity secondo lo standard early drop del 3GPP. Una seconda Fase di dispiegamento commerciale (approssimativamente dal '21 al '25) vedrà un'espansione della copertura sia sulla Nuova Radio sia sulle evoluzioni di LTE-A, oltre all'utilizzo più esteso di bande millimetriche (ad es. 24,25 – 27,5 GHz, 27.5-29.5 GHz e 31.8-33.4 GHz), per una piena espansione dei servizi verticali

del mondo business in compartecipazione con molteplici player industriali. Contestualmente l'evoluzione della tecnologia LTE già oggi consente agli Operatori Telco di aprirsi ai nuovi servizi e mercati adiacenti, sia grazie alle maggiori performance dell'LTE-A, sia grazie a nuove capability tecnologiche.

2.4.1 Architettura della rete e dei servizi in tecnologia 5G

L'architettura end to end del 5G si pone come obiettivo quello di fornire un'estrema flessibilità in termini di supporto e configurazione di funzionalità e servizi ed integrazione di accessi. Essa è composta da RAN, aggregatori, reti IP, nanocore ecc.
La Figura 7 illustra una generica architettura di rete 5G. Come evidenziato, la rete 5G utilizza un concetto di IP piatto in modo che diversi RAN (reti di accesso radio) possano utilizzare la stessa nanocore per la comunicazione. I RAN supportati dall'architettura 5G sono GSM, GPRS / EDGE, UMTS, LTE, LTE-advanced, WiMAX, WiFi, CDMA2000, EV-DO, CDMA One, IS-95, ecc.

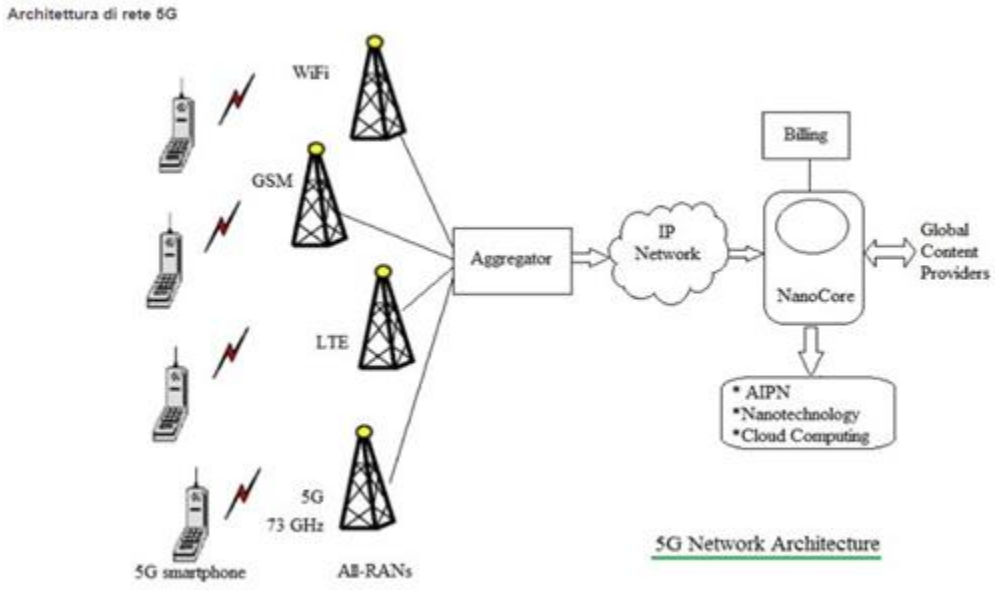


Figura 7. Architettura di rete 5G

L'aggregatore 5G aggrega tutti i traffici RAN e li indirizza al gateway. Il terminale mobile 5G ospita diverse interfacce radio per ciascun RAT al fine di fornire supporto per tutte le tecnologie di accesso allo spettro e wireless. Un altro componente nell'architettura di rete 5G è 5G nanocore. Si compone di nanotecnologie, cloud computing, architettura All-IP. Il cloud computing utilizza internet e server remoti centrali per mantenere i dati e le applicazioni degli utenti, permette ai consumatori di utilizzare le applicazioni senza alcuna installazione e accedere ai loro file da qualsiasi computer in tutto il mondo con l'uso di internet.

Un ulteriore componente è fornito dalle funzionalità di orchestrazione, per la gestione automatizzata delle capability di rete, delle applicazioni e dei servizi, ma anche per le attività di provisioning, administration e maintenance, per la riduzione della complessità di rete, dei costi di operation e del time to market dei servizi. Per supportare questi cambiamenti architetturali le reti di trasporto ottica e IP devono evolvere di pari passo, integrandosi all'interno di un framework unico di gestione e configurazione. Infine, è essenziale associare le capability della rete in fibra (banda, consumo di energia, gestione) con i requisiti sempre più stringenti del nuovo accesso radio (throughput, latenza, affidabilità).

Questo comporta sia il ricorso ad architetture virtualizzate in rete di accesso (V-RAN), sia alla sempre maggiore remotizzazione al bordo della rete dei contenuti per applicazioni real-time, secondo l'approccio MEC. Il design architetturale non dovrà limitarsi ai nuovi sistemi 5G, ma dovrà abbracciare l'evoluzione dei sistemi legacy, in modo da consentire un processo di sviluppo continuo dei servizi, evitando i fenomeni di decommissioning e frammentazione tipici delle architetture tradizionali.

2.4.2 Applicazione della tecnologia 5G nei sistemi IoT

La Figura 8 mostra il concetto delle 5G Network Slices ("fette" virtuali di rete 5G) elaborato dalla Next Generation Mobile Networks (NGMN) Alliance per consentire la gestione dei diversi mercati. Le risorse di rete generiche sono suddivise in: nodi di storage e cloud computing, posti sia nel centro (core) che nei bordi (edge) della rete, nodi di commutazione (routers), nodi di accesso e collegamenti trasmissivi. I nodi di accesso sono collegati con le stazioni radio base che impiegano differenti interfacce RAT (Radio Access Technology) a seconda del mercato indirizzato. Antenne radio, fronthauling e C-RAN sono usate per la virtualizzazione delle base station, mentre il backhauling collega la RAN al nucleo della rete. Fanno parte della fetta di rete anche le risorse poste nei dispositivi terminali (sensori e apparati di utente). Tutte le varie risorse possono essere dedicate alla singola "fetta di rete", oppure condivise tra fette di rete.

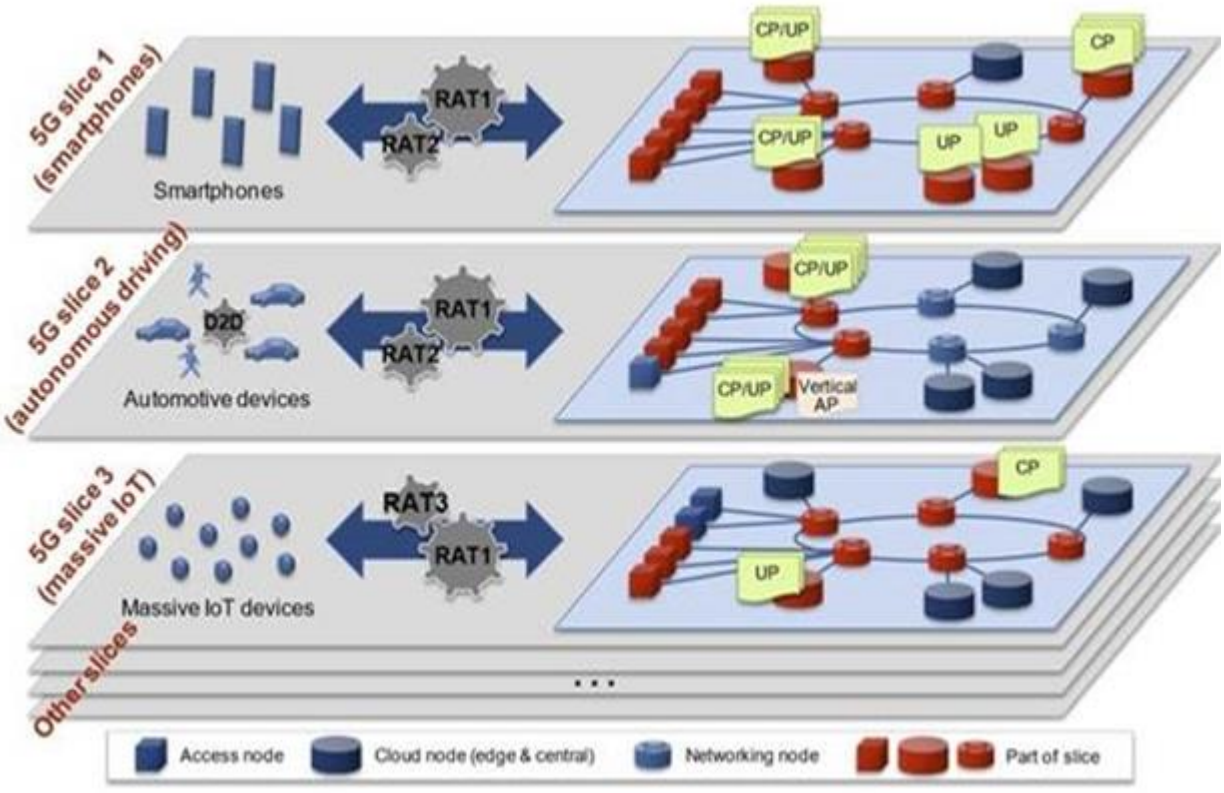


Figura 8. 5G Network Slicing

In Figura 8 sono mostrate a titolo di esempio **tre fette di rete**. La **prima** è dedicata ai servizi mobile broadband (eMBB): sono evidenziate in rosso le risorse utilizzate e i nodi di servizio sono marcati con le sigle CP e UP (Control Plane e User Plane) a seconda delle funzioni svolte. Un esempio di funzioni CP sono quelle per la gestione della mobilità presenti nell'elemento di rete detto MME (Mobility Management Entity).
La **seconda fetta** è dedicata al comparto automobilistico con applicazioni di connected car e autonomous driving (URLLC): i terminali posti nei veicoli permettono la comunicazione D2D (device to device) oltre che la comunicazione con le infrastrutture. In questa fetta si nota anche l'uso di un dispositivo di edge computing (indicato come "vertical AP", Application Plane) per migliorare la latenza delle comunicazioni V2I, "vehicle to infrastructure".
La **terza fetta** è infine dedicata ad applicazioni **di massive IoT** quali quelle delle smart homes/smart cities (mMTC). Il concetto di "Network Slicing" ha varie applicazioni nell'ambito dei sistemi 5G: una delle più importanti è quella della convergenza tra utenti fissi e utenti in mobilità, ambedue connessi via radio al 5G. Il sistema dovrà ottimizzare le risorse dell'accesso, del backhauling e del "core" al fine di riservare apposite risorse di rete (spettro radio, banda di connessione, autenticazione, ecc.) per servire gli utenti fissi collegati con piccole celle.

Si osserva poi che alcune fette virtuali di rete possono essere allocate a reti e servizi orientati alla pubblica amministrazione e ai cittadini, compresi i servizi di emergenza. Si osserva infine che nel recente rapporto BEREC sulla Net Neutrality si fa riferimento al 5G network slicing come un possibile strumento per veicolare su Internet "servizi specializzati".

Si possono identificare 3 classi di servizio:

- Alto throughput, per servizi video e di realtà aumentata (eMBB: enhanced Multimedia BroadBand)
- Bassa energia, per servizi massive IoT per sensori con batterie a lunga vita (10 anni) (mMTC: massive Machine Type Communication)
- Bassa latenza e alta affidabilità per servizi IoT mission critical (uMTC: ultra Machine Type Communication).

Questo approccio consente di affermare che le infrastrutture 5G promettono una maggiore efficienza ed efficacia in termini di consumi di energia, tempi di creazione del servizio e flessibilità nell'uso dell'hardware e diventano di conseguenza una applicazione naturale per lo **sviluppo dell'*IoT*** in entrambi i due grandi cluster:

- ***Massive IoT***: le applicazioni sono caratterizzate da basso costo, basso consumo, e bassa capacità di comunicazione, nonché da un grande numero di dispositivi connessi; trasporti e logistica, ambiente, casa intelligente, città intelligente, agricoltura, ecc.
- ***Mission Critical IoT***: le applicazioni sono caratterizzate da alta affidabilità, bassa latenza e alta capacità; automotive, energia (smart grid), sanità, sicurezza, realtà aumentata, automazione della fabbrica, ecc.

2.4.3 Applicazione della tecnologia 5G nei sistemi IoT: Use Case

Dagli esempi di seguito riportati, che costituiscono lo stato dell'arte degli studi sulle applicazioni supportabili della tecnologia 5G, emerge la potenzialità di utilizzo nell'ambito delle attività di sviluppo previste negli OR. Compatibilmente con la disponibilità della copertura di rete presso le sedi oggetto dei dimostratori e le esigenze dei servizi e applicazioni allo studio, il 5G potrà costituire la struttura di base atta ad alimentare la piattaforma cognitiva oggetto e obiettivo del progetto COMESTO.

UseCase – Smart Surveillance

Piattaforma di Videosorveglianza e di Video monitoring attraverso l'integrazione della tecnologia Cisco/Meraki con la rete TIM per una fruizione di servizio in sicurezza Anywhere, Any time.
Offre un monitoraggio sicuro con gestione semplice e un'alta scalabilità grazie alla disponibilità del servizio cloud. Dispone inoltre di sofisticati algoritmi di intelligenza artificiale per applicazioni di advanced analytics con rispetto della privacy per abilitare scenari evolutivi di computer vision (video detection e video analisi).

Diversi gli scenari di applicazione:

- Sorveglianza e monitoraggio indoor per aziende, negozi, spazi privati, ecc.
- Sorveglianza e monitoraggio outdoor per spazi ed eventi pubblici, gestione rifiuti, traffico, persone o infrastrutture.

Queste le possibilità fornite dalla soluzione:

- Local video streaming,
- Secure video streaming,
- Heat Map,
- Person Detection end People counting,
- Motion Alert in aree configurabili,
- Night vision con configurazione ad infrarossi.

La soluzione è attualmente disponibile su connettività wired e wireless in ambiente indoor e outdoor con prestazioni di bassissima latenza e elevatissima banda.

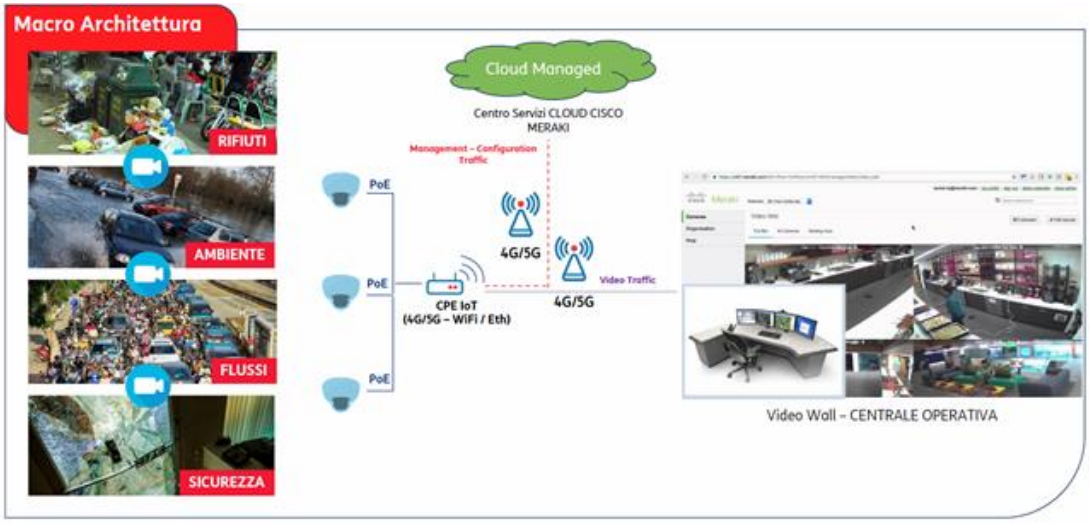


Figura 9. Smart Surveillance

Figura 10. Dettaglio Smart Surveillance

UseCase – IoT Universal Catalyst

Soluzione IoT industry 4.0 per la creazione dei **Digital Twin** in ambito IoT. Consente di creare un modello digitale di qualsiasi macchina industriale, robot, sensoristica di campo, indoor e out door consentendo la facile esportazione e gestione di tutti i parametri e dati generati, accoppiandoli e lavorandoli per offrirli ad applicativi di livello superiore.
Attraverso la pervasività della rete 5G e delle sue performance si offre la possibilità di una maggiore visibilità sulle macchine e su tutte le sensoristiche per avere un miglioramento delle performance.

Utilizzo e applicazioni verticali realizzabili:

- BUILDING MANAGEMENT SISTEM (BMS): Sistema di controllo e gestione per edifici o infrastrutture, monitoraggio e controllo di impianti meccanici o elettrici (Riscaldamento, ventilazione, illuminazione, sicurezza antincendio o antifurto).
- MONITORAGGIO AMBIENTALE: sistema di controllo per il monitoraggio di parametri ambientali (Acqua, Atmosfera, Rumore, Vibrazioni, Elettromagnetismo) per la verifica di norme vigenti o per sistema di allarmistica generale.
- GESTIONE ENERGETICA: sistema di controllo per l'analisi, il monitoraggio e l'ottimizzazione della risorsa energia per impianti privati, pubblici ed industriali.

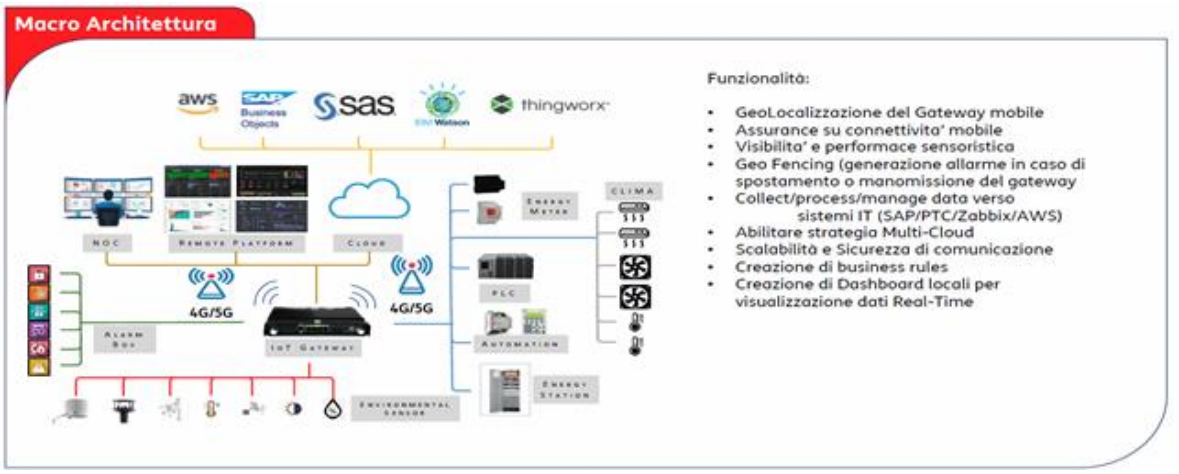
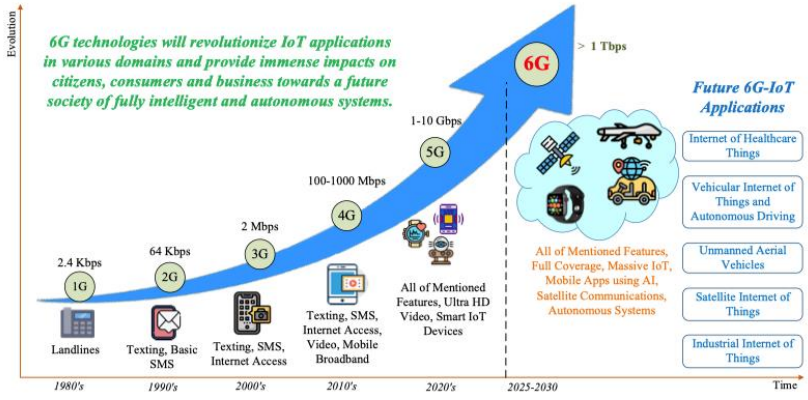


Figura 11. Macro Architettura

2.5 Tecnologia 6G

Nell'ultima decade si è assistito ad una crescita, senza fine, del traffico di rete globale. Il rapido sviluppo di applicazioni emergenti, come l'intelligenza artificiale (AI), la virtual reality (VR), il concetto di massive-IoT e l'Internet of Everything, ha portato alla generazione di un enorme volume di traffico sulla rete [3]. Di fatti, a causa dell'ormai popolare utilizzo di applicazioni video ad alta definizione, comunicazioni machine-to-machine, mobile edge service e altre applicazioni che riguardano il sempre più pervasivo paradigma IoT, di cui uno degli esempi più importanti è proprio il concetto di **Smart Grid**, il traffico mobile globale crescerà esponenzialmente fino a raggiungere un valore di 5016 exabyte al mese nel 2030 rispetto ai soli 62 exabyte al mese generati nell'anno 2020 [4][5].
Questi dati dimostrano l'importanza e la necessità di migliorare le correnti tecnologie per supportare al meglio tali applicazioni. Di fatti, il 5G non avrà la capacità di costituire una rete completamente autonoma e intelligente che sia in grado di fornire tutto come un servizio [6]. Per tali ragioni e per sopperire alle nuove necessità, il 6G è visto come la chiave che porterà nuove potenti tecnologie wireless e infrastrutture di rete innovative (come l'infrastruttura **space-air-ground/underground-see/underwater** che mira ad ottenere un accesso completamente ubiquo in qualsiasi punto della Terra) per poter realizzare una pletora di nuove applicazioni, tra cui specialmente IoT, soddisfacendo gli stringenti limiti di rete che sorgeranno anno dopo anno [7].



Evoluzione delle reti wireless, verso il 6G-IoT [2]

Di seguito si mostra una tabella (Tabella 2) che ha come scopo quello di comparare le caratteristiche dell'attuale tecnologia 5G con quelle che saranno le future caratteristiche della sesta generazione [8].

Tabella 2 – Caratteristi tecnologia 5G vs 6G

Parametro	5G	6G
Peak Data Rate	10-20 Gb/s	>1 Tbps
Latency	ms	< ms
Mobility	350 Km/h	>1000 Km/h
Traffic Density	$10\ Tb/s/Km^2$	$> 100\ Tb/s/Km^2$
Energy Efficiency	1000x relative to 4G	10x relative to 5G
End-to-End Reliability	$1 - 10^{-5}$	$1 - 10^{-7}$

Dalla tabella si può intuire come enormi saranno i benefici previsti dallo sviluppo e conseguente applicazione della tecnologia di sesta generazione per le reti. È chiaro che lo sviluppo del 6G poggerà sui blocchi costituenti dell'attuale tecnologia di rete (5G) portando però tali tecnologie al di là delle capacità mostrate e sfruttate oggigiorno. Stiamo riferendoci a tecnologie quali **Software Defined Networking**, **Network Function Virtualization**, **Network Slicing** etc… che tendono a portare le tecnologie di rete verso il concetto di piena virtualizzazione cercando di legare le funzionalità di rete il meno possibile all'hardware e rendere quindi le reti altamente flessibili, scalabili, intelligenti e fault tolerance. Per quanto riguarda l'**intelligenza** nelle reti ci si riferisce al concetto di **self-X networks**: attraverso l'impiego dell'intelligenza artificiale, del Machine Learning e di tutte le sue declinazioni l'intento è quello di giungere alla costruzione di reti che siano **completamente autonome** e quindi **self-healing**, **self-reconfiguration**, **self-optimization**, **self-learning**, **self-organization**, **self-aggregation** e **self-protection** [9][10].

Le reti elettriche stanno gradualmente mutando ed evolvendosi nell'innovativa tecnologia delle **smart grid**. Si tratta della tecnologia del futuro per la distribuzione intelligente dell'energia e in futuro ci si aspetta che le smart grid diventeranno delle reti altamente distribuite e dinamiche [11].
Nonostante i miglioramenti e la ricerca nel campo dell'IoT sia vasta in questo periodo [12], alcuni studi discutono anche del ruolo delle smart grid per quanto riguarda la gestione dell'energia e un monitoraggio affidabile nell'ambito delle smart grid IoT-enabled [13]. Le smart grid IoT-enabled, attraverso l'uso di dispositivi wireless connessi come antenne smart e altri moduli di comunicazione, permettono alle compagnie elettriche di ripristinare la corrente più rapidamente dopo il verificarsi di un blackout. Al verificarsi di un qualche evento di interruzione o blackout, trasformatori e sottostazioni IoT possono, in maniera automatica, ri-direzionare l'elettricità. Le antenne/sensori intelligenti IoT-enabled possono lavorare come dispositivi intelligenti per verificare e analizzare le condizioni dell'equipaggiamento elettrico in modo tale che se c'è la necessità di effettuare operazioni di manutenzione e riparazioni queste possano essere fatte prima che si verifichi un fallimento. Ciascun dispositivo e apparecchiatura in una smart grid può essere considerata come un oggetto al quale può essere assegnato un indirizzo IP basato sul protocollo di comunicazione 6LoWPAN.
Per garantire un sistema elettrico senza interruzioni, è essenziale monitorare e controllare continuamente il comportamento dinamico dei sistemi elettrici. Ad esempio, l'attività di **Partial Discharge** è un segnale di avvertimento di un guasto dell'apparato elettrico. Le conseguenze dei PD possono portare, su un lungo periodo, ad un blackout totale del sistema di alimentazione. La PD ha diversi fenomeni misurabili, come la radiazione a radiofrequenza (RF), i cambiamenti nelle proprietà chimiche, la caduta di tensione, l'impulso di corrente e il segnale acustico. È possibile evitare disastri catastrofici se le attività PD vengono monitorate costantemente e vengono effettuate prontamente delle "diagnosi" sul sistema.
Per un sistema elettrico ad alta affidabilità un sistema di monitoraggio real-time e non invasivo è essenziale. Gli impulsi PD da differenti risorse possono essere Ultra Wide Bande (UWB) in natura e possono avere diversi spettri di frequenza raggiungendo anche diversi GHz. Una smart grid altamente affidabile richiede un monitoraggio continuo e centralizzato nonché un sistema di diagnostica. La tecnica che fa uso di antenne intelligenti, non invasive e dispositivi IoT-based è un concetto nuovo. In particolare, in questo lavoro [12] si descrive una metodologia di progettazione di una antenna UWB per

la rilevazione di PD all'interno del range 3.02GHz – 11.17GHz. Lo sviluppo di questa antenna può essere applicato ai futuri dispositivi di monitoraggio 6G IoT-based per le smart grid.

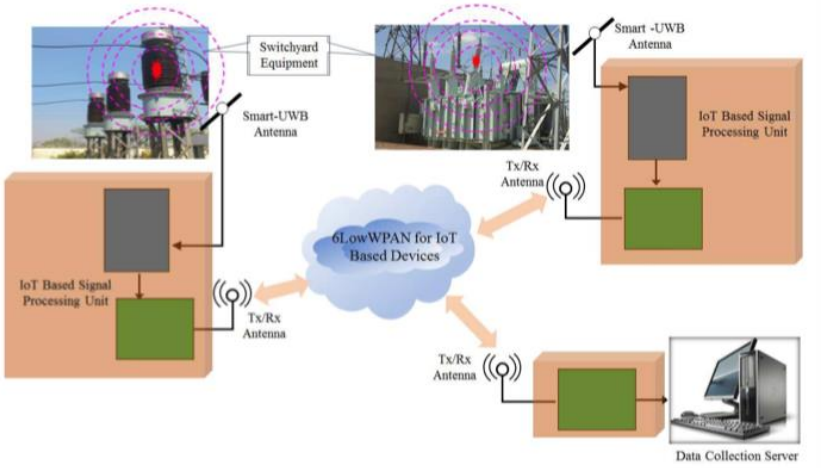


Figura 12. Rete &LowPAN per IoT

Considerando l'impiego di dispositivi IoT nello sviluppo delle smart grid, di fondamentale importanza è la questione riguardante la sicurezza dei dispositivi coinvolti. Tali dispositivi risultano essere il più delle volte **resource-and-power constrained** e non capaci di supportare la gestione aggiuntiva di un meccanismo di sicurezza o un meccanismo di sicurezza che possa definirsi abbastanza adeguato. Con l'avvento del 6G e delle nuove possibilità che saranno considerabili grazie alle sue peculiari e futuristiche caratteristiche, si prevede che questi limiti possano essere superati e nuovi meccanismi di sicurezza possano essere presi in considerazione anche per i dispositivi IoT e non solo [14]. Inoltre, le applicazioni chiave del 6G si possono identificare in UAV based mobility, Veicoli autonomi connessi alla rete, Smart Grid 2.0 etc… Tali applicazioni possono riguardare diversi stakeholder e richiedere differenti livelli di sicurezza nel 6G.

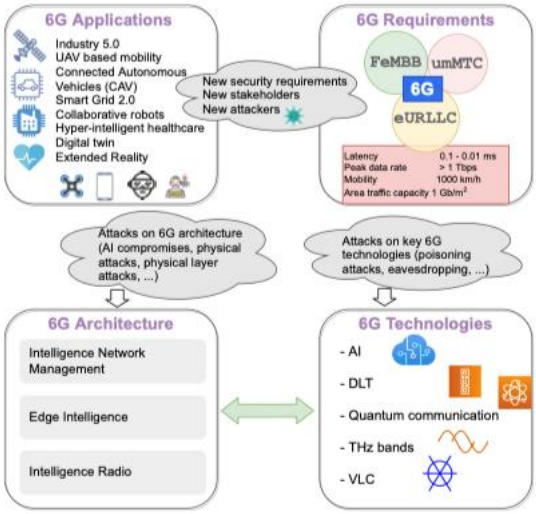


Figura 13. Applicazioni 6G

A causa della novità di questi domini applicativi e della potenza degli attaccanti, i requisiti di sicurezza e le sfide possono variare enormemente in 6G piuttosto che in 5G. Nella seguente figura (Figura 14) si mostrano i requisiti di sicurezza per ciascuna delle applicazioni citate, mettendo in risalto, però, le **smart grid**:

Potential 6G Applications	Security Requirements								Expected Security and Implantation Challenges									
	Ultra Lightweight Security	Extremely Low latency	Extreme Scalability	Zero-touch Security	High Privacy	Proactive Security	Security via Edge	Domain specific security	Limited resources	Diversity of Devices	High Mobility	Physical Tempering	Terrorist Attacks	Intermittent Connectivity	Localized environment	Lack of Security Standards	E2E Security orchestration	Energy Efficiency
UAV based mobility	M	H	H	H	L	M	H	L	H	M	H	M	H	L	L	L	H	H
Connected Autonomous Vehicles	L	H	H	H	M	H	H	H	L	M	H	M	H	L	L	L	H	M
Smart Grid 2.0	H	L	H	M	M	H	L	H	H	L	L	H	H	H	L	L	L	M
Collaborative Robots	M	H	M	H	L	L	H	H	M	L	M	M	L	L	H	L	M	M
Hyper-Intelligent Healthcare	H	H	H	M	H	M	H	H	H	H	M	M	L	M	H	M	H	H
Industry 4.0	M	H	H	H	L	H	H	H	H	H	M	L	M	L	H	M	H	H
Extended Reality	H	H	H	M	H	L	H	L	H	M	M	H	L	L	L	H	H	H

L Low Level Requirement/Impact M Medium Level Requirement/Impact H High Level Requirement/Impact

Figura 14. Sicurezza per le applicazioni 6G

Il Software Defined Internet Of Things (**SDIoT**) sta rendendo possibili diverse realtà industriali attraverso lo sharing di informazioni rilevanti per un monitoring efficace e un controllo attraverso dei controller centralizzati utilizzando tecnologie avanzate di "sensing & communication" [15]. Per esempio, nel contesto delle smart grid, le performance di una grid possono essere influenzate da cyberattack e disturbi fisici, come ad esempio fallimenti simmetrici e asimmetrici, che possono portare il sistema elettrico a condizioni di instabilità. Il framework di una smart grid è un esempio di un "**cyber-physical system**" (CPS), nel quale l'affidabilità della rete è incrementata attraverso l'impiego di infrastrutture di monitoraggio, comunicazione e controllo. Per un funzionamento efficace di una rete smart grid, è necessario un sistema informativo affidabile e sicuro dal punto di vista della sicurezza informatica. In particolare, se una smart grid è sotto attacco informatico, anche le prestazioni del livello fisico ne risentono, il che può comportare una diminuzione dell'affidabilità del sistema.
Al fine di rendere sicure e affidabili le operazioni di una rete elettrica, devono essere stabilite reti di comunicazioni sicure in modo tale che "**energy management system**" (EMS) e "**wide area measurement system**" possano operare senza interruzioni.
Una soluzione possibile è progettare un algoritmo di routing robusto tra reti eterogenee che possa consegnare le informazioni con la minima latenza soddisfacendo allo stesso tempo i requisiti di QoS.
Con l'avvento delle applicazioni AI-Driven e le reti IoT-enabled, come le smart grid, le architetture di comunicazione di quinta generazione (5G) non possono soddisfare le sfide poste in essere dai requisiti real-time dei servizi, come ad esempio la comunicazione URLLC: **Ultra Reliable Low Latency Communication**. Per esempio, uno dei requisiti per prevenire gli attacchi informatici nelle smart grid è ottenere una latenza minima per contrastare gli attaccanti e ridurre al minimo le possibilità di guasti a cascata.
L'architettura di comunicazione di sesta generazione (6G) prevista può affrontare queste sfide migliorando il data rate e ottenendo una latenza inferiore attraverso l'uso di un intervallo di frequenze più elevato e altre caratteristiche vantaggiose rispetto alle precedenti architetture di comunicazione. È necessario utilizzare un meccanismo di previsione con il 6G per limitare l'impatto degli attacchi informatici sulle reti intelligenti. Alcuni meccanismi di prevenzione includono **intrusion detection system**, che costituiscono la prima linea di difesa contro diversi tipi di attacchi. Dal punto di vista della rete intelligente, la linea di difesa iniziale è ottenuta tramite interruttori automatici o relè al fine di impedire che gli effetti a catena dei guasti si propaghino attraverso una rete di sistemi di alimentazione. Tuttavia, in situazioni in cui un meccanismo

preventivo non è in grado di mitigare i disturbi, è necessario utilizzare algoritmi di rilevamento per identificare anomalie impreviste nella rete. Tali meccanismi di rilevamento possono includere algoritmi per identificare stati del sistema non visibili, intrusi o attacchi cyber-fisici, o gli effetti di entrambe le varianti di attacco quando sono combinate [16][17][18][19]. Un altro approccio pratico è stato adottato in [20] dove è stata impiegata una tecnica di stima dello stato lineare (SE) basata su letture registrate da diverse unità di misura del fasore (PMU) e un controller robusto decentralizzato, incorporato nei singoli bus. Il controller robusto monitora lo stato dei bus e confronta anche lo stato stimato con le misurazioni delle PMU. Quando viene rilevata un'anomalia, il controller attiverà un sistema di accumulo di energia distribuito (DESS) che inietta o assorbe la potenza per fornire stabilita transitoria alla rete. Uno dei principali svantaggi di questa tecnica è che presuppone che il sistema complessivo, in tutte le sue parti, sia lineare, ma in pratica il comportamento del sistema è altamente non lineare. Con la linearizzazione si trascurano alcuni parametri importanti, che possono portare all'instabilità del sistema di alimentazione. Una delle tante possibili soluzioni per garantire un funzionamento stabile e sicuro di una rete di sistemi di alimentazione è eseguire una SE accurata. Per una migliore comprensione del funzionamento del sistema, SE è molto importante. I minimi quadrati ponderati (**WLS: Weighted least squares**) sono una delle tecniche tradizionali che stimano accuratemene lo stato di un sistema di alimentazione utilizzando letture di alta qualità, ottenute tramite PMU [21]. Tuttavia, il principale svantaggio di WLS è che non fornisce una buona approssimazione quando c'è una non linearità significativa nel profilo di potenza. In [22] è stato adottato un approccio di logica fuzzy per la stima del carico. Tuttavia, il principale inconveniente associato alla logica fuzzy è il tuning della funzione di appartenenza agli insiemi fuzzy e la complessità quando vengono approssimati un largo numero di parametri di stato non lineari. Per colmare questa lacuna, in [23] è stata proposta una stima basata su filtri di Kalman. L'addestramento di reti neurali basate sull'EFK ha mostrato risultati promettenti in SE [24]. Il principale svantaggio associato al filtro di Kalman e al WLS è la creazione delle matrici Jacobian e di covarianza dell'errore, che aumenta la complessità all'aumentare del numero di bus. Inoltre, se si osserva una non linearità significativa in una rete, WLS o il filtro di Kalman non possono stimare con precisione gli attacchi informatici. Inoltre, queste tecniche richiedono un modello di sistema accurato per eseguire SE parametrici precisi.

Una delle possibili soluzioni per prevenire cyber attacchi è l'uso efficace dell'SDIoT nelle smart grid, ossia l'uso della tecnica di Software Defined Networking. Nello SDIoT, il piano di controllo intelligente può monitorare gli stati della rete a livello globale in tempo reale e possono essere implementate tecniche di sicurezza per rilevare le minacce alla rete [25]. Si prevede che le reti 6G generino un traffico massiccio con stringenti requisiti di QoS. Le tecniche di analisi dei dati esistenti per la valutazione delle vulnerabilità delle minacce informatiche affrontano problemi di scalabilità e di ritardo durante l'addestramento sui dati. Un tale approccio per risolvere il problema del tempo di addestramento è proposto in [26], dove gli autori hanno adottato un framework abilitato per SDN che inoltra il compito di calcolo del path ad una GPU per ridurre il tempo di addestramento degli algoritmi. Pertanto, la natura centralizzata del piano di controllo SDN può essere efficace per implementare le attività di apprendimento automatico su una GPU.

Un'altra proposta riguarda un algoritmo per rafforzare l'affidabilità delle smart grid 6G-enabled per garantire sicurezza e operazioni sicure attraverso la stima dello stato reale, come tensioni e angoli di fase del sistema anche in presenza di attacchi cyber fisici. Per risolvere il problema del grande numero di connessioni e degli stati generati dal CPS, si propone uno state estimator robusto e adattivo GPU-enabled. Questo comprende un algoritmo di deep learning, una Long Short-Term Memory (LSTM) e un EKF non lineare, chiamato LSTMKF. Funziona sul controller SDN per fornire stime parametriche online dello stato. L'LSTM nell'algoritmo LSTMKF utilizza le misurazioni raccolte dalle PMU per stimare accuratamente lo stato del sistema. Quando un intruso attacca la rete di comunicazione delle PMU, provoca una sincronizzazione non corrispondente tra le diverse risorse di energia rinnovabile (RER). L'algoritmo LSTMKF proposto assegna una priorità più alta alla stima dello stato basata su LSTM e suggerisce che un intruso ha attaccato la rete. Inoltre, se i dati di input sono corrotti, verrà dato più peso alla SE delle PMU. Da questo momento in poi, sono abilitati due livelli di SE online per contrattaccare gli attacchi cyber fisici al sistema. Fino ad ora, questo lavoro [15] è il primo che introduce LSTM in un EKF per rilevare cyber attacchi nelle smart grid 6G e SDIoT-enabled.

2.6 La tecnologia LoRA

Le Low-Power Wide-Area Network (LPWAN) rappresentano un nuovo paradigma di comunicazione, che completerà le tradizionali tecnologie cellulari e wireless a corto raggio nell'affrontare i diversi requisiti delle applicazioni IoT. Esse sono un tipo di rete di telecomunicazione wireless che sono state progettate per consentire una comunicazione a lungo raggio ma con un bit rate ridotto tra i vari dispositivi connessi. In effetti, è stata progettata principalmente per consentire ai dispositivi IoT, con caratteristiche limitate, di poter comunicare senza sprecare troppo energie, sia a livello di comunicazione che a livello computazionale e per durare più a lungo possibile, perché non sempre conviene sostituire le batterie in un dispositivo IoT. Una di queste tecnologie principali risulta essere la tecnologia LoRa.
La tecnologia LoRa è una tecnologia wireless basata sulla banda ISM, viene usata per applicazioni che hanno bisogno di una grande copertura, un basso consumo energetico e un basso bitrate. Questa tecnologia è basata sul CSS (Chirp Spread Spectrum), che viene usata già da diversi anni in campo militare, grazie alla robustezza alle interferenze, ma LoRa è la prima tecnologia che l'ha implementata per uso commerciale. Da ciò è nato LoRaWAN, la quale è stata progettata e sviluppata dalla LoRa Alliance, lo sviluppo è iniziato dal basso in modo da ottimizzare le reti LPWAN, quindi ottimizzando la durata della batteria, l'ampio raggio di copertura, la scalabilità, l'affidabilità e i costi [27][28]. Inoltre, tra questi due termini, LoRa e LoRaWAN, come si è potuto già capire bisogna fare attenzione perché spesso vengono utilizzati come sinonimi, mentre in realtà hanno dei significati differenti [28][29]:

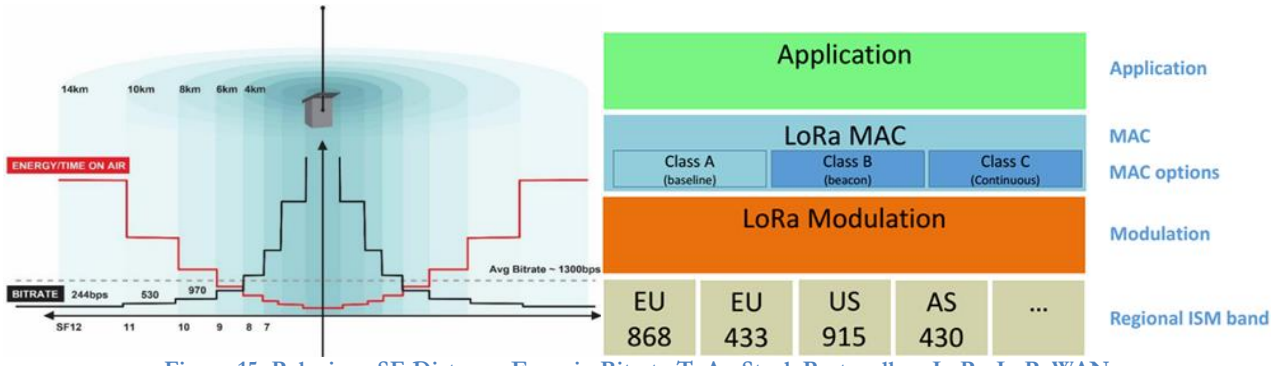


Figura 15. Relazione SF-Distanza-Energia-Bitrate-ToA - Stack Protocollare LoRa-LoRaWAN

Quando si parla di 'LoRa' si fa riferimento al livello fisico o alla modulazione wireless, essa si basa sulla modulazione chirp Spread Spectrum. Questa tecnica, come è stato appena detto, è stata utilizzata per diversi anni nelle comunicazioni militari e anche in quelle spaziali, questo grazie alle lunghe distanze di comunicazione che possono essere raggiunte e soprattutto per la robustezza alle interferenze causate da altre tecnologie che utilizzano le stesse frequenze. Da come si può vedere dalla immagine qui accanto si può dedurre che all'aumentare dello SF aumenta la distanza che può percorrere il segnale, questo perché aumenta la dimensione del pacchetto in maniera notevole ma allo stesso tempo è molto robusto e quindi anche se arriva con una potenza molto piccola al ricevitore si riesce comunque a ricevere in maniera corretta il pacchetto.
Per quanto riguarda invece il termine 'LoRaWAN' si fa riferimento al protocollo specifico che è stato progettato sulla base della tecnologia LoRa da parte della LoRa Alliance. Esso è un protocollo che definisce gli strati superiori a LoRa PHY; infatti, viene anche definito come un protocollo di rete. Può essere anche visto come estensione del livello fisico di LoRa, esso è stato standardizzato da LoRa Alliance [28]. Si occupa come viene organizzata la rete, la sua struttura e architettura, come accedere al mezzo, cosa fare quando un messaggio non arriva a destinazione e inoltre definisce tre tipologie di classi di dispositivi: A, B e C.
I vantaggi di LoRa sono molteplici, ma forse quello più notevole è la copertura vastissima, infatti, un singolo Gateway o Base station ha la possibilità di coprire un'intera città o 100km^2 (10km x 10km). Naturalmente l'area di copertura dipende molto dall'ambiente e dai vari ostacoli presenti in un'area. Ma il link budget di LoRa e LoRaWAN è uno dei maggiori a confronto con tutte le altre tecnologie di comunicazione standardizzata e questo, infatti, determina il raggio di copertura in un

ambiente. Il link budget di LoRa è di circa 155 dB. Questa copertura così ampia non sarebbe possibile in una rete basata solo sulla tecnologia WiFi; infatti, avremmo bisogno di un numero molto grande di Base Station Wi-Fi che significherebbe un costo molto più alto a livello economico[30]. Questa tecnologia però, siccome è adatta per le LPWAN, viene utilizzata in applicazioni dove non è richiesta una mole molto grande di dati come, ad esempio, lo streaming di video o di dati molto grandi, ma piuttosto il trasferimento di dati acquisiti da sensori, e delle informazioni diagnostiche relative ai dispositivi collegati in rete e trasmessi non frequentemente. Quindi in poche parole le caratteristiche sono:

- Ampio raggio di copertura (anche fino a 10km).
- Basso bitrate.
- Invio NON frequente di dati (basso duty cycle).
- Scalabile: si riesce facilmente a fare comunicare migliaia di nodi con un singolo Gateway.
- Basso consumo di energia, questo grazie al CSS e alla sensitività bassa che ha un ricevitore LoRa.

La rete LoRa è formata principalmente da quattro elementi base che sono:

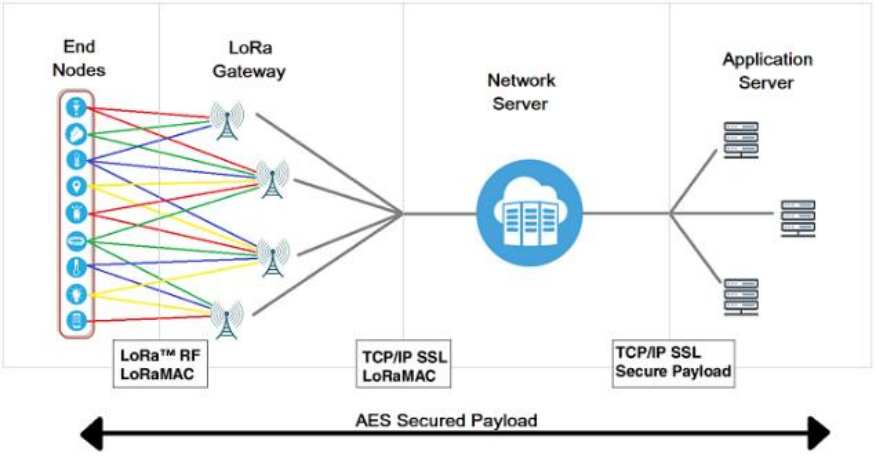


Figura 16. Architettura LoRa

- LoRa node, End Points, End Device, End Nodes (ED): questo sono i dispositivi finali della rete che possono essere dei sensori o degli attuatori;
- LoRa Gateway (GW): può essere definito come concentratore di informazioni, esso è collegato a ogni ED della rete e il compito principale del GW è quello che appena riceve un segnale 'riconoscibile' quindi che riesce a leggere lo inoltra verso il NS tramite un'altra tecnologia, come ad esempio Ethernet, Reti cellulari, Wi-Fi ecc.;
- Network Server (NS): è un elemento che analizza i dati proveniente dalla rete. Sui singoli pacchetti ricevuti vengono eseguiti dei controlli di sicurezza e poi se tutto va a buon fine vengono generati gli ACK e può applicare anche il meccanismo ADR;
- Application Server (AS): ha il compito di elaborare i messaggi dati specifici dell'applicazione ricevuti dai dispositivi finali

Smart Grid con LoRa

Per le Smart Grid (rete intelligente), le tecnologie di comunicazione e gli algoritmi di Routing sono molto importanti nella trasmissione dei dati. Tra le varie tecnologie di trasmissione dati, LoRa è una tecnologia di comunicazione wireless basata sull'Internet of Things (IoT), caratterizzata dalla sua lunga distanza di comunicazione fino a qualche chilometro e dal basso consumo energetico. Pertanto, LoRa è un candidato appropriato per la trasmissione dei dati nelle Smart Grid [31].

Attualmente però in diverse parti del mondo per i problemi di smart metering e automazione nella rete di distribuzione vengono utilizzati i servizi di telefonia mobile. In [32] è stato proposto di costruire una rete IoT a basso costo, ovvero di utilizzare rete LoRa. Per prima è stato definito quale sarebbe stata l'area HAN e NAN e parte della WAN che lo stesso studio di miglioramento della copertura dovrebbe essere realizzato, utilizzando partnership commerciali o implementazione di una propria rete per l'accesso. Poi è stata effettuata anche un'analisi economica dove gli obbiettivi M2M sono stati utilizzati come riferimento di costo più basso e la rete satellitare a bassa orbita (BGAN) come riferimento di costo contrattuale più alto nell'analisi degli endpoint. Inoltre, nel mercato del servizio al dettaglio la tecnologia LPWAN è la più adeguata grazie all'elevato volume di endpoint, dato che i costi dei servizi sono fino a dieci volte inferiori ai costi della rete cellulare. Inoltre, si possono anche sfruttare le infrastrutture del cliente attraverso l'implementazione di reti intelligenti sicure (Intelligent WAN o IWAN), che attraverso l'installazione di un firewall collegato al datacenter possono fornire la connettività degli elementi residenziali, commerciali e di automazione della misurazione. Nell'immagine successiva si possono vedere come tutte le tipologie di reti e i servizi potrebbero coesistere sulla stessa rete e il datacenter potrebbe trovarsi in una qualsiasi località.

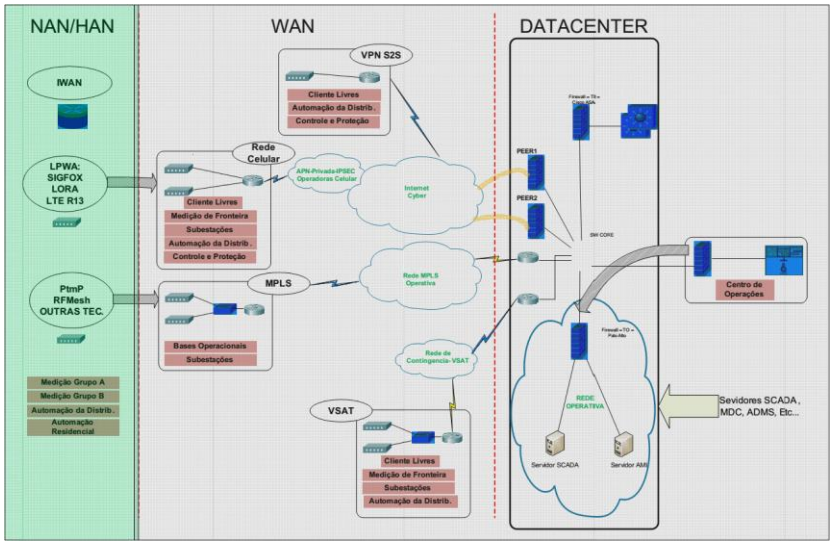


Figura 17. Rural Smart Grid Architecture

Dopodiché è stato effettuato un confronto tra una classica Smart Grid che utilizza una tecnologia RF Mesh e una Smart Grid che utilizza la tecnologia LoRa con i seguenti risultati:

RF Mesh x LoRaWAN		
	RF Mesh	**LoRaWAN**
Topology	Mesh	Star
Maximum Data Rate per Terminal	10-100 kbps	50 kbps
Average Latency	700 ms per hop (remcommended up to 3 hops)	1 s
Maximum Aggregation per Concentrator	10,000 terminals	15,000 terminals
Outdoor RF Concentrator Average Cost per Terminal	US$ 0,50	US$ 0,07
Technology Maturity Level	S.G.: Estabilished IoT: In development	In positioning
Mobility of Endpoints	Possible with restrictions	Possible

Figura 18. Confronto tra RF Mesh e LoRaWan

Dalla Tabella possiamo dedurre che con LoRa siamo vincolati ad una topologia a stella ma abbiamo delle performance più o meno simili ma con il vantaggio di poter utilizzare molti più dispositivi e nettamente più economici. Questo sistema è stato testato nella città di Santa Maria in Brasile [32]. Un altro utilizzo di LoRa applicato all'ambito delle Smart Grid riguarda le AMI (Advanced Meter Infrastructure) [33], ovvero un'infrastruttura avanzata dei contatori, le risorse energetiche distribuite e i veicoli elettrici. Tra queste applicazioni, le AMI rappresentano il primo passo verso la futura implementazione delle smart grid, consentendo la lettura e la registrazione del consumo di energia elettrica su richiesta o in base anche a una pianificazione. In questo studio, l'obiettivo principale è la progettazione di un'architettura basata sull'IoT per supportare la segnalazione e la fatturazione giornaliera degli AMI in una rete residenziale. In primo luogo, viene discussa la modellazione della rete per AMI. La tecnologia a lungo raggio, LoRa, è stata studiata e considerata come uno dei candidati promettenti per le reti a lungo raggio a bassa potenza (LPWANs). Un'analisi completa dell'architettura basata su LoRa è data per un caso di studio di una reale rete residenziale, in Puerto Montt, Cile. I risultati sono analizzati in funzione al rapporto di consegna dei pacchetti, l'energia consumata, throughput, numero di collisioni e distribuzione della frequenza. Un'architettura tipica per tale applicazione è mostrata nella seguente figura:

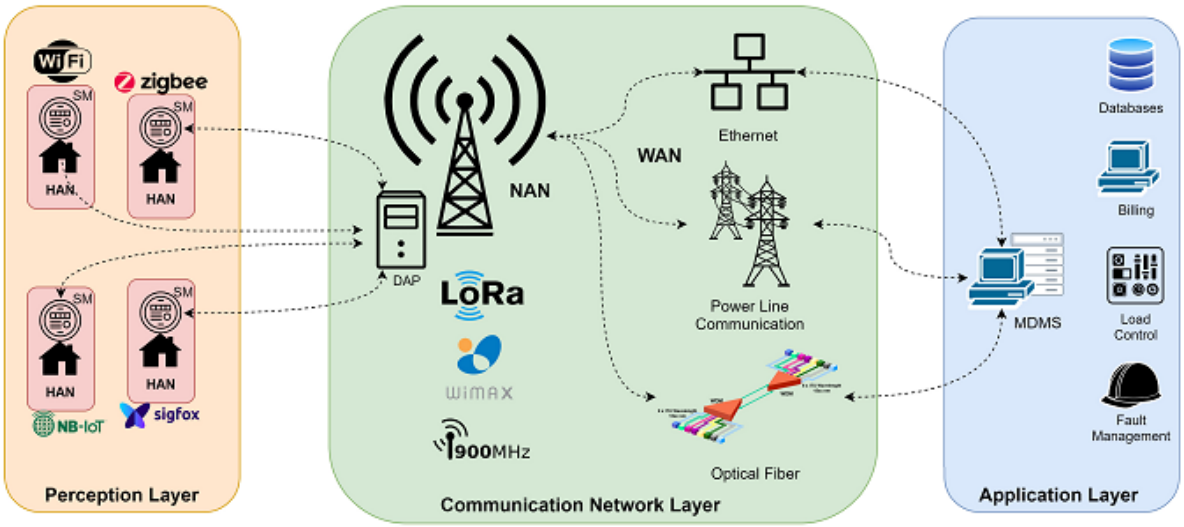


Figura 19. Architettura IoT-based per smart grid AMI in ambiente residenziale

L'infrastruttura di misurazione avanzata (AMI) consiste di quattro diversi domini di rete: home area network (HAN), neighborhood area network (NAN), wide area network (WAN), e utility network (UN). La HAN è legata alla comunicazione tra gli apparecchi e lo smart meter (SM). La NAN è relativa alla comunicazione tra i contatori intelligenti e un dispositivo concentratore chiamato punto di aggregazione dati (DAP), che funge da gateway per raccogliere i dati dalle case. La WAN è relativa al canale di comunicazione tra i DAP e la rete di utilità, che deve consegnare i dati generati dagli SM alle sedi di utilità. La UN è legata a tutti i task relativi ai dati di cui le utility sono responsabili, come i sistemi di fatturazione, i programmi di risposta alla domanda (DR), il controllo e l'amministrazione della rete.
D'altra parte, le tecnologie Internet of Things (IoT) permettono agli oggetti fisici di vedere, sentire, pensare ed effettuare processi facendoli ''parlare'' insieme per condividere informazioni e coordinare le decisioni. Le tecnologie IoT trasformeranno i dispositivi statici in dispositivi intelligenti sfruttando le loro tecnologie di base, come l'informatica ubiqua e pervasiva, i dispositivi embedded e le tecnologie di comunicazione. In termini di dimensioni, gli oggetti intelligenti con i loro compiti generano applicazioni e servizi specifici che possono trasformarli in diversi domini. Il principale vantaggio dell'uso dell'IoT nelle reti intelligenti è il contributo alla qualità della vita e alla crescita dell'economia mondiale. Poiché questo aprirà un nuovo mondo di benefici, le applicazioni devono crescere proporzionalmente per soddisfare le esigenze dell'industria e dei clienti; e i dispositivi devono anche essere sviluppati per adattarsi alle esigenze dei clienti in termini di disponibilità ovunque e in qualsiasi momento. Per fondere entrambi i mondi già descritti, è necessario realizzare alcune caratteristiche: un'architettura stratificata flessibile, data l'eterogeneità dei possibili oggetti connessi e dei compiti critici che devono completare. Per soddisfare questi requisiti, diversi autori hanno proposto diverse architetture come argomento generale per affrontare questo compito. La figura 19 mostra una tipica architettura IoT-based per smart grid AMI in ambiente residenziale.
Spieghiamo nel dettaglio i livelli mostrati nella Figura 19:

- **Perception Layer**: consiste in nodi di sensori e dispositivi di misurazione. Qui, il livello fisico dei sensori copre la raccolta e l'elaborazione dei dati. Si possono trovare diversi tipi di nodi sensori e dispositivi di misurazione come la lettura dei contatori, l'interrogazione della posizione, la lettura della temperatura, i sensori di movimento e l'umidità. Diversi dispositivi possono essere trovati anche a livello di casa/edificio, come gli elettrodomestici, i sistemi di gestione dell'energia domestica (HEMS) e gli SM.
- **Communication Network Layer**: è incaricato di comunicare tra i dispositivi finali (livello di percezione) con i server MDMS e viceversa. Il primo compito di questo strato è quello di permettere il trasferimento dei dati sotto canali di misurazione sicuri utilizzando diversi mezzi di trasmissione. La seconda funzione è la funzione di memorizzazione, che può essere soddisfatta utilizzando il software middleware che comunica con i server di memorizzazione e i diversi dispositivi che si possono trovare. WAN e NAN sono le reti principali in questo senso, e devono essere progettate per soddisfare diversi requisiti di affidabilità e disponibilità per i collegamenti dati.
- **Application Layer**: è incaricato di tutti i compiti relativi all'elaborazione dei dati, alla visualizzazione e alla disposizione dei dati ai clienti e alle utenze per le diverse applicazioni che sono destinate. Per i clienti, i servizi intelligenti e la gestione remota devono soddisfare diversi requisiti. Per le utility, i compiti lato business includono applicazioni di modelli di business, grafici, diagrammi di flusso e analisi di big data per i dati percepiti dal livello di percezione. Sotto questo dominio, le seguenti applicazioni possono coesistere: cruscotto utente e display di misurazione, sistema di gestione dei dati del contatore (MDMS) e sistemi di fatturazione. Un vincolo importante per questo livello è che, poiché sia i dati dei clienti che quelli delle utility saranno memorizzati e gestiti insieme, entrambi devono condividere un protocollo comune o un'interfaccia di comunicazione per parlare con altri sistemi.

2.7 Conclusioni

Dall'analisi effettuata quindi, appare chiaro che le caratteristiche tecniche ed i servizi offerti dalle reti NB IoT, 4G-LTE e dalla rete 5G sono notevolmente diversi e variegati sia per throughput, latenza, consumi distanza coperta etc. e, pertanto nella

fase di sviluppo dei singoli dimostratori, in funzione delle esigenze e specifiche tecniche e delle necessità trasmissive dei singoli nodi/oggetti, verrà individuata di concerto con i rispettivi responsabili di Obiettivo Realizzativo, le soluzioni più idonee allo scopo. E' in dubbio che la disponibilità della rete 5G aumenterà la flessibilità dei sistemi allo studio in questo OR e saranno di importanza chiave nello sviluppo dei dimostratori previsti in altri OR del progetto ComESto, dove la rete è presente.
Opportunamente equipaggiati, gli apparati periferici individuati nelle attività precedenti, ai fini dell'interfacciamento verso il framework COMESTO e l'accesso al Cloud ove previsto; con la tecnologia 5G potranno essere raggiunti e collegati anche in eventuale assenza di LAN, Wi.Fi. e accessi Xdsl.

3 STANDARD IEC-61850

IEC 61850 è uno standard per la progettazione dei sistemi di automazione per le sottostazioni elettriche. Fa parte della Commissione Elettrotecnica Internazionale (International Electrotechnical Commission o IEC in inglese) Commissione Tecnica 57 (TC57) [34]. Il modello di dati astratto definito nel IEC 61850 può essere "mappato" su un diverso numero di protocolli. Sono previste le mappature su MMS, GOOSE, Sampled Values (SMV) e alcuni servizi web. Generalmente questi protocolli girano su reti TCP/IP o LAN di stazione con switch ethernet molto performanti per rispondere ai requisiti stringenti dei dispositivi di protezione, che necessitano tempi di risposta inferiori a 4 o 5 millisecondi.
Questo standard propone una soluzione unificata dell'aspetto di comunicazione per l'automazione delle sottostazioni. Oggi, le sottostazioni elettriche sono per lo più gestite da sistemi di automazione delle sottostazioni. Questi sistemi utilizzano computer e applicazioni specifiche per ottimizzare la gestione delle apparecchiature delle sottostazioni e per migliorare l'efficienza operativa e di manutenzione con un intervento umano minimo [35].
Un tempo, i sistemi di automazione delle sottostazioni utilizzavano protocolli di comunicazione semplici, diretti e altamente specializzati [36]. Questi protocolli si preoccupavano meno della semantica dei dati scambiati, i cui tipi di dati erano relativamente primitivi. L'attrezzatura e i sistemi erano semplici. Tuttavia, oggi i sistemi di automazione delle sottostazioni non possono più godere di tale semplicità a causa della loro crescente complessità: le apparecchiature diventano più intelligenti e la maggior parte di quei vecchi sistemi semplici sono stati gradualmente sostituiti da sistemi aperti, che abbracciano il vantaggio di tecnologie emergenti come i sistemi di database relazionali, il funzionamento multi-task dei sistemi e il supporto per la tecnologia di visualizzazione grafica all'avanguardia.
Inoltre, dispositivi di produttori diversi utilizzavano protocolli di automazione di sottostazioni [37][38][39] diversi, impedendogli di dialogare tra loro. Oggi la maggior parte dei produttori di dispositivi ne ha riconosciuto la necessità per uno standard internazionale unificato per supportare la cooperazione senza soluzione di continuità tra prodotti di diversi fornitori.
Lo standard internazionale IEC 61850, redatto da esperti del dominio dell'automazione delle sottostazioni di 22 Paesi, cerca di affrontare la situazione di cui sopra. Questo standard si avvale di un modello di dati completo orientato agli oggetti e della tecnologia Ethernet, riducendo notevolmente i costi di configurazione e manutenzione. A differenza del suo predecessore, l'Utility Communication Architecture protocollo 2.0 (UCA 2.0) [39], lo standard IEC 61850 è progettato per essere compatibile con domini oltre all'automazione delle sottostazioni. Per rendere il nuovo protocollo meno dipendente dal dominio, il comitato dello standard ha cercato di enfatizzare la semantica dei dati, ritagliando la maggior parte dei dettagli di comunicazione.

Oggi, la rete elettrica sta attraversando una transizione significativa verso una rete intelligente, affidabile e completamente automatica che si chiama Smart Grid.
L'automazione delle sottostazioni è uno dei servizi derivati dalla Smart Grid e può essere realizzato incorporando tecnologie informatiche all'avanguardia con il sistema di alimentazione. La chiave della Smart Grid è la rete di comunicazione che funge da struttura informativa fondamentale per fornire comunicazioni bidirezionali end-to-end nella Smart Grid. Sebbene una miriade di tecnologie di comunicazione esistenti possano essere applicate alla Smart Grid, i nuovi protocolli di comunicazione e il miglioramento dei protocolli esistenti sono una parte indispensabile della Smart Grid.
Nell'ultimo decennio, industrie e persino clienti residenziali si stanno muovendo verso un mondo digitale. Pertanto, si prevede che ogni componente dello standard possieda una sorta di impostazione, monitoraggio e controllo. Per consentire ai dispositivi di comunicare tra loro, è necessario un nuovo modello di comunicazione per migliorare l'interoperabilità e l'intercambiabilità tra i dispositivi. Tale modello è stato sviluppato e standardizzato nella Commissione Elettrotecnica Internazionale (IEC) 61850 – "Reti e sistemi di comunicazione di servizi pubblici in sottostazioni"[40],[41],[42].
La famiglia IEC 61850 di standard di sistemi di comunicazione per sottostazioni è stata rilasciata nei primi anni 2000. Questi standard includono connessioni a livello di processo basate su Ethernet [43] tra commutatori e sale di controllo; tuttavia, poiché è uno standard molto nuovo nel settore elettrico, le sue prestazioni in servizio dovrebbero essere testate accuratamente per assicurarsi che funzioni correttamente poiché la rete elettrica è una rete molto sensibile e i proprietari

di queste reti potrebbero perdere milioni di euro in caso di implementazione errata. Talvolta si ritiene erroneamente che lo standard IEC 61850 sia un altro protocollo di comunicazione, ma in realtà è una serie ben strutturata e coerente di pubblicazioni che definisce una serie di requisiti di sistema. Poiché il sistema di alimentazione è molto sensibile e costoso, è molto importante prevedere in anticipo il comportamento effettivo di qualsiasi cambiamento in esso, prima di metterlo in azione. Tuttavia, a causa dei numerosi vantaggi dello standard IEC 61850 per le utility e della facilità d'uso in sottostazione, i fornitori multinazionali come ABB, Siemens o Schneider Electric e così via, stanno procedendo verso la standardizzazione dei loro prodotti rispetto a questo standard.

I dispositivi compatibili IEC 61850, contrariamente ai protocolli legacy, sono in grado di funzionare tra loro; quindi, se una sottostazione è dotata di dispositivi di protezione ABB, nel caso in cui uno di essi necessiti di manutenzione, può essere sostituito con il dispositivo di protezione di Siemens. Non è più necessario sostituire un dispositivo con un altro della stessa marca.

Lo standard IEC 61850 (reti e sistemi di comunicazione nelle sottostazioni) è stato sviluppato per fornire interoperabilità tra *dispositivi elettronici intelligenti* (**Intelligent Electronic Devices** (**IEDs**)) per la protezione, il monitoraggio, il controllo e l'automazione nelle sottostazioni [44]. IEC 61850 non si occupa solo della comunicazione ma anche della modellizzazione delle informazioni, adattata alle esigenze del settore dell'energia elettrica. Inoltre, definisce un linguaggio di configurazione basato su XML che standardizza l'ingegneria / configurazione dei dispositivi di automazione delle sottostazioni. Sebbene IEC 61850 riguardi originariamente l'automazione delle sottostazioni, esistono già ulteriori modelli di informazione definiti in base al modello IEC 61850 per altri settori come le turbine eoliche in IEC 61400-25 o le centrali idroelettriche in IEC 61850-7.
IEC 61850 è uno standard internazionale progettato per lo scambio sicuro di informazioni all'interno di un sistema di alimentazione. Sviluppato originariamente per l'automazione delle sottostazioni, oggi copre anche le **risorse energetiche distribuite** (**Distributed Energy Resources** (**DER**)) e, in aggiunta, la sicurezza delle informazioni - all'interno dello stesso framework secondo IEC TC57. IEC 61850 non è solo un protocollo in grado di scambiare un blocco di dati da A a B: è anche un modello informativo, che definisce una convenzione di denominazione unica per tutti i blocchi all'interno del sistema di alimentazione e della struttura DER.
Lo standard IEC 61850 è composto da più parti che affrontano problematiche diverse, vedi Figura 20 e Figura 21. Le diverse parti sono riassunte nella figura successiva. Le parti da 1 a 5 forniscono una panoramica e definiscono i requisiti, compresi i requisiti relativi all'hardware (parte 3) e alla progettazione (parte 4).

Part #	Title
1	Introduction and Overview
2	Glossary of terms
3	General Requirements
4	System and Project Management
5	Communication Requirements for Functions and Device Models
6	Configuration Description Language for Communication in Electrical Substations Related to IEDs
7	Basic Communication Structure for Substation and Feeder Equipment
7.1	- Principles and Models
7.2	- Abstract Communication Service Interface (ACSI)
7.3	- Common Data Classes (CDC)
7.4	- Compatible logical node classes and data classes
8	Specific Communication Service Mapping (SCSM)
8.1	- Mappings to MMS (ISO/IEC 9506 – Part 1 and Part 2) and to ISO/IEC 8802-3
9	Specific Communication Service Mapping (SCSM)
9.1	- Sampled Values over Serial Unidirectional Multidrop Point-to-Point Link
9.2	- Sampled Values over ISO/IEC 8802-3
10	Conformance Testing

Figura 20. Parti dello standard IEC 61850

Il meta modello dello IEC 61850 con concetti come **Logical Node** (**LN**) e classi di dati è definito nella parte 7.1 e 7.2. La **Substation Configuration Language** (**SCL**) definita nella parte 6 è usata per configurare le sorgenti di dati così come ricevere informazioni dalle sorgenti di dati. Il mapping della tecnologia dai servizi astratti a quelli concreti è dato in 8.1 così come in 9.1 e 9.2. L'8.1 definisce un mapping a **Manifacturing Messaging Specification** (**MMS**) e Ethernet per **Generic Object-Oriented Substation Events** (**GOOSE**). La parte 9.1 definisce un mapping dei valori campionati verso la connessione seriale e 9.2 verso Ethernet.

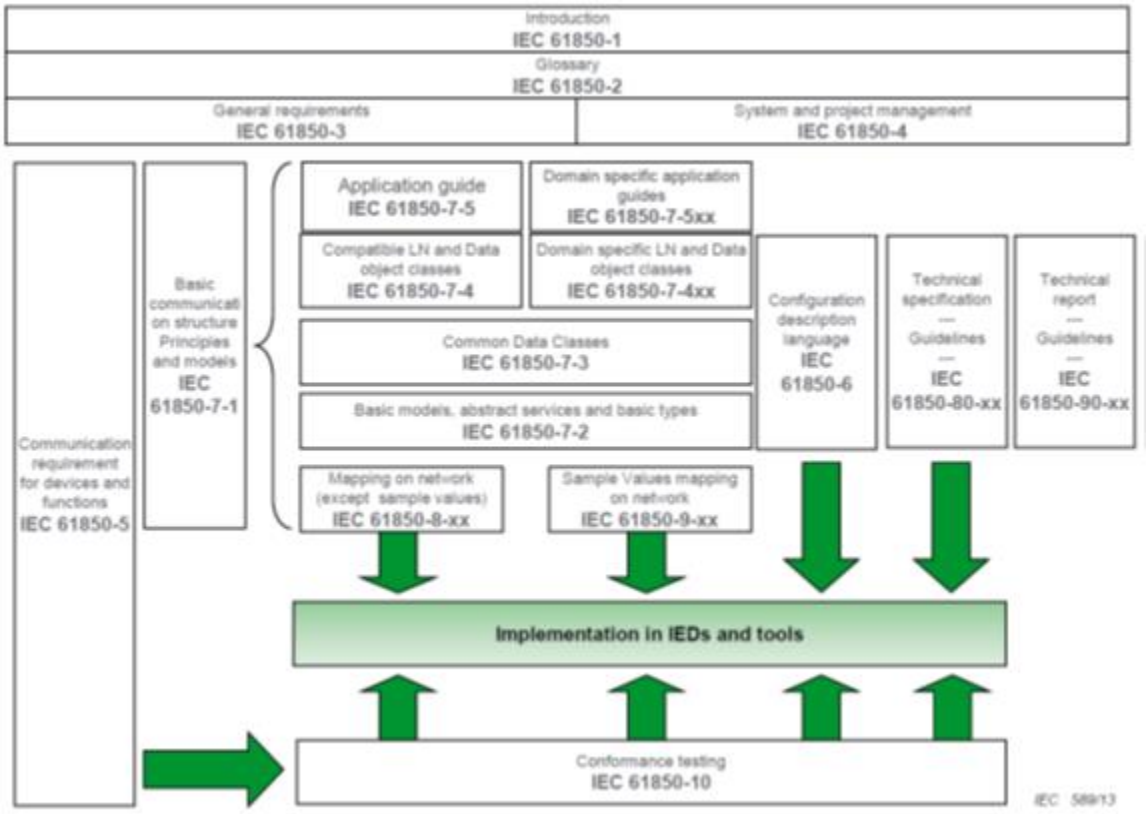


Figura 21. La serie di standard IEC 61850

Lo standard IEC 61850 e i relativi standard possono essere raggruppati come di seguito:

General information including basicterms and definition
IEC 61850 Part 1: Introduction and overview
IEC 61850 Part 2: Glossary
IEC 61850 Part 3: General requirements
IEC 61850 Part 4: System and project management
IEC 61850 Part 5: Communication requirements for functions and device models
IEC/TS 62351-1: Introduction
IEC/TS 62351-2: Glossary of Terms
IEC/TR 62351-12: Resilience and Security Recommendations for Power Systems with DER

Configuration and guidelines
IEC 61850 Part 6: Configuration description language for communication in electrical substations related to IEDs
IEC 61850 Part 90-1: Use of IEC 61850 for the communication between substations
IEC 61850 Part 90-2: Using IEC 61850 for the communication between substations and control centres
IEC 61850 Part 90-3: Using IEC 61850 for condition monitoring
IEC 61850 Part 90-4: Network Engineering Guidelines - Technical report
IEC 61850 Part 90-5: Using IEC 61850 to transmit synchro phasor information according to IEEE C37.118
IEC/TR 62351-13: Guidelines on What Security Topics Should Be Covered in Standards and Specifications
IEC/TR 62351-90-1: Guidelines for Using Part 8 Roles

Information model

IEC 61850 Part 7-1: Basic communication structure – Principles and models
IEC 61850 Part 7-3: Basic communication structure – Common data classes
IEC 61850 Part 7-4: Basic communication structure – Compatible logical node classes and data classes
IEC 61850 Part 7-410: Hydroelectric power plants – Communication for monitoring and control
IEC 61850 Part 7-420: Basic communication structure – Distributed energy resources logical nodes
IEC 61850 Part 7-5: IEC 61850 – Modelling concepts
IEC 61850 Part 7-500: Use of logical nodes to model functions of a substation automation system
IEC 61850 Part 7-510: Use of logical nodes to model functions of a hydro power plant
IEC 61850 Part 7-520: Use of logical nodes to model functions of distributed energy resources
IEC 61850 Part 90-7: Object models for power converters in distributed energy resources systems
IEC 61850 Part 90-8: Object Model for E-Mobility – now a joint activity (JWG11) with IEC TC69
IEC 61400-25-4: Basic communication structure for Wind Turbinesas, Wind turbines – Communications for monitoring and control of wind power plants.

Protocols and services

IEC 61850 Part 7-2: Basic communication structure – Abstract communication service interface (ACSI)
IEC 61850 Part 8-1: Specific communication service mapping (SCSM) – Mappings to MMS (ISO 9506-1 and ISO 9506-2) and to ISO/IEC 8802-3
IEC 61850 Part 8-2: Communication networks and systems for power utility automation - Part 8-2: Specific communication service mapping (SCSM) - Mapping to Extensible Messaging Presence Protocol (XMPP)
IEC 61850 Part 80-1: Guideline to exchange information from a CDC based data model using IEC 60870-5-101/104
IEC 61850 Part 80-4: Translation from COSEM object model (IEC 62056) to the IEC 61850 data model
IEC 61850 Part 9-2: Specific communication service mapping (SCSM) – Sampled values over ISO/IEC 8802-3

Conformance testing

IEC 61850 Part 10: Conformance testing
IEC 62351-100-1: Conformance test cases for IEC 62351-5 and companion standards

Cyber security

IEC/TS 62351-3: Security for profiles including TCP/IP
IEC/TS 62351-4: Security for profiles including MMS
IEC/TS 62351-6: Security for IEC 61850 profiles
IEC/TS 62351-7: Objects for Network Management
IEC/TS 62351-8: Role-Based Access Control
IEC/TS 62351-9: Key Management
IEC/TS 62351-10: Security Architecture
IEC 62351-14 Security Event Logging and Reporting
IEC/TR 62351-90-2 Deep Packet Inspection

3.1 Funzionalità di base e formato protocollare

Di seguito vengono evidenziate quelle che sono le funzioni principali dello standard.

3.1.1 IEC 61850: Standard for the Digital Substation

Lo standard IEC 61850, intitolato "Reti e sistemi di comunicazione per l'automazione delle utenze elettriche", definisce l'architettura complessiva e i protocolli di comunicazione da utilizzare tra gli IED nelle sottostazioni digitali [45]. L'intento dello standard è di abilitare l'interoperabilità tra IED di diversi produttori definendo protocolli di comunicazione comuni e modelli di informazione che devono essere supportati da tutti loro.

3.1.2 Architettura della sottostazione

3.1.2.1 Attrezzatura primaria nella sottostazione

Lo scopo di una sottostazione è trasformare la tensione e la corrente dell'energia che scorre sulle linee elettriche in entrata [46]. Un esempio di uno schema unifilare che mostra l'apparecchiatura primaria di una sottostazione di distribuzione da IEC 61850-5 [47] è mostrato nella Figura 22. La sottostazione D2-1, come viene chiamata, contiene due vani trasformatore (E01 ed E02) e sei vani alimentatore (da K02 a K07, anche se alcuni non sono completamente disegnati) e un vano di commutazione (switching) (K01). La potenza in ingresso da più linee di trasmissione passa prima attraverso gli alloggiamenti dei trasformatori, dove la sua tensione e corrente vengono modificate dai trasformatori (disegnati utilizzando due cerchi). L'alimentazione fluisce quindi agli alloggiamenti di alimentazione utilizzando il bus.

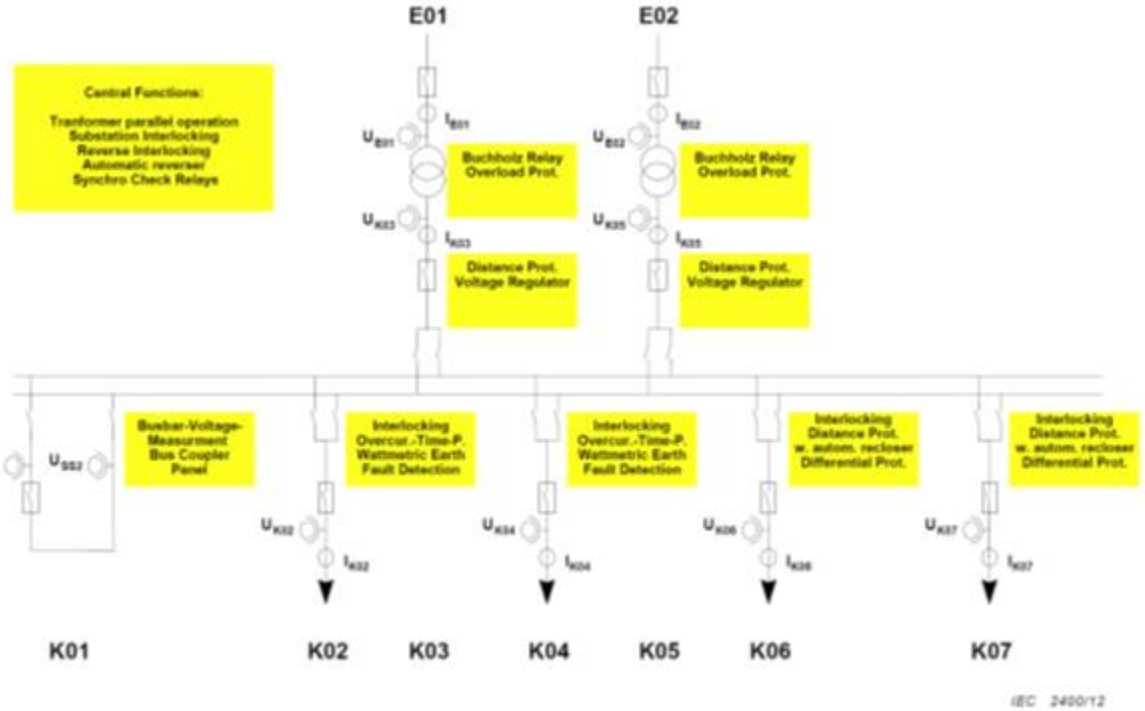


Figura 22. Esempio di sottostazione D2-1 in accordo allo standard IEC 61850

In ogni vano, i trasformatori di corrente (**Current Transformer - CT**) (etichettati da I_{E01} a I_{K07}) e i trasformatori di tensione (**Voltage Transformer - VT**) (etichettati da U_{E01} a U_{K07}) vengono utilizzati per ottenere misurazioni della potenza all'interno dell'alloggiamento. Si noti che nel caso degli alloggiamenti del trasformatore, le misurazioni sono ottenute da entrambi i lati del trasformatore per monitorarne la funzionalità. Inoltre, gli interruttori di circuito (**Circuit Breaker - CB**) (disegnati come rettangoli) trovati in ogni vano vengono utilizzati per interrompere il flusso di elettricità quando necessario. Ciò può essere

fatto per trasferire carichi, isolare parti della sottostazione o mitigare i guasti rilevati dai relè che monitorano i valori dei CT e dei VT [46].

3.1.2.2 Rete e livelli di comunicazione

Oltre all'apparecchiatura primaria, la sottostazione digitale comprende anche IED interconnessi tramite una rete di comunicazione. Ad esempio, una rete di comunicazione aggiunta alla sottostazione di distribuzione D2-1 si trova nel lavoro di Zhang et al. [48] e mostrato nella Figura 23. In ogni postazione monitorata da CT e VT è presente un'unità di fusione (**Merging Unit - MU**), un IED dell'interruttore per ogni CB e un IED di **protezione e controllo** (**P&C**) per vano. Le **MU** in un alloggiamento trasmettono le misurazioni dai CT e VT all'IED P&C utilizzando il protocollo **Sampled Values** (**SV**) [48]. Il P&C IED monitora le misurazioni. Se rileva un guasto in base alla sua configurazione, invia un segnale di scatto utilizzando il protocollo **GOOSE** (**Generic Object Oriented Substation Event**) ai relativi IED dell'interruttore, che quindi fa scattare il CB [48]. I protocolli GOOSE e SV sono discussi più avanti.

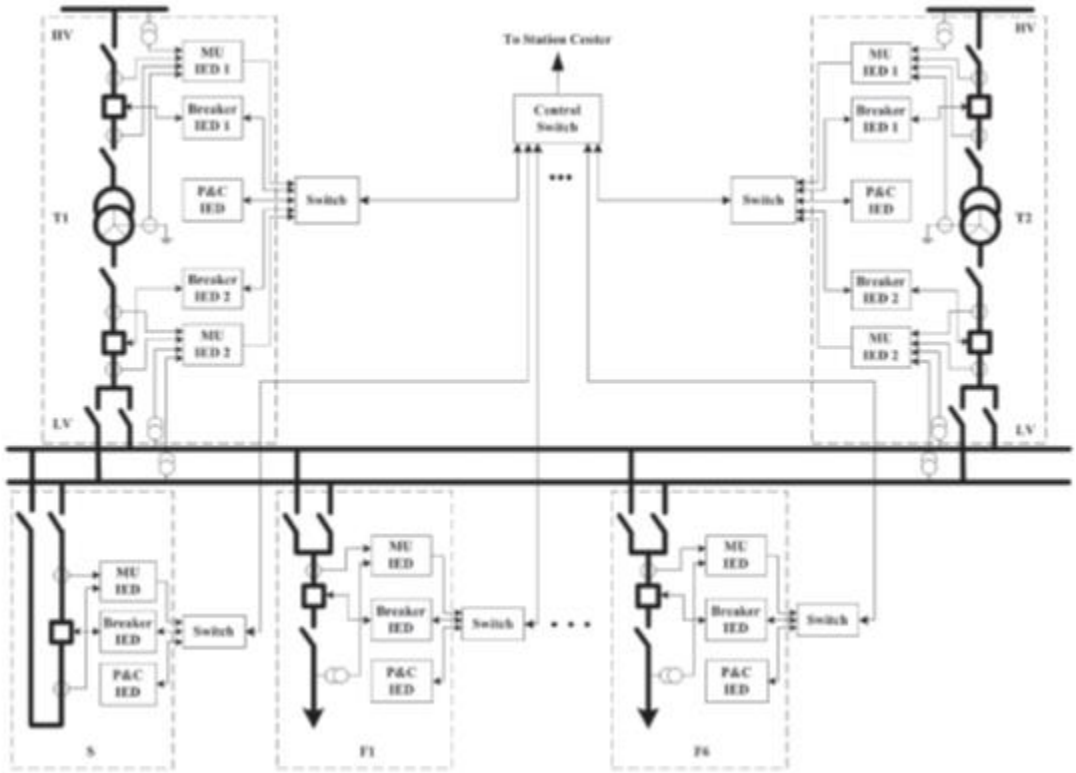


Figura 23. Esempio di sottostazione D2-1 con rete di comunicazione

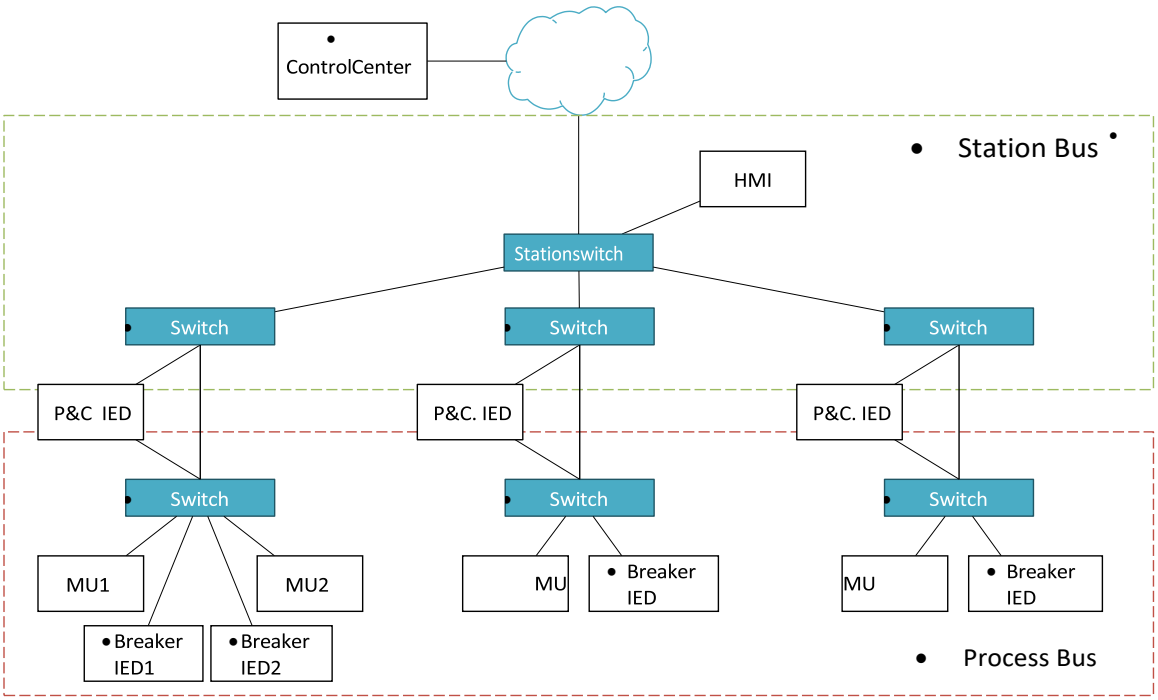


Figura 24. Esempio di rete di comunicazione per sottostazione semplificata D2-1 con stazione e bus di processo distinti

Lo standard IEC 61850 specifica un'architettura globale da utilizzare per la rete di comunicazione di una sottostazione. Divide la rete in due livelli: il **bus di stazione** e il **bus di processo** [49]. Non è necessario mantenerli come due reti isolate e fisicamente separate: come mostrato nella Figura 23, i due bus possono essere uno e lo stesso. Tuttavia, si consiglia di isolarli per evitare la congestione del traffico [49]. Un esempio in cui il bus di processo e il bus di stazione sono più distinti ma ancora collegati è mostrato nella Figura 24 (l'apparecchiatura primaria non è mostrata e viene mostrata solo una per ogni tipo di alloggiamento). Se collegare o meno i bus dipende dal contesto. Entrambi i bus possono utilizzare qualsiasi topologia di rete adatta, molte delle quali sono documentate in IEC 61850-90-4 [49].

Bus di stazione

Il bus di stazione collega gli switch al gateway per consentire le comunicazioni all'esterno della sottostazione. Trasporta molti tipi di traffico, in particolare gli scambi basati sul protocollo **TCP** (**Transmission Control Protocol**) come **MMS** (**Manufacturing Message Specification**), **FTP** (**File Transfer Protocol**) o **SNTP** (**Simple Network Time Protocol**) [49]. Può anche trasferire il traffico **GOOSE** [49]. A causa della natura del traffico che trasporta, il bus della stazione ha generalmente requisiti di prestazione meno rigidi rispetto al bus di processo.

Bus di processo

Il bus di processo viene utilizzato per collegare le **Merging Unit**(**MU**) e i relativi **IED** all'interno di un vano. Ha requisiti di prestazione molto più severi rispetto al bus di stazione poiché deve supportare le comunicazioni **SV** e **GOOSE**, entrambe critiche per la sottostazione [49]. Può anche trasportare **MMS** [49]. Il bus di processo può essere completamente isolato dal bus di stazione, se lo si desidera, il che significa che solo gli IED collegati a entrambi i bus possono essere raggiunti dal bus di stazione.

3.1.2.3 Modello informativo e interfaccia del servizio di comunicazione astratta

Per il funzionamento di una sottostazione IEC 61850 sono necessari diversi tipi di messaggi. Sono classificati in 6 tipi di messaggi [47],[49] elencati nella Tabella 3. La norma IEC 61850 definisce un modello di informazioni e servizi correlati per consentire l'uso di questi messaggi.

Tabella 3 – Tipi di messaggi defini in IEC 61850

Message type	Use
Type 1	Fast messages for protection functions
Type 1A	Trip messages, highest priority among Type 1
Type 2	Medium speed messages for automation functions
Type 3	Low speed messages for operator functions
Type 4	Raw data messages for continuous streams of data from IEDs
Type 5	File transfer functions
Type 6	Exchanges using access control for highest security

Il modello informativo utilizza un approccio orientato agli oggetti. Un IED contiene un dispositivo logico (**Logical Device - LD**) astratto che contiene diversi nodi logici (**Logical Node - LN**) che rappresentano ciascuno una funzione nell'IED gli esempi includono un CB, una misurazione o un guasto a terra transitorio [50]. I nomi dati agli LN sono definiti dallo standard e intendono riflettere la funzione di ogni LN, migliorando la leggibilità. La prima lettera nel nome del LN indica a quale dei 19 gruppi LN appartiene il LN. Ad esempio, il LN per un CB è denominato XCBR, con la lettera X che indica che questo LN fa parte del gruppo delle funzioni del quadro [50]. Le informazioni in ogni LN vengono ulteriormente suddivise in oggetti dati (**Data Object - DO**) e Attributi dei dati (**Data Attribute - DA**). Un XCBR detiene un DO denominato Pos (**la posizione**) che include il DA stVal per indicare se l'interruttore è aperto, chiuso o in qualche altro stato [50]. Gli LN e gli DO disponibili sono standardizzati in IEC 61850-7-2 [51] e IEC 61850-7-4 [52].L'**ACSI** (**Abstract Communication Service Interface**) fornisce una descrizione dei servizi utilizzati per accedere e agire sui dati trovati in vari LN. L'ACSI è astratto, come indica il nome, e deve essere mappato su un protocollo applicativo per essere utilizzato nella pratica. Questa operazione viene eseguita utilizzando un **SCSM** (**Specific Communication Service Mapping**) [50].

3.1.2.4 Protocolli applicativi e mappatura dei servizi di comunicazione specifici

Lo standard IEC 61850 fornisce **SCSM** per mappare il modello di informazioni e **ACSI** su tre protocolli: **GOOSE**, **SV** e **MMS** [50]. Ognuno ha il proprio ruolo e vengono utilizzati insieme ai protocolli esistenti per scopi quali la sincronizzazione temporale. Un riepilogo dei protocolli dell'applicazione è mostrato nella Figura 25. La quantità tipica di traffico che si produce in una sottostazione secondo la norma IEC 61850-90-4 [49] è mostrata nella Tabella 4. I messaggi codificati e inviati utilizzando qualsiasi protocollo applicativo che implementa i servizi ACSI sono denominati **Protocol Data Unit** (**PDU**) [50]. La parte di una PDU che trasporta i dati dell'applicazione è indicata come **APDU** (**Application Protocol Data Unit**) [53].

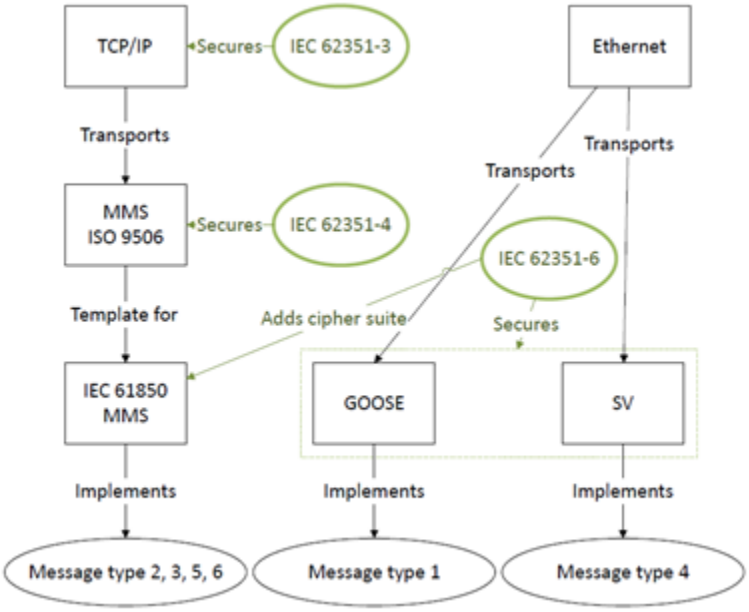


Figura 25. Protocolli applicativi utilizzati in IEC 61850 e relativi standard

Tabella 4 – Tipi di traffico nelle sottostazioni IEC 61850

Protocol	Message types	Affected buses	Approximate amount of traffic
GOOSE	1, 1A	Station and process	1 kbit/s (steady-state), 1 Mbit/s (bursts)
SV	4	Mostly process	6 Mbit/s per IED
MMS	2, 3, 5, 6	Station and process	10 kbit/s per IED
SNTP	Time synchronization	Mostly station	Unknown
PTP	Time synchronization	Mostly process	Unknown

3.1.2.5 Generic Object Oriented Substation Event (GOOSE)

I messaggi GOOSE fanno parte del modello di comunicazione Generic Substation Event (GSE) che mira a trasmettere rapidamente e in modo affidabile le informazioni sugli eventi che si verificano nella sottostazione [51].
Sono utilizzati in IEC 61850 per trasferire messaggi di tipo 1 e 1A, facendo riferimento ai messaggi "fast" e ai messaggi "trip" nella Tabella 3. Entrambi hanno la massima priorità. Le GOOSE PDU vengono create e inviate utilizzando un modello publisher-subscriber secondo condizioni pre-configurate che indicano eventi e contengono dati rilevanti sugli eventi in questione [49]. Sono utilizzati, ad esempio, per inviare segnali critici "trip" ai CB in caso di instabilità del sistema. Queste PDU hanno requisiti di prestazione molto rigidi a seconda delle informazioni che trasportano. Il requisito più rigoroso per trasmettere e ricevere un messaggio è di 4 ms [4].
Piuttosto che fare affidamento su TCP a livello di rete 3, come fa l'MMS, il protocollo GOOSE funziona a livello di rete 2. In effetti, è un'estensione di Ethernet, il che significa che gli IED nella sottostazione si indirizzano l'un l'altro utilizzando indirizzi **MAC** (**Media Access Control**). Sfrutta le funzioni multicast di Ethernet per consentire al publisher di inviare GOOSE PDU a tutti i suoi subscriber contemporaneamente [49].
Il traffico GOOSE viene rilevato sia a livello di processo che di sottostazione. Una particolarità delle GOOSE PDU è che vengono trasmesse costantemente. Anche quando non sono state apportate modifiche, le GOOSE PDU vengono ritrasmesse

a intervalli preconfigurati e portano le stesse informazioni. Ciò serve a tenere conto degli errori nella rete [49] e delle apparecchiature di nuova installazione che necessitano delle informazioni [51].
Questa ritrasmissione di PDU spiega la piccola quantità di traffico GOOSE visto nella Tabella 4 in stato stazionario. Quando si verifica un nuovo evento, la quantità di traffico GOOSE può aumentare notevolmente poiché il tempo tra le ritrasmissioni viene temporaneamente ridotto di molto rispetto a condizioni invariate, nel tentativo di garantire che gli abbonati ricevano almeno una delle GOOSE PDU in tempo [54].
I campi inclusi in ogni GOOSE PDU sono definiti nella IEC 61850-8-1[54] e i campi aggiuntivi sono definiti nella IEC 62351-6 [55]. Sono elencati nella Tabella 5. In sostanza, la GOOSE PDU rappresenta lo stato di un DO da comunicare agli abbonati, come lo stato di un CB. Il campo allData contiene i valori, mentre datSet chiarisce il significato dei valori contenendo il nome della DO in questione. I valori stNum (numero di stato) e sqNum (numero di sequenza) vengono utilizzati per distinguere i nuovi PDU (nuovi stati) da quelli ritrasmessi [54]. Le PDU che trasportano nuovi allData hanno uno stNum di (precedente stNum+1) e un sqNum di 0. Le PDU ritrasmesse hanno gli stessi allData e stNum delle PDU precedenti e hanno un sqNum di (precedente sqNum+1). In quanto tale, il destinatario delle PDU deve solo guardare stNum per scoprire se i valori del DO monitorato sono cambiati.

Tabella 5 – Campi nella PDU GOOSE

Field name	Data type	Description
APPID	Integer	Application identification (0x0000 to 0x3FFF for GOOSE type 1, or 0x8000 to 0xBFFF for GOOSE type 1A)
length	Integer	Length of this PDU, which is always (length of APDU + 8) bytes
gocbRef	String	GOOSE control block reference
TAL	Integer	Short for timeAllowedToLive, the maximum time (in milliseconds) the subscriber should wait for the next PDU
datSet	Integer	Data set, a reference to the specific DO and DAs being monitored and for which *allData* is applicable
goID	String	GOOSE identification, user-defined name for this message
t	Timestamp	Timestamp, the time of the last *stNum* change
stNum	Integer	State number, the number of times values in *allData* have changed
sqNum	Integer	Sequence number, the number of retransmissions since the last *stNum* change
simulation/test	Boolean	Test flag, if *true*, *allData* contains simulated values for testing purposes, and if *false*, it contains real data
confRev	Integer	Configuration revision, the number of configuration updates to the data set named by *datSet*
ndsCom	Boolean	Needs commissioning flag, if *true*, this association requires maintenance
numDatSetEntries	String	Number of values contained in *allData*
allData	List	Values of the DO and DAs in the data set named by *datSet*
AuthenticationValue	Octet String	RSA digital signature for integrity checking (IEC 62351-6 only)

3.1.2.6 Valori campionati (Sampled Value - SV)

Come mostrato nella Tabella 3 e nella Tabella 4 il protocollo SV viene utilizzato per comunicare messaggi di tipo 4, cioè valori grezzi di corrente e tensione in formato digitale, ad altri IED che li richiedono. Questi valori vengono inviati dalle MU che convertono i valori analogici ricevuti dalle apparecchiature da campo in SV PDU digitali e li trasferiscono a diversi ricevitori contemporaneamente utilizzando un modello publisher-subscriber [49]. In molti modi, il protocollo SV ricorda GOOSE. È anche un'estensione di Ethernet, il che significa che funziona a livello di rete 2 e utilizza indirizzi MAC per identificare editori e abbonati. Utilizza le funzioni multicast di Ethernet per trasferire continuamente PDU a molti ricevitori. Tuttavia, SV differisce da GOOSE per la quantità di traffico molto più significativa e costante, come si vede nella Tabella 4. Le SV PDU

vengono trasferite a velocità comprese tra le migliaia al secondo, determinando un carico più pesante sulla rete. La norma IEC 61850-90-4 stima che un massimo di 6 publisher SV possono essere collegati allo stesso bus a 100 Mbit/s [49].
I campi per SV PDU sono definiti da IEC 61850-7-2 [51], IEC 61850-9-2 [53] e i campi aggiuntivi sono definiti in IEC 62351-6 [55]. Ogni SV PDU contiene una APDU che a sua volta può trasportare diverse **ASDU** (**Application Service Data Unit**). Notare che le SV ASDU generalmente non portano tutti i campi a causa di problemi di prestazioni: si consiglia di ridurre al minimo la dimensione totale dell'APDU utilizzando nomi molto piccoli per **svID** e non utilizzando campi opzionali [49].
Proprio come una PDU GOOSE, una PDU SV rappresenta i valori di un DO a cui fa riferimento **datSet**. In questo caso, i valori del DO si trovano nel campo sample. Inoltre, ogni SV ASDU porta un valore **smpCnt** (conteggio campioni) per identificarlo. Questo contatore tiene traccia dell'ordine delle SV ASDU. Il suo valore è (**smpCnt precedente + 1**). Poiché **smpCnt** è solo un valore a 16 bit, non può superare 2 ^ 16 -1 (65.535), un valore che normalmente sarebbe raggiunto entro alcuni secondi data la velocità tipica di SV PDU in una sottostazione. Questo di solito non accade poiché **smpCnt** si azzera a ogni impulso di sincronizzazione [53], che si verifica quasi ogni secondo [56]. Lo **smpCnt** nel traffico tipico è quindi in costante aumento linearmente, con azzeramenti periodici a 0.

3.1.2.7 Manufacturing Message Specification (MMS) e MMS IEC 61850

L'MMS, come definito nel proprio standard denominato ISO 9506 [57], viene utilizzato in diverse applicazioni dei sistemi di controllo. È un protocollo di comunicazione generico che fornisce un'ampia varietà di funzionalità per modellare e scambiare dati tra client e server. Gli sviluppatori possono usarlo come modello per implementare nuovi protocolli applicativi e abilitare le comunicazioni con dispositivi come robot, IED o unità terminali remote (RTU) [58]. Ad esempio, i derivati esistenti dell'MMS vengono utilizzati per molte applicazioni, tra cui la gestione dell'inventario, la movimentazione dei materiali e la gestione dell'alimentazione [58]. Qualsiasi protocollo che utilizza MMS utilizza un modello denominato **Virtual Manufacturing Device** (**VMD**) che specifica gli oggetti, i servizi e il comportamento del protocollo dell'applicazione [58]. Utilizza un approccio orientato agli oggetti in cui client e server MMS mantengono oggetti che rappresentano il loro stato e forniscono servizi ad altre parti per agire su questi oggetti [59]. Un esempio di un servizio generico offerto da MMS è il servizio Get per leggere il valore di un oggetto [59]. I possibili attributi degli oggetti includono variabili, log, file e altro [54].
Nel contesto della IEC 61850, il protocollo MMS viene utilizzato per implementare le comunicazioni tra IED nella stazione o nel bus di processo e nei gateway o nei client esterni, come lo SCADA [49]. Questa specifica implementazione del protocollo è spesso indicata come IEC 61850 MMS [60]. L'MMS è stato scelto per implementare questo tipo di comunicazioni, poiché fornisce molti oggetti e servizi che si adattano bene al modello di dati astratto richiesto dalla IEC 61850 parti da 7-2 a 7-4 [54]. Come mostrato nella Tabella 4, l'MMS IEC 61850 viene utilizzato per comunicare qualsiasi messaggio di un tipo non specificamente coperto dagli altri protocolli. È quindi più interessato alle comunicazioni che non sono critiche in termini di tempo come le altre, come il trasferimento di file, le azioni dell'operatore, ecc. Si basa sul protocollo TCP per garantire la consegna delle PDU [54]. L'MMS IEC 61850 richiede solo un sottoinsieme di tutte le funzionalità fornite dal protocollo MMS generico [54]. In questo report, distinguiamo i termini "MMS" da "IEC 61850 MMS" quando necessario.

3.1.2.8 Time Synchronization

La sincronizzazione temporale è fondamentale per il corretto funzionamento della sottostazione. A tal fine, la IEC 61850 raccomanda l'uso di SNTP sul bus della stazione e di **Precision Time Protocol** (**PTP**) sul bus di processo [49]. SNTP funziona su **UDP** (**User DatagramProtocol**) utilizzando un modello client-server per fornire la sincronizzazione dell'ora. PTP viene utilizzato sul bus di processo e invece trasmette l'ora utilizzando il multicast di livello 2. PTP è raccomandato in quanto è considerato più accurato di SNTP [49], che è necessario soprattutto per le MU che utilizzano il protocollo SV [56].

3.1.2.9 Modello Informativo

IEC 61850 offre potenti capacità di modellazione delle informazioni. In IEC 61850, le informazioni possono essere scambiate usando ACSI (insieme a una mappatura tecnologica). Il file IED Capability Description (ICD) che viene fornito con i dispositivi

di protezione di una specifica azienda, passerà attraverso lo strumento di configurazione e verrà convertito in file System Configuration Description (SCD). viene fatto perché il file ICD può essere riconosciuto solo con un dispositivo specifico di un marchio specifico ma una volta convertito in un file SCD e caricato negli IED, i diversi IED sono in grado di comunicare tra loro. L'ACSI si concentra sullo scambio di dati concreti. Il Meta modello di IEC 61850 che costituisce la base per i modelli di informazioni IEC 61850 è riassunto in Figura 26, che contiene solo le parti accessibili tramite ACSI.

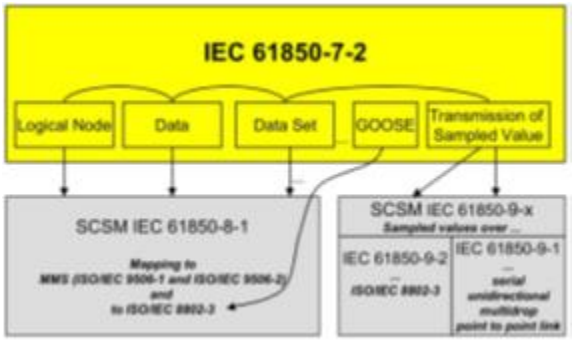


Figura 26. ASCI mapping (concettuale)

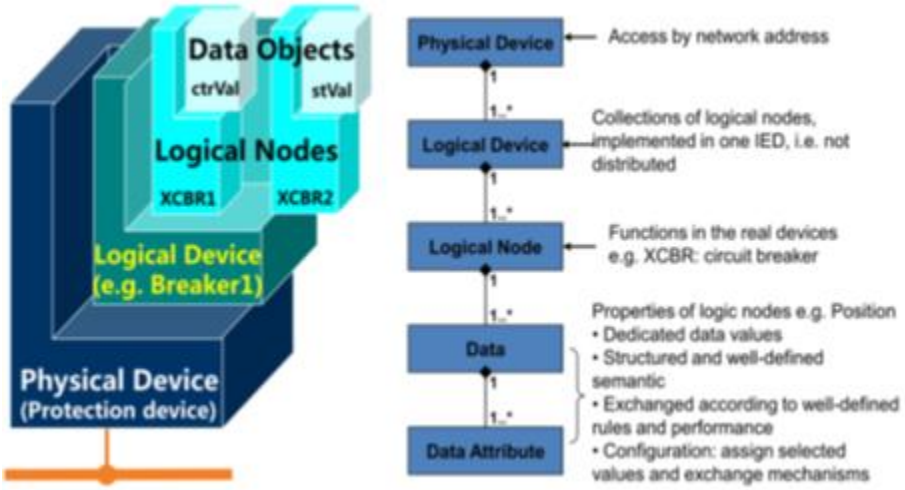


Figura 27. Architettura dei dati nel modello IEC 61850

Il Meta modello contiene costrutti che rappresentano il modello informativo (valore misurato, set point, sequenza di eventi) e costrutti che rappresentano la configurazione della comunicazione (chiamato modello di scambio di informazioni in IEC 61850-7-1), come la configurazione di un blocco di controllo (report con buffer) utilizzato da un client o dalle applicazioni associate. Come mostrato, ogni sorgente di dati in IEC 61850 inizia con un server che ha file e dispositivi logici.
Ogni dispositivo fisico (Physical Device (PD)) può rappresentare molti dispositivi logici. Un server deve avere almeno un dispositivo ma può averne molti, ad esempio, quando rappresenta un'intera sottostazione. Ogni dispositivo logico è composto da molti nodi logici (LN), con almeno un nodo logico 0 contenente alcune informazioni di carattere generale come la targhetta del dispositivo logico.
Il nodo logico si basa su una classe di nodi logici. Tuttavia, alla classe del nodo logico viene fatto riferimento solo per nome. Le classi di nodi logici sono definite negli standard del modello informativo per IEC 61850 come 7-4, 7-410 e 7-420. Le classi di nodi logici definiscono la semantica e possono essere molto generiche come GGIO che rappresenta solo Input/Output (IO) generici o più specifiche come XCBR che rappresenta le capacità di un interruttore. Gli LN contengono diversi costrutti

del modello di scambio di informazioni e in aggiunta almeno un oggetto dati. L'oggetto dati consentito di un nodo logico e se sono obbligatori, facoltativi o vincolati sono definiti dalla classe del nodo logico.
Gli oggetti dati sono definiti dalle classi di dati comuni, definite in 7-3 per le classi di dati comuni generali applicabili o nelle parti che definiscono le classi di nodi logici se è necessario definire una classe di dati nuova o molto specifica. Come la classe di nodo logico, le classi di dati comuni sono referenziate solo per nome.
Classi di dati comuni sono definite per varie cose come lo stato intero, valori misurati o impostazioni analogiche. Non definiscono una semantica molto forte; questo viene fatto per gli oggetti dati nel contesto dei loro nodi logici definiti dalle loro classi di nodi logici. Ad esempio, l'oggetto dati OpCnt della classe di nodo logico XCBR (Circuit Breaker) rappresenta il conteggio delle operazioni dell'interruttore. OpCnt utilizza lo stato intero della classe oggetto dati, vedi Figura 27.
Per semplicità, il nome dell'oggetto dati in realtà rappresenta la semantica e quindi OpCnt viene utilizzato da diverse classi di nodi logici che richiedono un conteggio delle operazioni. Gli oggetti dati contengono infine attributi di dati e possono contenere altri oggetti dati.
Gli attributi dei dati hanno un tipo che può essere semplice come un numero intero o un valore booleano, ma può anche essere complesso e creato da tipi di dati semplici.
Nell'ultima colonna viene indicato se i dati sono M (Mandatory - obbligatorio) o O (Optional - facoltativo).

3.1.2.10 Tipi di nodi logici

Esistono nodi logici per il controllo automatico i cui nomi iniziano tutti con la lettera "A". Esistono nodi logici per la misurazione e la misurazione i cui nomi iniziano tutti con la lettera "M". Allo stesso modo ci sono nodi logici per:

- Controllo di Supervisione
- Funzioni Generiche
- Interfaccia / Archiviazione (I)
- Nodi logici di sistema (L)
- Protezione (P)
- Relativo alla protezione (R)
- Sensori (S)
- Trasformatori di strumenti (T)
- Quadro (X)
- Trasformatori di potenza (Y)
- Altre attrezzature (Z)

Tabella 6 - Anatomia del nodo logico dell'interruttore in IEC 61850-07-4

XCBR class				
Data object name	**Common data class**	**Explanation**	**T**	**M/O/C**
LNName		The name shall be composed of the class name, the LN-Prefix and LN-Instance-ID according to IEC 61850-7-2, Clause 22.		
Data objects				
Descriptions				
EEName	DPL	External equipment name plate		O
Status information				
EEHealth	ENS	External equipment health		O
LocKey	SPS	Local or remote key (local means without substation automation communication, hardwired direct control)		O
Loc	SPS	Local control behaviour		M
OpCnt	INS	Operation counter		M
CBOpCap	ENS	Circuit breaker operating capability		O
POWCap	ENS	Point on wave switching capability		O
MaxOpCap	INS	Circuit breaker operating capability when fully charged		O
Dsc	SPS	Discrepancy		O
Measured and metered values				
SumSwARs	BCR	Sum of switched amperes, resettable		O
Controls				
LocSta	SPC	Switching authority at station level		O
Pos	DPC	Switch position		M
BlkOpn	SPC	Block opening		M
BlkCls	SPC	Block closing		M
ChaMotEna	SPC	Charger motor enabled		O
Settings				
CBTmms	ING	Closing time of breaker		O

Ogni nodo logico ha un ID di istanza LN come suffisso al nome del nodo logico. Ad esempio, supponiamo che ci siano due ingressi di misurazione in un dispositivo per misurare due alimentatori trifase. Il nome standard del nodo logico per un'unità di misura per alimentazione trifase è MMXU. Per delineare tra le misurazioni per questi 2 alimentatori verranno utilizzati i nomi dei nodi logici IEC61850 di MMXU1 e MMXU2. Ciascun nodo logico può anche utilizzare un prefisso LN specifico dell'applicazione opzionale per fornire un'ulteriore identificazione dello scopo di un nodo logico, vedi Figura 28.

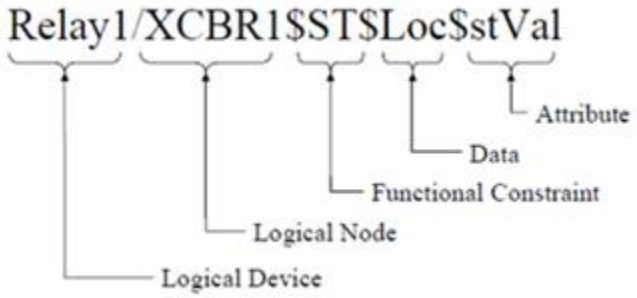


Figura 28. Anatomia di un nome oggetto IEC 61850-8-1

3.1.2.11 Comunicazione in IEC 61850

Lo IEC 61850 fornisce diversi mapping tecnologici per la comunicazione, vedi Figura 29. La mappatura su connessione seriale non avrà più un ruolo importante, pertanto viene considerata solo la mappatura su Ethernet. I valori campionati (SV) forniscono una comunicazione ciclica molto veloce, come 4000 campioni al secondo [61],[62]. Poiché IEC 61850 si basa su Ethernet senza alcun supporto specifico per il comportamento in tempo reale, non supporta il comportamento deterministico in tempo reale ma può essere considerato molto veloce e fornisce dati relativi in tempo reale con un'elevata precisione tra le diverse origini dati [63]. I valori campionati (Sampled Value (SV)) possono essere forniti da multicast, consentendo a più ricevitori di accedere agli stessi dati inseriti una sola volta sul cavo o inviando unicast i dati a un solo ricevitore.

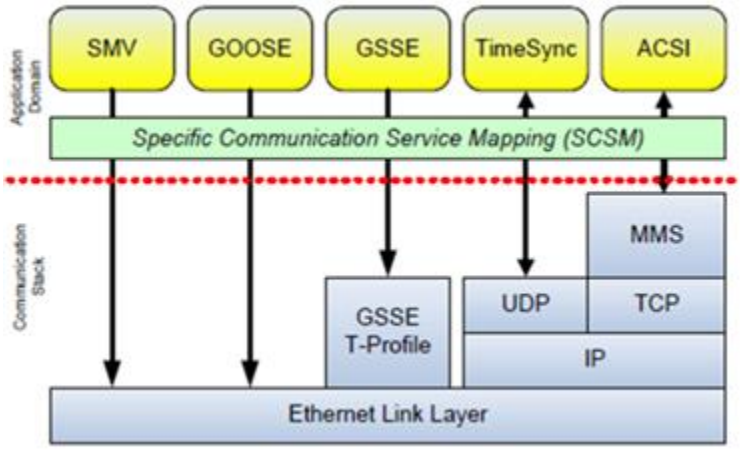


Figura 29. Mapping della comunicazione nello IEC 61850

Il messaggio GOOSE è un meccanismo di comunicazione molto veloce e viene inviato quando si verifica un evento come diversi tipi di guasti.
MMS utilizza un modello client-server Inoltre, supporta le funzionalità di reporting (con buffer e senza buffer) e l'accesso ai registri contenenti dati storici.
Quando si fa riferimento a un oggetto DataObject, vengono inclusi tutti i relativi attributi di dati. Pertanto, i set di dati raggruppano i dati in nuove categorie e i dati possono essere contenuti in diversi set di dati. I set di dati possono essere preconfigurati tramite Substation Configuration Language (SCL) o definiti dinamicamente tramite un servizio MMS. Utilizzando MMS, i set di dati possono essere utilizzati per leggere e scrivere dati. Tuttavia, il ruolo più importante dei set di dati consiste nell'utilizzarli come origine dei dati per diversi meccanismi di segnalazione.
Un blocco di controllo del registro 'log' utilizza i dati di un set per memorizzare la cronologia dei dati in un registro 'log'. I client possono eseguire query sul registro tramite MMS. Utilizzando blocchi di controllo dei report, con buffer o senza buffer, i server inviano i dati a un client. Anche qui MMS è usato come protocollo.
La differenza tra bufferizzato e non bufferizzato è che nel primo caso il server memorizza nel buffer i dati per il client preconfigurato in modo che il client possa ricevere i dati in un secondo momento, anche se non è connesso quando i dati devono essere inviati. Tuttavia, nel caso senza buffer qualsiasi client può iscriversi ai report in modo dinamico, ma le modifiche che si verificano mentre il client non è connesso andranno perse. I blocchi di controllo co buffer e senza buffer fanno riferimento a un set di dati per definire quali dati riportano.
Un set di dati può essere utilizzato da diversi blocchi di controllo. La configurazione del blocco di controllo del report specifica quando deve essere attivato un report e può essere impostato per modifiche di qualità, modifiche dei dati, aggiornamenti dei dati, integrità o interrogazioni generale. Gli ultimi due vengono attivati all'esterno dei dati per ricevere di volta in volta lo stato corrente (integrità) o richiedere lo stato dal client (interrogazione generale), ad esempio, al momento della riconnessione.

Quando le modifiche attivano il report, vengono restituiti solo i valori modificati di un set di dati. Pertanto, fa la differenza se un set di dati contiene un oggetto dati o tutti gli attributi di un set di dati singolarmente. Nel primo caso vengono restituiti tutti gli attributi se un valore cambia, mentre nel secondo caso viene restituito solo quello modificato. Poiché gli attributi possono rappresentare, ad esempio, valore, qualità e timestamp, è importante selezionare l'oggetto dati se devono essere ricevuti anche la qualità e il timestamp.
Un blocco di controllo GOOSE specifica a quali set di dati dovrebbe riferirsi un messaggio GOOSE. Può essere abilitato o disabilitato. Quando abilitato, invia tutti i dati del set di dati di riferimento utilizzando la comunicazione GOOSE basata su Ethernet.
I blocchi di controllo di esempio unicast e multicast definiscono quali dati devono essere inviati dal server in modo periodico (comunicazione ciclica). Quando un blocco di controllo fa riferimento a un set di dati, tutti i dati verranno inviati anche se non sono stati modificati. La frequenza di campionamento definisce la frequenza con cui i dati devono essere inviati. Nel caso di unicast, il blocco di controllo contiene anche informazioni sul client a cui inviare i dati. Oltre al suddetto accesso ai dati, la mappatura MMS supporta anche le capacità di navigazione per accedere alla struttura dei dati (dispositivi logici, nodi logici, oggetti dati, ecc.) E alla configurazione (set di dati, blocchi di controllo, etc.) e leggere e scrivere file nel server IEC 61850.

3.1.2.12 Architecture of the communication system

In passato veniva utilizzato il cablaggio in rame per collegare gli interruttori di circuito, i sezionatori, i trasformatori (Current Transducer (CT), Voltage Transducer (VT)) [64] alla sala di controllo e quindi la manutenzione e i costi erano molto elevati. Oggi, la tecnologia wireless e la fibra ottica sono molto popolari e comuni e invece di mazzi di cavi in rame, viene utilizzato un cavo in fibra ottica. Oggi vengono utilizzati trasformatori di strumenti non convenzionali (NCIT) come sensori di tensione capacitivi e trasformatori di corrente ottici che sono anche più sicuri. La trasmissione digitale di tensione e corrente riduce anche la quantità di cablaggio richiesta. L'uso diffuso di NCIT è stato messo in discussione poiché non esisteva un'interfaccia standard per l'interoperabilità multi-vendor. Questo sta cambiando con la crescente implementazione dei valori campionati IEC 61850-9-2 per l'interfaccia digitale tra NCIT e relè di protezione.
Le "interfacce" sono definite nello standard IEC 61850 per collegare i livelli di processo, alloggiamento e stazione di una sottostazione. La modellizzazione delle informazioni definisce i servizi, gli oggetti di dati, gli attributi che consentono lo scambio rapido delle informazioni. In totale ci sono 11 interfacce tra i diversi livelli di una sottostazione: questi sono mostrati in Figura 30. Ad esempio, l'interfaccia IF4 e IF5 sono riassunti di seguito. IF4 e IF5 insieme sono considerati essere bus di processo.

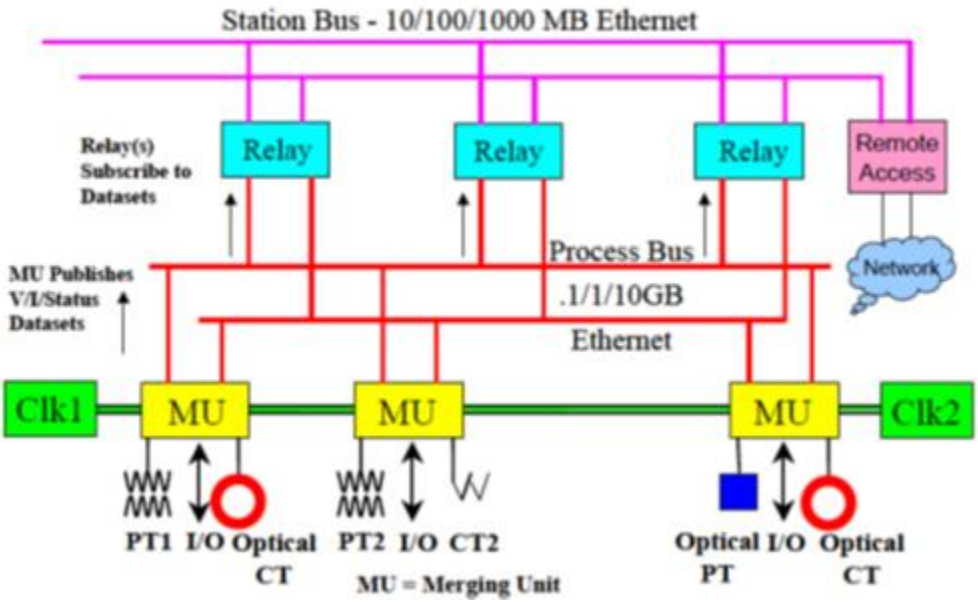


Figura 30. Modello di sottostazione dello standard IEC 61850

I dispositivi di automazione delle sottostazioni conformi a IEC 61850 sono interoperabili ma non necessariamente intercambiabili mediante l'uso di nodi logici e interfacce comuni [65]. L'infrastruttura di comunicazione in una sottostazione è composta da 3 livelli, ovvero livello del bus di processo, livello di alloggiamento e livello di stazione. Di seguito ognuno di essi è descritto in dettaglio [66].

L'architettura e la topologia delle reti process bus sono un'area di ricerca attiva. Le topologie Ethernet del process bus con valori di campionamento possono essere classificate come collegamenti punto a punto o reti commutate. La topologia punto a punto replica il layout del cablaggio secondario CT e VT analogico, sebbene con una rete in fibra ottica, ed è l'approccio adottato dal sistema "HardFiber" di General Electric. La capacità di rete non è un problema in un process bus punto a punto poiché ogni collegamento è limitato a due dispositivi.

Alcuni dispositivi process bus, che possono essere utilizzati in una rete commutata, vengono utilizzati nelle configurazioni punto a punto per soddisfare esigenze specifiche. Un esempio è la sottostazione Loganlea di Powerlink Queensland, in cui vengono utilizzati collegamenti punto-punto multipli per evitare la necessità di una sincronizzazione temporale centralizzata.

La “comunicazione orizzontale” nelle sottostazioni è la comunicazione tra i relè di protezione (inter-sgancio) o all'interno di un alloggiamento (connessioni di processo). La “comunicazione verticale” è il controllo delle apparecchiature della sottostazione attraverso un'interfaccia operatore locale o da un centro di controllo remoto. La comunicazione orizzontale è generalmente più sensibile al tempo e utilizza un modello publisher-subscriber, mentre la comunicazione verticale è più focalizzata sull'affidabilità e un modello client-server è più comunemente usato.
Per comprendere le comunicazioni tra i dispositivi in una griglia, la rete è divisa in alcune parti per semplicità. Di seguito, in Figura 31, vengono presentate e spiegate queste sezioni.

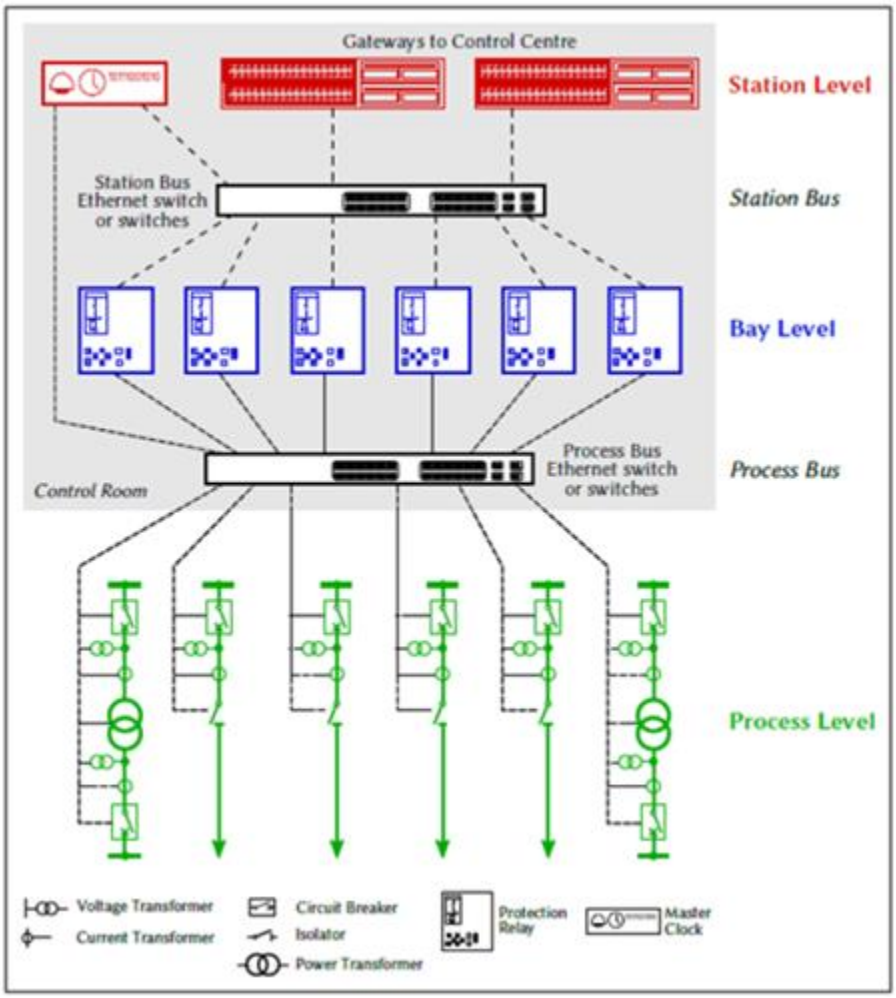


Figura 31. Comunicazione tra livello processo, livello stazione e livello alloggiamento

Livello Processo

Consiste in tutte le funzioni che interagiscono con il processo e sono fondamentalmente IO binari e analogici come l'acquisizione dei dati (come il campionamento) e l'emissione di comandi. Gli strumenti di questo livello sono trasformatore di corrente, trasformatore di tensione, trasformatore di potenza, isolatore e interruttore automatico. Sono in realtà l'apparato primario della rete elettrica che attraverso le interfacce e lo switch Ethernet del bus di processo inviano i dati ai relativi IED a livello di baia. Le funzioni a livello di processo sono per lo più implementate negli IED a livello di alloggiamento, altrimenti queste funzioni sono incluse negli IED a livello di processo.

Bus di processo

Il bus di processo è principalmente uno switch Ethernet che attraverso di esso gli stati e gli SV prelevati dall'apparecchiatura a livello di processo vengono inviati agli IED a livello di baia e in base alla configurazione software degli IED vengono emessi i comandi che passano nuovamente attraverso il bus di processo alle apparecchiature a livello di processo per implementare la funzione richiesta. La capacità del bus di processo è ancora un problema aperto e molte ricerche sono state fatte al riguardo.

Livello Alloggiamento
Consiste nelle funzioni all'interno di un alloggiamento. Un alloggiamento è una sotto parte di una sottostazione come un quadro o un alimentatore di linea. Gli IED a livello alloggiamento rispetto alla configurazione del software al loro interno, decidono in merito alle azioni da eseguire quando si ricevono i dati dall'apparecchiatura a livello di processo.

Bus di stazione
Lo switch Ethernet che riceve i dati dal livello di processo e IED a livello di baia e li invia a un centro di controllo remoto come SCADA o altri programmi di supervisione tramite Gateway e riceve i comandi dall'operatore in Control Center per configurare la rete nel modo in cui è obbligatorio.

Livello Stazione
Consiste nella parte di monitoraggio della rete e invia anche comandi agli IED per la configurazione di rete desiderata.

Gateway
Sono dispositivi che hanno la funzione di modificare i protocolli per consentire la comunicazione tra i dispositivi. Comunemente, i sistemi SCADA utilizzano il protocollo DNP3 e gli IED in base al livello o al livello di processo hanno lo standard IEC 61850. Per comprendere per SCADA e IED la struttura delle informazioni che comunicano, è necessario modificare il protocollo. Tuttavia, c'è uno sforzo per cambiare il protocollo di SCADA in IEC 61850.

3.2 Applicazione dello standard nel contesto delle smart-grids

I sistemi energetici non possono lavorare efficientemente senza un sistema di scambio di informazioni. Il mercato dell'energia dipende dalle informazioni di domanda e offerta. I produttori di energia hanno bisogno di monitorare e operare efficacemente sulle proprie facilities. C'è bisogno di uno scambio di informazioni a differenti livelli e che queste informazioni di scambio siano adeguatamente sicure. Le tecnologie sono pronte e gli standard tecnici sono già presenti in versione draft, così ora è tempo per le aziende ICT di integrazioni di sistemi di fornire soluzioni adeguate.
A seconda del tipo di utilizzo dello standard IEC 61850, ovvero: essere un operatore all'interno della struttura, un operatore esterno alla struttura o un integratore di sistema con la configurazione dei prodotti IEC 61850, possono esserci diversi modi di utilizzo, vedi Figura 32.

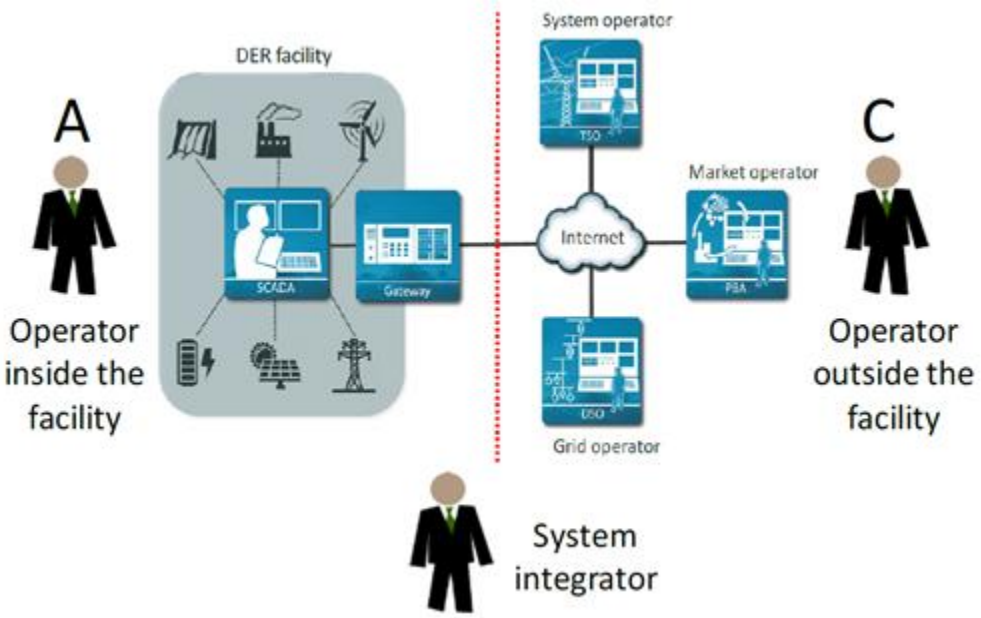


Figura 32. Attori presenti nel sistema

3.2.1 Operatore all'interno della struttura DER

Come proprietario e gestore di una struttura DER, l'attenzione sarà sempre rivolta alla conservazione delle risorse e all'ottenimento di una produzione ottimale e, in secondo luogo, alle interazioni con gli operatori esterni alla struttura.
Tuttavia, essere connessi al sistema di alimentazione oggi richiede sempre più attenzione ad avere uno stretto coordinamento e interazione tra la struttura DER e gli attori del sistema di alimentazione, a beneficio dei servizi ausiliari e dei servizi del mercato dell'energia.
Da un punto di vista della comunicazione dei dati, la funzione DER dovrebbe concentrarsi sui seguenti elementi:
1. Garantire l'accesso condiviso alle informazioni gestite dalla struttura DER;
2. Strumento DER in qualità di creatore dell'origine dati e proprietario di dati non aggregati;
3. Il punto di comunicazione (interfaccia PCOM) dovrebbe essere basato su standard internazionali e aperti.

3.2.2 Integratore di sistema

Come integratore di sistema, l'obiettivo sarebbe quello di avere un buon business basato sulla struttura DER e questo implica anche fornire il miglior servizio tecnico.
Laddove questa specifica si concentra sull'interfaccia PCOM (Point of Communication) esterna (è l'interfaccia tra la struttura DER e qualsiasi attore esterno alla struttura DER, in termini di comunicazione dei dati e scambio di informazioni.), è possibile utilizzare IEC 61850 anche in altri processi di integrazione del sistema interno. Si noti inoltre che le interfacce e gli elementi di comunicazione IEC 61850, compresa la sicurezza delle informazioni, dovrebbero essere basati su standard internazionali. Ciò ridurrà i costi per la struttura DER e andrà anche a beneficio dell'integratore di sistemi, poiché la formazione e il reclutamento del personale, la manutenzione di soluzioni proprietarie e la riduzione dei costi dei componenti, possono alla fine beneficiare delle entrate aziendali.
Dal punto di vista della comunicazione dei dati, l'integratore di sistemi dovrebbe concentrarsi sui seguenti elementi:

1. Sugli strumenti ICT che supportano il processo di integrazione del sistema;
2. Supportare gli standard internazionali e ridurre i costi di manutenzione delle soluzioni proprietarie;
3. Vedere i servizi di sicurezza delle informazioni come parte obbligatoria della tua attività.

3.2.3 Operatore esterno alla struttura DER

In qualità di operatore esterno alla struttura DER, indipendentemente dal fatto che tu sia un operatore di sistema, un operatore di mercato o un operatore di rete, il principale punto di interesse è se la struttura DER è una risorsa attendibile, sia dal punto di vista delle risorse DER che dal punto di vista della sicurezza.

- L'operatore di sistema si concentrerà su "Sicurezza dell'approvvigionamento";
- L'operatore del mercato si concentrerà su come utilizzare la funzione DER a condizioni di mercato;
- L'aggregatore si concentrerà su quanto sia affidabile e controllabile la struttura DER;
- L'operatore Grid si concentrerà su come utilizzare la funzione DER in caso di gestione della qualità dell'energia.

Dal punto di vista della comunicazione dei dati, gli operatori dovrebbero concentrarsi sui seguenti elementi:

1. L'interfaccia con una struttura DER dovrebbe avere un accesso sicuro e condiviso;
2. L'operatore dovrebbe essere in grado di comunicare con tutte le strutture DER, utilizzando la stessa interfaccia standard;
3. La sicurezza end-to-end dovrebbe essere obbligatoria, sulla base di un quadro di fiducia comune.

3.2.4 Architettura di Riferimento

Una architettura di riferimento è una descrizione concettuale che descrive i principali attori, componenti e le loro generiche interconnessioni, vedi Figura 33.

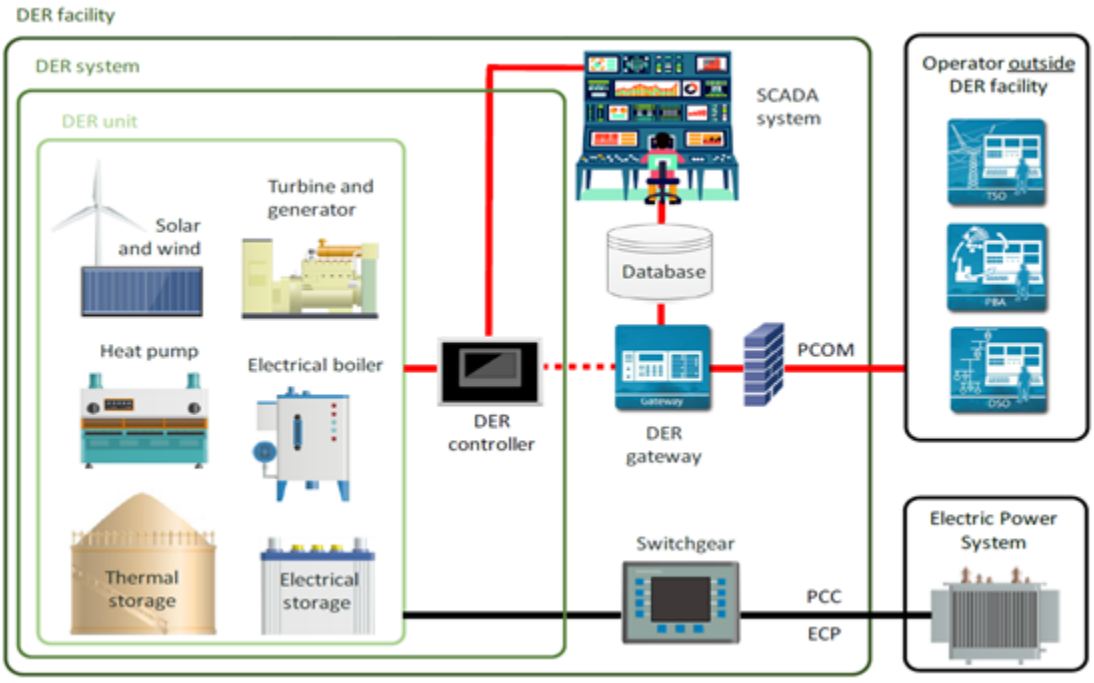


Figura 33. Architettura di riferimento DER

La linea rossa rappresenta la comunicazione dei dati tra le unità DER, controller / gateway DER, SCADA e apparecchiature di rete, all'interno della struttura DER. Di seguito la nomenclatura usata:
Struttura DER: è il termine usato per l'intera struttura, che ha un'interfaccia di comunicazione dati chiamata PCOM
Sistema DER: è il termine usato per una funzionalità che combina più unità DER in un sistema (ad esempio diversi gruppi motore-generatore, array FV, accumulo elettrico o turbine eoliche)
Unità DER: è il termine usato per il singolo DER (ad es. Turbina a gas, pompa di calore, caldaia elettrica, gruppo motore-generatore)
Gateway DER: è il componente fisico che ha una funzionalità server IEC e può comunicare con i client IEC al di fuori della struttura DER.
Controller DER: è un componente fisico o virtuale con funzionalità che controlla e aggrega diverse unità DER per un sistema DER.
PCOM: è l'interfaccia tra la struttura DER e qualsiasi attore esterno alla struttura DER, in termini di comunicazione dei dati e scambio di informazioni.
PCC: è il "Punto di accoppiamento comune" in cui la struttura DER è elettricamente collegata alla rete pubblica di fornitura di elettricità.
ECP: è il "Punto di collegamento elettrico" in cui ciascuna unità DER è elettricamente collegata alla rete elettrica della struttura locale; gruppi di unità DER (un sistema DER) hanno un ECP, dove si collegano alla rete elettrica della struttura DER; L'ECP per la funzione DER è identica al PCC.

3.2.5 Caso d'uso base per lo scambio di informazioni

La Figura 34 mostra sette differenti casi d'uso con lo scambio di informazioni tra 4 diversi attori (operatore di sistema, operatore di mercato, aggregatore e operatore grid) e una struttura DER.

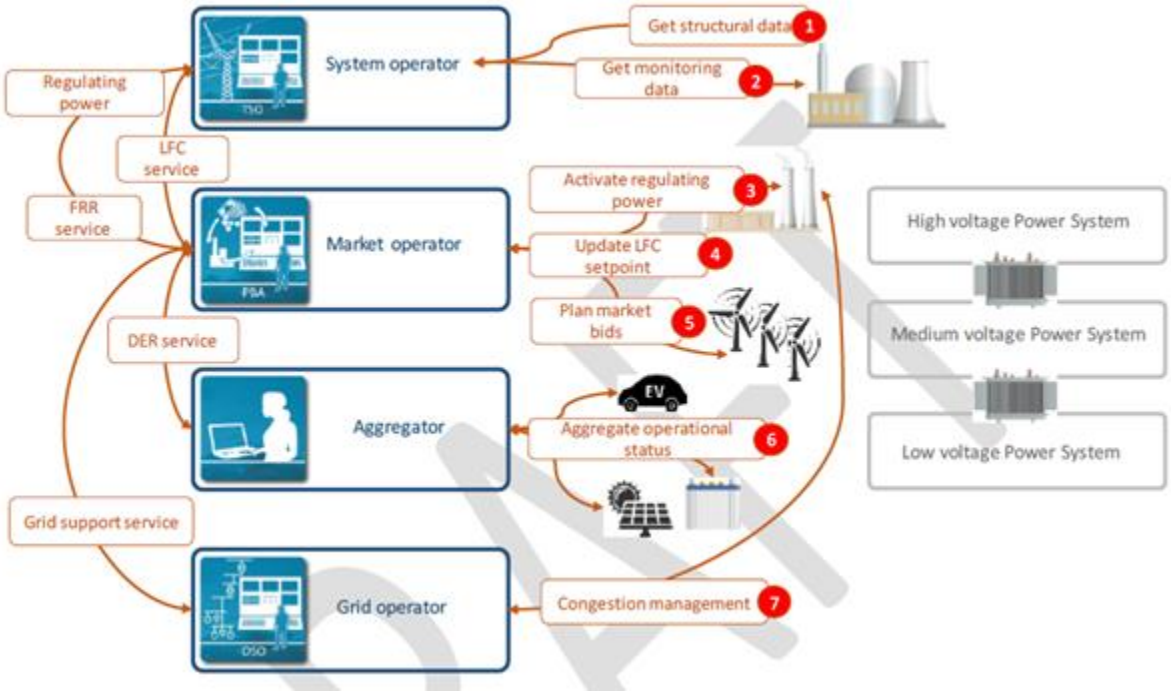


Figura 34. Esempio di caso d'uso

3.3 Tecnologie supportanti lo standard

Le principali tecnologie di comunicazione di ausilio allo standard IEC-61850, vedi Figura 35, sono:

- Power Line Carrier (PLC);
- Internet;
- WiFi;
- WiMAX.

Figura 35. Principali tecnologie supportanti lo standard

PLC Utilizza come mezzo fisico le linee di potenza già presentisovrapponendovi segnali a frequenze elevate (kHz). Se da una parte non richiede la realizzazione di nuove strutture un limite è dovuto alla presenza di disturbi che possono corrompere le informazioni e alla continuità del servizio che non è più garantita in caso di guasti o sezionamenti nella rete.

Internet è il mezzo di comunicazione più diffuso e accessibile. Non richiede installazione di una effettiva rete fisica ma solo del collegamento alla rete telefonica. Possibili limiti sono legati alla necessità di garantire la sicurezza dei dati e dei processi, e ai tempi di risposta richiesti da applicazione real time.

Wi-Fi e WiMAX sono protocolli di comunicazione relativi ai livelli ISO/OSI 1 e 2 che utilizzano, come mezzo fisico, onde radio (wireless).
WiFi Basato sullo standard 802.11:

- Utilizza bande di frequenza libere;
- Ha una copertura che varia da alcune decine di metri a qualche centinaia;
- La frequenza di trasmissione è di 2; 4 o 5GHz;
- La velocità di trasferimento dati è di 54 _ 125Mbps;
- L'accesso a mezzo fisico è di tipo CSMA/CA (Carrier Sense;
- Multiple Access with Collision Avoidance) e il canale wireless è quindi half-duplex.

WiMAX Basato sullo standard WiMAN 802.16:

- Utilizza sia le bande libere che quelle licenziate;
- Ha una copertura dell'ordine dei chilometri;

- La frequenza di trasmissione è di 2 _ 11GHz per utenza fissa e 2 - 6GHz per utenza mobile;
- La velocità di trasferimento dati è di 75Mbps.

Benefici e limiti.
WiMAX presenta un set di frequenze più ampio per cui l'uso delle bande licenziate è indicato per coprire aree dense e competitive, dove l'interferenza rappresenta un problema importante, mentre le bande libere sono indicate per coprire aree ristrette, per limitare interferenze e costi iniziali.
WiMAX rispetto a Wi-Fi risulta superiore sotto due aspetti: la velocità di trasmissione e il range di copertura delle celle, per cui è adatto a una trasmissione sia di tipo urbano che rurale (last mile).
WiMAX presenta tempi di latenza bassi dell'ordine di 10ms.
Mentre Wi-Fi risulta più adatto all'accesso ad una rete locale, WiMAX è adatto per l'accesso punto-multipunto alla banda larga.
Per entrambe le tecnologie risulta importante la visibilità delle antenne, in caso contrario contrario il range tende a decadere rapidamente, anche se decisamente meno nel caso WiMAX.
Intrinseco di WiMAX è il QoS (Quality of Services) e una maggiore sicurezza: prioritizza e ottimizza il traffico e, a differenza del Wi-Fi, implementa diverse tecniche di crittografia, sicurezza ed autenticazione contro le intrusioni.
Una delle esigenze principali di una Smart Grid è quella di realizzare una rete di comunicazione WAN (Wide Area Network).

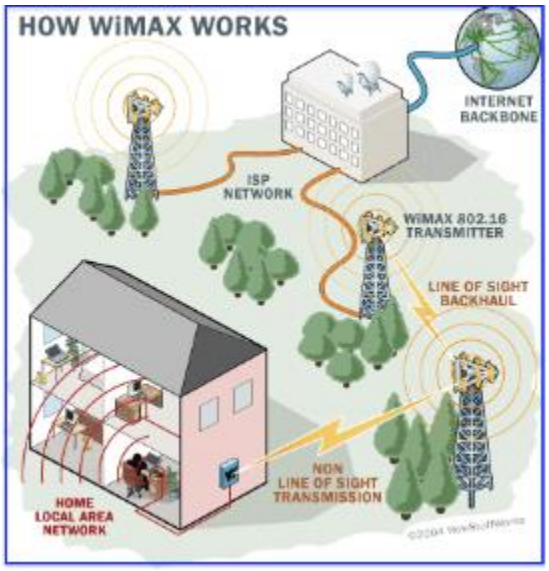


Figura 36. WiMAX concept

WiMAX è una tecnologia in grado di integrarsi con quelle presenti, soddisfacendo diverse tra le specifiche imposte da una tipica Smart Grid (Figura 36):

- Massima accessibilità e interoperabilità;
- Tempi di latenza inferiori ai
- 50ms;
- Larghezza di banda 5MHz;
- Throughput di 1 – 4 Mbps.

Lo standard IEC 61850 comprende due bus basati sulla tecnologia Ethernet commutata [64]:

- Il bus della stazione [65] collega tutte gli elementi e il livello di supervisione della stazione; trasporta principalmente informazioni di controllo, come misurazioni, interblocco e selezione prima dell'uso. In genere, il protocollo MMS (Manufacturing Messaging Specification) viene utilizzato per trasferire i dati tra i dispositivi elettronici intelligenti (IED) a livello di stazione e quelli a livello di baia mentre gli eventi di sottostazione orientati agli oggetti generici (GOOSE) si occupano del trasferimento dei dati da IED da baia a IED da baia.
- Il bus di processo [63] interconnette gli IED all'interno di una baia e trasporta principalmente misure, note come valori campionati (SV), per la protezione. Gli SV sono campionati ad un valore nominale di 4 kHz in reti a 50 Hz (4,8 kHz in reti a 60 Hz).

IEC 61850 non prescrive una topologia ad albero, stella o anello. In effetti, la stessa rete Ethernet fisica potrebbe trasportare sia la stazione che il bus di processo.

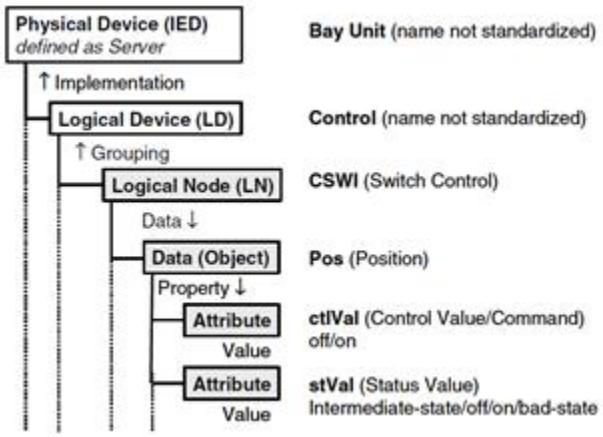


Figura 37. Il modello a oggetti dello standard IEC-61850

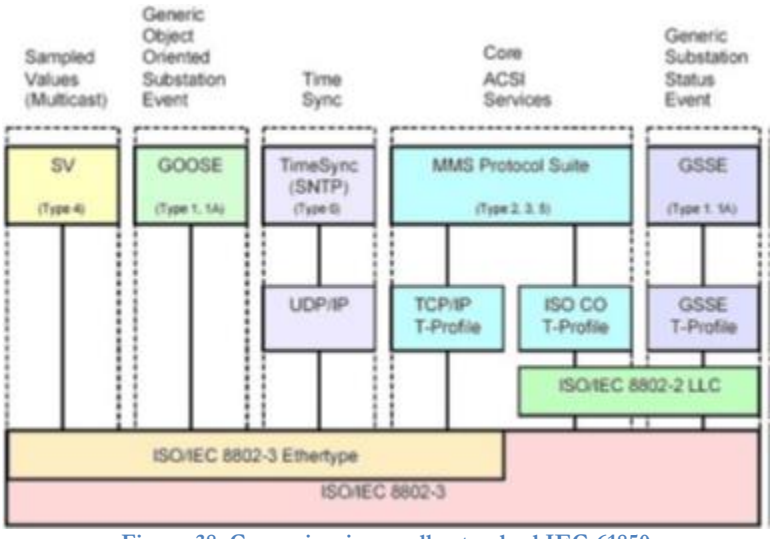


Figura 38. Comunicazione nello standard IEC-61850

Conclusioni
Rispetto alle esigenze della Smart Grid, IEC-61850 soddisfa i seguenti requisiti:

- Garantisce l'interoperabilità tra gli elementi tipici di una rete, appartenenti a produttori diversi, fornendo uno strumento di comunicazione adeguato e aperto;
- Garantisce la flessibilità richiesta in termini di possibili evoluzioni del sistema facilmente implementabili e fornisce un linguaggio comune per la condivisone delle informazioni;
- Supporta servizi (GOOSE, GSSE, SMV) che garantiscono le prestazioni richieste da applicazioni real-time tipiche della gestione della rete elettrica;
- Riduce i costi di configurazione, installazione e trasportabilità della strumentazione.

Limiti

- Presenta costi relativamente elevati per l'installazione dei server e dei dispositivi atti alla gestione globale dei dati;
- Non definisce tutti i livelli di comunicazione e, adeguandosi a protocolli esistenti, ne presenta alcuni limiti;
- L'architettura, sebbene non sia richiesta una configurazione manuale, si presenta piuttosto complessa e articolata.

3.4 Benefici dello standard IEC-61850

Una delle sfide significative nella conduzione delle reti è giustificare gli investimenti nell'automazione delle sottostazioni. Gli impatti positivi che l'automazione ha sui costi operativi, sull'aumento della qualità del servizio e sulla riduzione della risposta alle interruzioni sono ben noti. È essenziale prestare la massima attenzione a come l'uso di uno standard di comunicazione influisce sui costi di costruzione e gestione della sottostazione. I protocolli di comunicazione legacy sono stati tipicamente sviluppati con il duplice obiettivo di fornire le funzioni necessarie richieste dai sistemi di alimentazione elettrica riducendo al minimo il numero di byte che sono stati utilizzati dal protocollo a causa delle gravi limitazioni di larghezza di banda tipiche della tecnologia di collegamento seriale disponibile 10-15 anni fa, quando molti di questi protocolli furono inizialmente sviluppati. Successivamente, con la diffusione di Ethernet e dei moderni protocolli di rete come TCP / IP, questi protocolli legacy sono stati adattati per funzionare su TCP / IP-Ethernet. Questo approccio ha fornito le stesse capacità di base del sistema di alimentazione elettrica della versione con collegamento seriale, portando i vantaggi delle moderne tecnologie di rete nella sottostazione. Ma questo approccio ha un difetto fondamentale: i protocolli utilizzati erano ancora progettati per ridurre al minimo i byte in rete e non sfruttare l'enorme aumento della larghezza di banda che le moderne tecnologie di rete forniscono offrendo un livello più elevato di funzionalità che può ridurre significativamente i costi operativi e di implementazione dell'automazione delle sottostazioni [67],[68].

IEC 61850 non è un precedente protocollo di collegamento seriale rifuso su TCP / IP-Ethernet. IEC 61850 è stato progettato da zero per funzionare su moderne tecnologie di rete e offre una quantità di funzionalità senza precedenti che semplicemente non è disponibile dai protocolli di comunicazione legacy. Queste caratteristiche uniche della IEC 61850 hanno un impatto diretto e positivo sui costi di progettazione, costruzione, installazione, messa in servizio e funzionamento dei sistemi di alimentazione. Mentre i protocolli legacy su Ethernet consentono alle unità preposte alla conduzione di fare esattamente la stessa cosa che è stata fatta 10-15 anni fa utilizzando Ethernet. IEC 61850 consente miglioramenti fondamentali nel processo di automazione della sottostazione che semplicemente non sono possibili con un approccio legacy, con o senza TCP / IP-Ethernet. Per comprendere meglio i vantaggi specifici, esamineremo prima alcune delle caratteristiche e capacità chiave della IEC 61850 e poi spiegheremo come queste si traducono in vantaggi significativi che non possono essere raggiunti con l'approccio legacy.

3.4.1 Caratteristiche principali

Le caratteristiche principali dello standard IEC 61850 che consentono vantaggi unici sono numerosi, pertanto verranno elencati quelli ritenuti più importanti. Alcune di queste caratteristiche sono apparentemente piccole ma possono avere un

enorme impatto sui sistemi di automazione delle sottostazioni. Ad esempio, l'uso di VLAN e flag di priorità per GOOSE e SMV consente un uso molto più intelligente degli switch Ethernet che di per sé possono offrire vantaggi significativi agli utenti che non sono disponibili con altri approcci. Per motivi di brevità, elencheremo qui alcune delle caratteristiche chiave che forniscono vantaggi significativi agli utenti:

• **Utilizzo di un modello virtualizzato**. Il modello virtualizzato di dispositivi logici, nodi logici, ACSI e CDC consente di definire la definizione dei dati, dei servizi e del comportamento dei dispositivi oltre ai protocolli utilizzati per definire la modalità di trasmissione dei dati sulla rete.

• **Uso di nomi per tutti i dati**. Ogni elemento dei dati IEC 61850 è denominato utilizzando stringhe descrittive per descrivere i dati. I protocolli legacy, d'altra parte, tendono a identificare i dati in base alla posizione di archiviazione e utilizzano numeri di indice, numeri di registro e simili per descrivere i dati.

• **Tutti i nomi degli oggetti sono standardizzati e definiti in un contesto di sistema di alimentazione**. I nomi dei dati nel dispositivo IEC 61850 non sono dettati dal fornitore del dispositivo o configurati dall'utente. Tutti i nomi sono definiti nello standard e forniti in un contesto di sistema di alimentazione che consente di identificare immediatamente il significato dei dati senza dover definire mappature che mettono in relazione i numeri di indice e di registro con i dati del sistema di alimentazione come tensione e corrente.

• **I dispositivi si auto-descrivono**. Le applicazioni client che comunicano con i dispositivi IEC 61850 sono in grado di scaricare la descrizione di tutti i dati supportati dal dispositivo senza alcuna configurazione manuale di oggetti dati o nomi.

• **Servizi di alto livello**. ACSI supporta un'ampia varietà di servizi che supera di gran lunga quanto disponibile nel tipico protocollo legacy. GOOSE, GSSE, SMV e log sono solo alcune delle funzionalità uniche di IEC 61850.

• **Linguaggio di configurazione standardizzato**. SCL consente di definire con precisione la configurazione di un dispositivo e il suo ruolo nel sistema di alimentazione utilizzando file XML.

3.4.2 Maggiori benefici

Le caratteristiche sopra descritte per IEC 61850 offrono vantaggi sostanziali agli utenti che le comprendono e ne traggono vantaggio. Anziché limitarsi ad avvicinarsi a un sistema basato su IEC 61850 allo stesso modo di qualsiasi altro sistema, un utente che comprende e sfrutta le capacità uniche realizzerà vantaggi significativi che non sono disponibili utilizzando approcci legacy.

• **Elimina l'ambiguità degli acquisti**. SCL può essere utilizzato non solo per configurare dispositivi e sistemi di alimentazione, ma anche per definire con precisione i requisiti dell'utente per sottostazioni e dispositivi. Utilizzando SCL, un utente può specificare esattamente e in modo inequivocabile cosa ci si aspetta da fornire in ogni dispositivo che non è soggetto a interpretazioni errate da parte dei fornitori.

• **Costo di installazione inferiore**. Lo standard IEC 61850 consente ai dispositivi di scambiare rapidamente dati e stato utilizzando GOOSE e GSSE sulla LAN della stazione senza dover cablare collegamenti separati per ciascun relè. Ciò riduce significativamente i costi di cablaggio utilizzando in modo più completo la larghezza di banda LAN della stazione per questi segnali e i costi di costruzione riducendo la necessità di cavidotti, condotti, condutture, ecc.

• **Riduzione dei costi del trasduttore**. Anziché richiedere trasduttori separati per ciascun dispositivo che necessita di un segnale particolare, un'unica unità di fusione che supporta SMV può fornire questi segnali a molti dispositivi utilizzando un singolo trasduttore, cablaggio, calibrazione e costi di manutenzione.

• **Costi di messa in servizio inferiori**. Il costo per configurare e mettere in servizio i dispositivi è drasticamente ridotto perché i dispositivi IEC 61850 non richiedono la stessa configurazione manuale dei dispositivi legacy. Le applicazioni client non devono più essere configurate manualmente per ogni punto a cui devono accedere perché possono recuperare l'elenco dei punti direttamente dal dispositivo o importarlo tramite un file SCL. Molte applicazioni non richiedono altro che l'impostazione di un indirizzo di rete per stabilire le comunicazioni. La maggior parte della configurazione manuale viene eliminata drasticamente riducendo errori e rilavorazioni.

• **Minori costi di migrazione delle apparecchiature**. Poiché IEC 61850 definisce più aspetti visibili esternamente dei dispositivi oltre alla semplice codifica dei dati sul cavo, il costo per le migrazioni delle apparecchiature è ridotto al minimo. Differenze comportamentali da una marca di dispositivo a un altro vengono ridotte al minimo e, in alcuni casi,

completamente eliminate. Tutti i dispositivi condividono le stesse convenzioni di denominazione, riducendo al minimo la riconfigurazione delle applicazioni client quando questi dispositivi vengono modificati.

• **Costi di estensione inferiori**. Poiché i dispositivi IEC 61850 non devono essere configurati per esporre i dati, le nuove estensioni vengono facilmente aggiunte nella sottostazione senza dover riconfigurare i dispositivi per esporre i dati a cui non si accedeva in precedenza. L'aggiunta di dispositivi e applicazioni in un sistema IEC 61850 esistente può essere eseguita con un impatto minimo, e nel caso, su qualsiasi apparecchiatura esistente.

• **Costi di integrazione inferiori**. Utilizzando la stessa tecnologia di rete ampiamente utilizzata nell'azienda di servizi pubblici, il costo per integrare i dati delle sottostazioni nell'azienda viene sostanzialmente ridotto. Anziché installare costose RTU che devono essere configurate e gestite manualmente per ogni punto di dati necessario nell'applicazione del centro di controllo e dell'ufficio tecnico, le reti IEC 61850 sono in grado di fornire dati senza front-end di comunicazione separati o dispositivi di riconfigurazione.

• Implementare **nuove funzionalità**. I servizi avanzati e le caratteristiche uniche di IEC 61850 abilitano nuove funzionalità che semplicemente non sono possibili con la maggior parte dei protocolli legacy. Gli schemi di protezione di aree estese che normalmente sarebbero proibitivi in termini di costi diventano molto più fattibili. Poiché i dispositivi sono già collegati alla LAN della sottostazione, il costo incrementale per l'accesso o la condivisione di più dati del dispositivo diventa insignificante consentendo applicazioni nuove e innovative che sarebbero troppo costose da produrre altrimenti.

4 PROTOCOLLI MACHINE-TO-MACHINE A SUPPORTO DELLE NANO-GRIDE DELLA PIATTAFORMA DI GESTIONE COMESTO

4.1 SCENARIO DI RIFERIMENTO

Prima di descrivere i principali protocolli **Machine-To-Machine (M2M)** introduciamo lo scenario di riferimento sul quale basarci per effettuare opportunamente lo studio dei protocolli e delle loro principali caratteristiche.
La rete di comunicazione fisica considerata è composta principalmente da dispositivi IoT capaci sia di interagire con l'ambiente esterno e circostante tramite l'utilizzo di appositi sensori, sia di comunicare tra loro e con i gateway di riferimento, attraverso l'utilizzo di nuovi protocolli legati al contesto dell'**Internet Of Things (IoT)**, ovvero **CoAP (Constrained Application Protocol)** e **MQTT (Message Queue Telemetry Transport)**.
L'architettura utilizzata per lo studio delle feature dei suddetti protocolli è rappresentata nelle due figure sottostanti. Esse mostrano lo scenario utilizzando i 2 principali protocolli M2M: l'MQTT (Message Queue Telemetry Transport) e il CoAP (Constrained Application Protocol).
Dalle Figura 39 e Figura 40 si possono vedere i vari collegamenti tra i vari dispositivi considerati: il collegamento tra i dispositivi di campo (sensori) verso il gateway, il collegamento gateway verso internet e quindi verso un potenziale cloud dove si è supposto l'utilizzo di un classico protocollo HTTP.

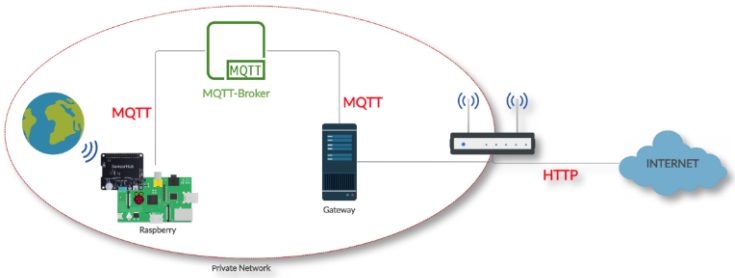


Figura 39. Rete IoT che usa il protocollo MQTT.

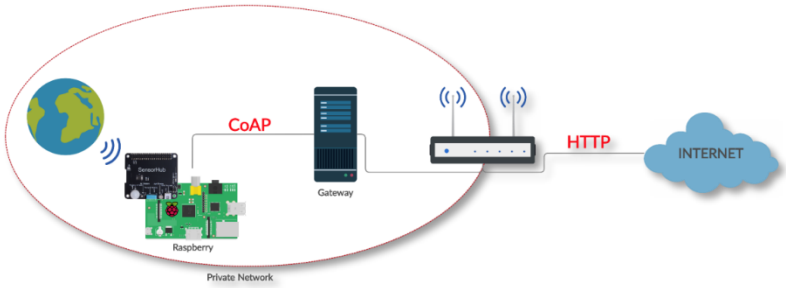


Figura 40. Rete IoT che usa il protocollo CoAP.

Come si può intuire analizzando entrambe le due figure appena introdotte, ovvero la Figura 39 e la Figura 40, tra di esse risulta essere presente una piccola differenza architetturale, legata alla tipologia di protocollo che è stato utilizzato per garantire la corretta trasmissione dei dati e delle informazioni, tra il nostro **dispositivo IoT** e il **Gateway di riferimento**: difatti si è deciso di operare lo sviluppo del nostro sistema, utilizzando e sfruttando entrambi i due principali protocolli di

Comunicazione adattati a livello applicativo e maggiormente riconosciuti ed utilizzati nel contesto dell'Internet of Things, ovvero da una parte, il protocollo CoAP (Constrained Application Protocol) e dall'altra lo standard protocollare per eccellenza, ovvero il protocollo MQTT (Message Queue Telemetry Transport
Prima però di analizzare nel dettaglio le caratteristiche principali dei protocolli M2M, è opportuno fare una piccola descrizione tecnica e precisa delle componenti hardware e fisiche, che definiscono la struttura interna del nostro tipo di dispositivo IoT utilizzato; il quale risulta essere composto da due principali elementi di interesse: da una parte emerge la base funzionale del dispositivo, risultante rappresentata da un semplice Raspberry Pi 3 Model B+, mentre dall'altra parte, per quanto riguarda l'utilizzo del sensore esterno da integrare, si è optato per un DockerPi Sensor Hub Development Board vedi Figura 41.

Raspberry Pi 3 Model B+
CARATTERISTICHE
- Broadcom BCM2837B0, SoC Cortex-A53 da 1,4 GHz e 64 bit
- Connetività wireless dual-band 802.11ac
- Bluetooth 4.2
- Ethernet veloce (Gigabit Ethernet su USB 2.0)
- Supporto PoE (con PoE HAT separato)
- Rete PXE potenziata
- Avvio da dispositivi USB Mass Storage
- Migliore gestione termica

SPECIFICHE
- 1 GB di memoria SDRAM LPDDR2
- Alimentazione cc di ingresso da 5 V/2,5 A (micro USB)
- Connettività wireless:
- LAN wireless da 2,4 GHz e 5 GHz IEEE 802.11.b/g/n/ac e Bluetooth 4.2/BLE
- Gigabit Ethernet su USB 2.0 (resa massima a 300 Mbps)
- 4 porte USB 2.0
- Header con GPIO a 40 pin
- Uscita video HDMI full-size
- Uscita audio stereo a 4 poli e porta video composito
- Porta CSI per il collegamento di una fotocamera Raspberry Pi
- Porta DSI per il collegamento di un display touchscreen Raspberry Pi
- Porta micro SD per caricare il sistema operativo e conservare i dati
- Temperatura di funzionamento: da 0 °C a +50 °C
- Dimensioni: 120 mm x 75 mm x 34 mm
- Peso: 75 g

DockerPi Sensor Hub Development Board SKU(EP-0106)

SPECIFICHE
- DockerPi Series
- Programmable
- Read directly(without programming)
- Extend the GPIO Pins
- Ext. Temperature Detection, Thermistor Detection Temperature Range -30℃~127℃
- OnBoard Temperature DHT11 -20℃~60℃

- P. Temperature Sensor -40°C~80°C.
- Humidity detection, sensor detection range 20% Rh ~ 95% Rh
- Light intensity detection, detection range: 0Lux~1800Lux
- Pressure detection, detection range: 300 Pa ~ 1100 hPa
- Biopsy test (biopsy test with corresponding indicator), maximum detection angle of 100 degrees, maximum distance of 12m
- Can Stack with other Stack board
- Independent of the mainboard hardware (require I2C support)

Figura 41. Scheda Raspberry con board DockerPi

4.2 Il protocollo COAP

Il **Constrained Application Protocol (CoAP)** è un particolare e specializzato protocollo applicativo (Application Protocol) definito all'interno del RFC 7252 [69], nato con lo scopo principale di favorire lo scambio informativo tra "constrained devices", anche chiamati nodi, risultanti interconnessi tra loro all'interno di quella che spesso viene chiamata col termine di WSN (Wireless Sensor Network).
A livello puramente architetturale, tale protocollo segue totalmente il paradigma Request/Response, supportando la possibilità di gestire variegati metodi di richiesta, come GET, PUT, POST e DELETE: ciò fa capire come, in realtà, CoAP non è altro che una versione più semplificata, efficiente e diretta del protocollo HTTP, riadattata proprio per funzionare correttamente in contesti di rete legati al mondo dell'Internet of Things (IoT). Una piccola e sostanziale differenza che intercorre però tra i due protocolli è essenziale definirla, e riguarda il fatto che a "livello Trasporto", di quello che può essere visto come Stack Protocollare TCP/IP, CoAP utilizza prevalentemente UDP mentre HTTP si basa unicamente su TCP. Un'altra importante feature che CoAP mette a disposizione, è rappresentata dalla possibilità di creare e rendere disponibile Observing Resources, ovvero risorse che risultano essere "osservabili": ciò significa che viene data la possibilità ad un nodo generico della rete, chiamato Observer di "registrarsi" nei confronti di una particolare risorsa Subject messa a disposizione da parte di un altro qualsiasi nodo della stessa rete, affinché, ogni qualvolta lo stato interno della risorsa considerata subisce degli aggiornamenti, l'Observer possa essere notificato di tale cambiamento e dei parametri che l'hanno indotto.
Chiaramente la caratteristica principale di tale protocollo è quella di realizzare una comunicazione tra dispositivi con risorse limitate in termini computazionali ed energetici. Il lavoro di standardizzazione del protocollo è opera del Constrained RESTful Environment (CoRE) Working Group della Internet Engineering Task Force (IETF) [69].

Prima però di andare ad analizzare le caratteristiche che maggiormente caratterizzano il protocollo appena descritto, è opportuno, tramite appositi diagrammi di sequenza, analizzare quella che è stata la logica funzionale adottata all'interno della struttura di rete precedentemente introdotta, al fine di garantire la corretta realizzazione dello stesso protocollo:

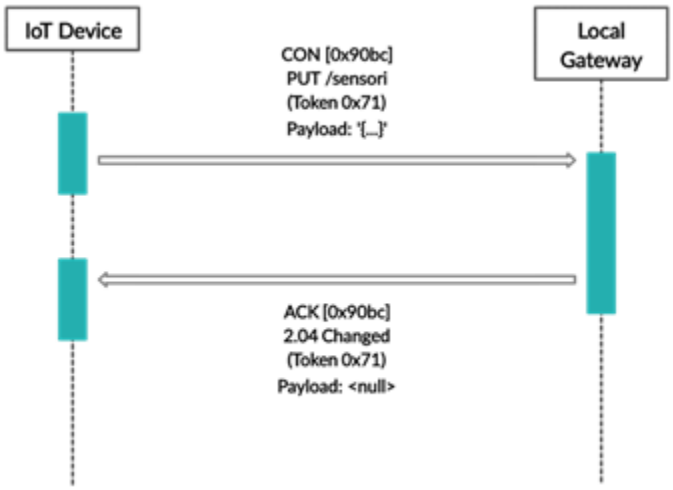


Figura 42. IoT Network-LAN-SeqDiag-COAP-PUT (versione base)]

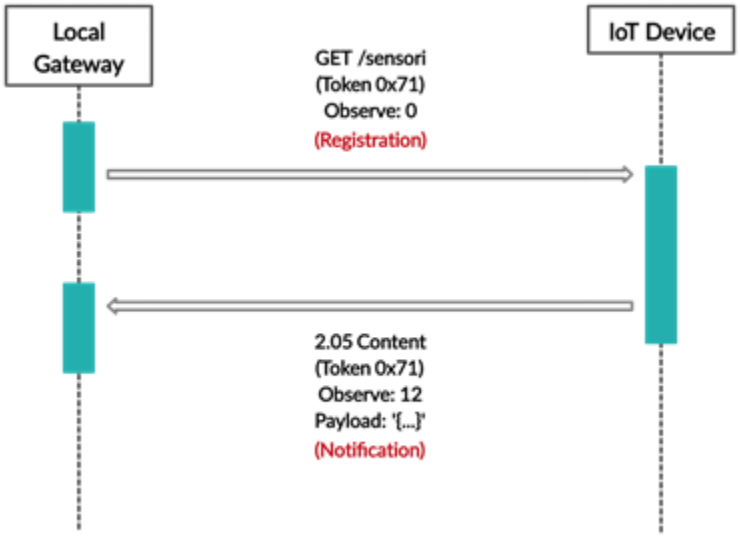


Figura 43. IoT Network-LAN-SeqDiag-COAP-OBS (versione base)]

Analizzando la Figura 42, si può notare come l'utilizzo applicativo del protocollo CoAP, nel nostro contesto di rete interna, è molto semplice e lineare: difatti abbiamo, da una parte, un'entità chiamata **Local Gateway** che, assumente ovviamente il ruolo di **CoAP Server**, crea e aggiunge all'interno del namespace associato alle sue risorse esposte e raggiungibili, una risorsa **/sensori**, che conterrà, in formato JSON, i dati ambientali prelevati dal **dispositivo IoT**, per poi inviarli ed inoltrarli, tramite protocollo HTTP, al Server Remoto; dall'altra parte invece spicca la presenza del dispositivo IoT (Rapsberry+Docker Sensor Hub) che molto semplicemente, fungendo da **CoAP Client**, invia una richiesta di tipo PUT verso l'indirizzo coap://<ip_local_gateway>/sensori associato al Local Gateway di riferimento, contenente come payload proprio i dati estratti ed ottenuti tramite sensore (sottoforma di JSON).
Analizzando invece la Figura 43, la situazione, anche se può sembrare molto simile al caso appena descritto, è potenzialmente differente: infatti emerge sempre la presenza sostanziale di due entità fondamentali, le stesse analizzate nella figura precedente, ovvero il **dispositivo IoT** e il **Local Gateway**, ma a differenza della situazione precedente, in questa circostanza si ha che il ruolo del CoAP Server è svolto dal dispositivo IoT mentre il ruolo del CoAP Client è svolto dal rispettivo Local Gateway: infatti, come si può ben notare dai messaggi scambiati all'interno del diagramma di sequenza mostrato, è il Local Gateway che invia al dispositivo IoT una richiesta di tipo GET con il valore di *Observe Option* settato a 0, indicante proprio il volersi "registrare" alla ricezione delle potenziali notifiche derivanti da successivi cambiamenti di stato della risorsa /sensori presente e messa a disposizione proprio da parte dello stesso dispositivo IoT; d'altra parte invece quest'ultimo, ricevuta la richiesta di registrazione da parte dell'**Observer (Local Gateway)**, lo inserirà all'interno della lista dei dispositivi associati alla risorsa /sensori, e periodicamente, ogni volta che cambierà lo stato di tale risorsa, come si evince dalla figura, lo andrà a notificare a tutti gli Observer ad essa associati, compreso quindi anche il Local Gateway, il quale, una volta ricevuta la notifica di questi aggiornamenti di stato, li inoltrerà poi verso il Server Remoto, tramite l'ausilio del protocollo HTTP.
Terminata questa descrizione puramente architetturale e protocollare sulla struttura interna e generale della nostra LAN "privata" basata sull'utilizzo di CoAP, si può passare ad analizzare e ad approfondire maggiormente nel dettaglio gli aspetti implementativi e tecnici che ne hanno permesso una sua corretta realizzazione. Al fine di raggiungere lo scopo implementativo appena citato, le componenti software che sono state progettate e sviluppate con l'obbiettivo di governare l'intera logica applicativa interna sia del Local Gateway che del dispositivo IoT sono state completamente realizzate utilizzando come linguaggio di programmazione di base **Python**, integrando successivamente la versione basilare di quest'ultimo con un insieme di moduli e librerie di terze parti necessarie a gestire la corretta realizzazione dei diversi passi

protocollari per CoAP. Al fine di realizzare ciò, è stato utilizzato un particolare package chiamato e conosciuto col termine di **aiocoap**, che risulta essere attualmente una delle principali e migliori librerie Python che gestiscono la corretta implementazione dei dettagli protocollari del CoAP, nel rispetto dell'apposito RFC 7252.

4.2.1 Caratteristiche del protocollo CoAP

Le principali caratteristiche che il protocollo CoAP presenta sono elencate di seguito:

- protocollo web per nodi di rete con risorse limitate adatto a comunicazioni macchinamacchina;
- trasporto su protocollo a datagramma come UDP con a_dabilità opzionale;
- scambio asincrono di messaggi;
- basso overhead e bassa complessità di parsing dell'header;
- supporto a risorse URI e ad informazioni content-type del payload;
- realizzazione in modo semplice d'intermediari (proxy);
- possibilità di memorizzare in cache risposte per ridurre tempi di risposta ed occupazione di banda;
- traducibilità nel protocollo privo di stati HTTP con possibilità di realizzare proxy per garantire a nodi HTTP l'accesso a risorse CoAP e viceversa.

4.2.2 Formato del messaggio CoAP

L'interazione tra nodi CoAP avviene in modo simile al modello **client/server** del protocollo HTTP. Tuttavia la natura delle interazioni macchine-to-macchine che avvengono tra dispositivi remoti in IoT suggerisce una implementazione del protocollo CoAP in cui ogni nodo (end point) agisca sia da client che da server. Una richiesta CoAP è equivalente ad una richiesta HTTP: essa è inviata da un client ad un server per richiedere a questo l'esecuzione di una azione (tramite un codice di metodo) su una risorsa (identificata da un URI). Il server invia poi al client originario una risposta contenente un codice di risposta ed una eventuale rappresentazione della risorsa richiesta. Diversamente da HTTP, il protocollo CoAP realizza questi scambi di richieste/risposte in maniera asincrona su un protocollo di trasporto a datagramma come UDP.

I messaggi CoAP possono essere di 4 tipi:

1. Confirmable (CON): messaggio trasmesso in maniere affidabile;
2. Non confirmable (NON): messaggio trasmesso in maniere non affidabile;
3. Acknowledgement (ACK): serve per confermare la ricezione di un messaggio di tipo CON;
4. Reset (RST): conferma la ricezione di un messaggio, ma indica l'impossibilità di processarlo.

I codici inclusi in alcuni di questi messaggi specificano se si tratta di una richiesta o di una risposta. Si può pensare al protocollo CoAP come ad un livello composto da due sottolivelli (vedi Figura 44):

- un sottolivello di messaggistica che si occupa della gestione dello scambio di messaggi che come detto prima è asincrono e vincolato ad UDP;
- un sottolivello delle interazioni richiesta/risposta che utilizza i codici di Metodo e di Risposta per elaborare la richiesta o risposta.

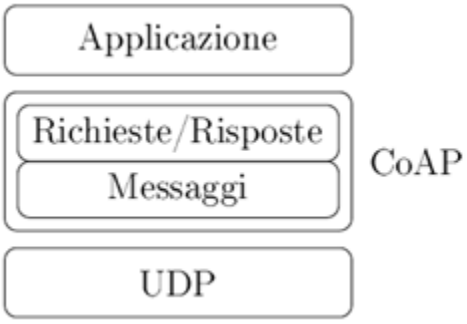


Figura 44. Struttura a livelli del protocollo CoAP.

L'informazione trasportata da un messaggio CoAP è codificata in formato binario nei seguenti blocchi illustrati in Figura 45:

- Header: il primo elemento è un header di lunghezza fissa pari a 4 byte:
  - Ver: Versione, intero non negativo di 2 bit. Indica la versione del protocollo CoAP implementata nel nodo. Poichè la standardizzazione del protocollo non è ancora stata ultimata, tale versione viene impostata ad 1 riservando numeri successivi a versioni future.
  - T: Tipo, intero non negativo di 2 bit. Indica se il messaggio è di tipo Confermabile (0), Non-Confermabile (1), Acknowledgement (2) o Reset (3).
  - TKL: Token Length, intero non negativo di 4 bit. Indica la lunghezza del campo token che come detto è variabile da 0 a 8 byte. Il suo valore serve a correlare richieste e risposte. Lunghezze che vanno da 9 a 15 byte sono riservate, non vanno inviate e se ricevute devono essere interpretate come errore del messaggio del formato.
  - Codice: intero non negativo di 8 bit. Indica se il messaggio è una richiesta (codici 1-31), una risposta (codici 64-191) o è vuoto (codice 0). Tutti gli altri codici sono riservati. Nel caso di una richiesta il campo Codice ne indica il Metodo mentre nel caso di una risposta tale campo indica il Codice di Risposta.
  - Message ID: intero non negativo di 16 bit con ordinamento dei byte network byte order. È utilizzato per la rivelazione di messaggi duplicati e per associare messaggi di tipo Acknowledgement/Reset a messaggi di tipo Confermabile/Non-Confermabile secondo le regole di generazione ed associazione.

- Token value: segue il campo contenente il valore del token di lunghezza variabile tra 0 ed 8 byte. Una risposta viene abbinata alla rispettiva richiesta per mezzo di un token che viene incluso dal client nella richiesta. Esso è utilizzato in modo “locale” per differenziare richieste concorrenti. Può essere inteso come un identificatore di una richiesta. Ogni messaggio trasporta un token, il quale può anche essere di lunghezza nulla. Il client dovrebbe generare i token in maniera tale che quelli attualmente in uso, per una data associazione di end point, siano unici.

- Opzioni: dopo il token vi è una eventuale sequenza di zero o più opzioni CoAP codificate in formato type-length-value. Il CoAP definisce un certo numero di opzioni che possono essere incluse nei messaggi. Un messaggio può contenere più opzioni e ogni istanza di un’opzione specifica:
  - il numero dell’opzione CoAP definita;
  - la lunghezza del valore dell’opzione (Option Length);
  - il valore dell’opzione (Option Value).

Il numero dell'opzione CoAP non viene specificato in modo diretto (Figura 45). Per ogni istanza di un'opzione, il numero associato a quest'ultima viene calcolato come la somma di un delta e del numero che rappresenta l'istanza dell'opzione precedente (cioè come la somma di tutti i delta delle istanze precedenti). Le istanze devono apparire in ordine rispetto ai loro numeri di opzione. Per la prima istanza in un messaggio, si assume che essa sia preceduta da un'istanza con numero di opzione pari a zero. Multiple istanze di una stessa opzione possono essere incluse utilizzando un delta pari a zero. I campi di un'opzione sono definiti come segue:

- Option Delta: intero a 4 bit senza segno. Un valore da 0 a 12 indica il delta dell'opzione;
- Option Length: intero a 4 bit senza segno. Un valore da 0 a 12 indica la lunghezza (in byte) dell'option value;
- Option Value: una sequenza di byte di lunghezza pari a option length, la cui lunghezza e il cui formato dipendono dalla rispettiva opzione.

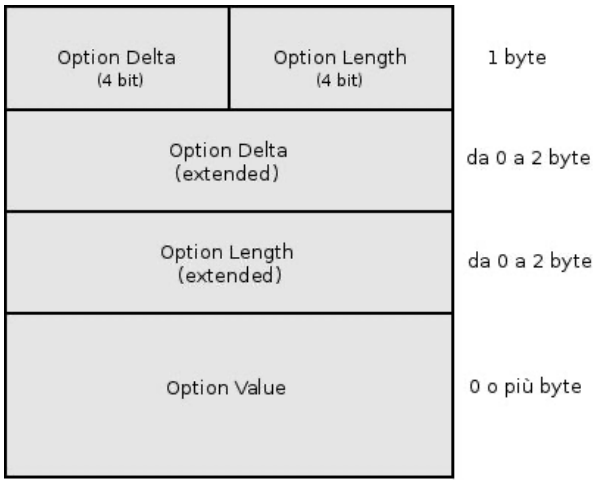


Figura 45. Formato delle Opzioni CoAP.

- Payload: infine, dopo la sequenza di opzioni, può seguire un eventuale payload che occupa la parte rimanente del datagramma. Se tale payload è presente e di dimensione non nulla, esso viene preceduto da un marker lungo 1 byte formato da bit tutti unitari (0xFF) che sta ad indicare la fine del campo delle opzioni e l'inizio del campo Payload.

Il payload rappresenta la parte utile del messaggio. Se presente e di lunghezza non nulla, è preceduto da un payload marker di lunghezza fissa pari a 1 byte e di valore pari a 0xFF. Il payload marker indica la fine delle opzioni e l'inizio del payload. L'assenza del payload marker denota un payload di lunghezza nulla mentre la presenza dello stesso, seguito da un payload di lunghezza nulla, va processata come "formato errato del messaggio"

Un messaggio vuoto ha il campo Code settato a zero. In questo caso il campo TKL deve essere settato a zero e non ci devono essere dei byte dopo il campo Message ID, altrimenti essi devono essere processati come formato errato del messaggio. I dati del payload si estendono fino alla fine del datagramma UDP e la lunghezza del payload viene calcolata in base all'intera dimensione del datagramma (vedi Figura 46).

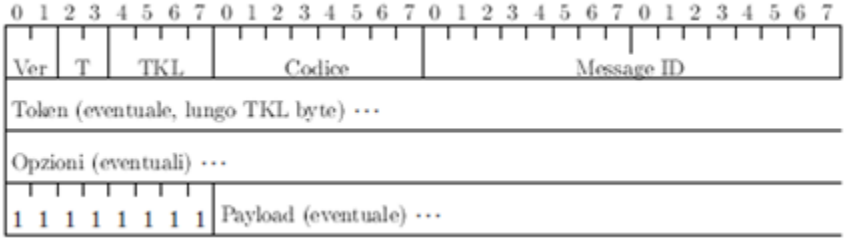


Figura 46. Formato del messaggio CoAP.

4.2.3 Modalità di trasmissione dei messaggi

I messaggi CoAP vengono scambiati in maniera asincrona fra gli endpoint. Il CoAP è spesso abbinato a protocolli di trasporto non affidabili come, ad esempio, l'UDP. Esso implementa un meccanismo di affidabilità leggero tentando di ricreare un insieme completo di caratteristiche simile a quello che contraddistingue il protocollo TCP. Tale meccanismo prevede le due seguenti caratteristiche [69]:

- trasmissione affidabile con exponential back-off per i messaggi CON
- individuazione della duplicazione dei messaggi sia per i messaggi CON che per i messaggi NON.

4.2.3.1 Messaggi Trasmessi in maniera affidabile

La trasmissione affidabile di un messaggio inizia marcando il messaggio come CON nell'header. Un messaggio CON trasporta sempre una richiesta o una risposta e non deve essere vuoto a meno che non venga usato per suscitare un messaggio di Reset (RST).
Il destinatario (receiver) deve riconoscere il messaggio inviando un messaggio ACK oppure, nel caso in cui non riesca a processare correttamente il messaggio (es. message format error), deve rifiutarlo.
Il messaggio ACK deve avere lo stesso Message ID del messaggio CON e deve trasportare una risposta o essere vuoto. Per rifiutare un messaggio CON, il destinatario, può spedire un messaggio RST oppure semplicemente ignorare il messaggio in entrata.
Il mittente (sender) ritrasmette il messaggio CON a intervalli di tempo esponenziali crescenti fino a che non riceve un ACK o un RST, o esaurisce il numero massimo di tentativi di ritrasmissione. La ritrasmissione è controllata da due fattori, di cui un endpoint deve tenere traccia per ogni messaggio CON che invia, mentre attende un ACK o un RST, vedi Figura 47 [69]:

- un timeout: per un nuovo messaggio CON il timeout iniziale è settato ad un numero casuale compreso fra i valori ACK TIMEOUT e (ACK TIMEOUT · ACK RANDOM FACTOR)
- un contatore di ritrasmissione (retransmission counter): per un nuovo messaggio CON è settato a zero.

Exponential back-off mechanism

Quando il timeout viene raggiunto e il retransmission counter è minore del valore MAX RETRANSMIT, il messaggio viene ritrasmesso, il retransmission counter incrementato e il timeout raddoppiato. Se il retransmission counter raggiunge il valore MAX RETRANSMIT in un timeout o se l'end point riceve un RST, il tentativo di trasmettere il messaggio è cancellato e il processo viene informato del failure.
D'altra parte, se invece l'endpoint riceve un ACK in tempo, la trasmissione si considera effettuata con successo.

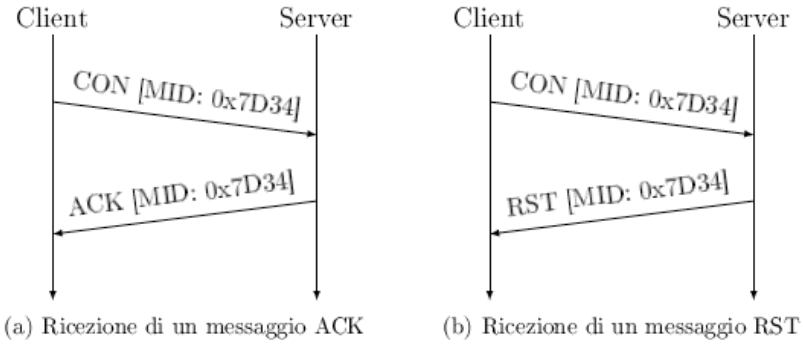


Figura 47. Trasmissione affidabile di messaggi CoAP.

4.2.3.2 Messaggi Trasmessi in maniera non affidabile

I messaggi che non richiedano una trasmissione affidabile come ad esempio la comunicazione periodica di una lettura di un sensore dove è sufficiente anche una conoscenza non immediata dello stato della risorsa monitorata, possono essere trasmessi senza richiedere riscontro da parte del destinatario specificando nell'header che il messaggio è di tipo *Non-Confermabile* (NON).
Un messaggio di tipo Non-Confermabile trasporta sempre o una richiesta o una risposta e non può essere vuoto. Ogni destinatario alla ricezione di un messaggio Non-Confermabile non deve inviare un messaggio di riscontro al mittente (Figura 48). Qualora il destinatario si accorga di non essere in grado di elaborare e/o interpretare correttamente il messaggio deve scartare il messaggio ignorandolo semplicemente ed eventualmente inviando un messaggio di RST corrispondente.

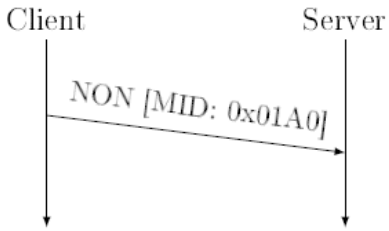


Figura 48. Trasmissione non affidabile dei messaggi CoAP.

A livello CoAP non c'è modo per il sender di sapere se un NON è stato ricevuto o meno. Il sender potrebbe decidere di trasmettere copie multiple di un messaggio NON entro un tempo pari a MAX TRANSMIT SPAN, vedi Figura 49. Per permettere al receiver di agire su di un'unica copia del messaggio, i messaggi Non-confirmable specificano un Message ID.

Parametro	Valore
ACK_TIMEOUT	2 secondi
ACK_RANDOM_FACTOR	1.5
MAX_RETRANSMIT	4
MAX_TRANSMIT_SPAN	45 secondi

Figura 49. Valori di default di alcuni dei parametri di trasmissione previsit dal CoAP.

4.2.3.3 Associazione dei messaggi

Un riscontro di tipo ACK o RST è associato ad un messaggio CON o NON attraverso il suo Message ID presente nell'header del messaggio. Esso viene generato dal mittente di una richiesta CON/NON in modo che lo stesso Message ID non sia riutilizzato all'interno della finestra temporale TEXCHANGE (tempo che intercorre tra l'invio di un messaggio CON e l'istante in cui il mittente di tale messaggio è sicuro di non poter più ricevere riscontro) per comunicare con lo stesso host. Tale Message ID viene quindi riportato dal destinatario nell'header del messaggio di riscontro ACK/RST per consentire al mittente la corretta associazione con il messaggio CON/NON corrispondente.
Il metodo più semplice per generare un diverso Message ID ogni volta che si comunica con lo stesso host all'interno di TEXCHANGE consiste nel memorizzare staticamente una variabile per valore del Message ID e cambiarla ogniqualvolta venga inviata una nuova richiesta CON/NON indipendentemente dall'host di destinazione. Esistono 2^{16} possibili Message ID. Esiste la possibilità che un destinatario riceva più copie dello stesso messaggio CON/NON (avente cioè stesso message ID, stesso indirizzo e stessa porta di origine) all'interno di TEXCHANGE o TNON (tempo che intercorre tra l'invio di un messaggio NON e l'istante in cui il mittente può riutilizzare il Message ID di tale messaggio). Questo può accadere ad esempio quando dei riscontri ACK/RST non giungano a destinazione o arrivino dopo il timeout di ritrasmissione del mittente oppure quando un host decide di inviare più copie dello stesso messaggio NON. In tale situazione l'host che riceve tali duplicati deve: o inviare un riscontro ACK identico per ogni messaggio CON ricevuto all'interno di TEXCHANGE ed elaborare ogni richiesta/risposta del messaggio una sola volta scartando i duplicati; o ignorare ogni messaggio NON duplicato ricevuto all'interno di TNON ed elaborare ogni richiesta/risposta del messaggio una sola volta scartando i duplicati.

4.2.4 Richieste e risposte

Il CoAP opera secondo un modello di richieste e risposte simile a quello utilizzato dal protocollo http, la comunicazione tra host CoAP avviene nel seguente modo: un client invia una o più richieste ad un server che elabora la richiesta ed invia una risposta. Diversamente da HTTP le richieste e risposte non vengono inviate attraverso una connessione stabilita in precedenza ma sono scambiate in modo asincrono attraverso i messaggi CoAP.

4.2.4.1 Richieste

Una richiesta CoAP è formata da un metodo da applicare ad una risorsa, l'identi_catore URI di tale risorsa, un eventuale payload con indicatore del formato ed eventuali metadati aggiuntivi. Un messaggio confermabile o non confermabile di richiesta viene creato specificando nel campo Codice dell'header il codice di Metodo della richiesta (codici 1-31) assieme ad altre informazioni aggiuntive incluse nel messaggio. Un host che riceva una richiesta avente codice di metodo non riconosciuto o non supportato deve inviare una risposta piggy-backed con codice di risposta 4.05 (Method Not Allowed).

Metodi

È possibile definire i metodi supportati da CoAP in base al tipo di comportamento da esso adottato, mutuati dal protocollo HTTP:
GET: recupera le informazioni relative alla rappresentazione di una determinata risorsa, la quale è identificata dall'URI della richiesta, tramite una funzione eseguita dal server dove tale risorsa risiede. Il metodo GET è sicuro (non modifica in alcun modo la rappresentazione di una risorsa) ed idempotente (l'effetto di più richieste identiche è lo stesso di quello di una sola richiesta). I codici di risposta possono essere 2.05 (Content) oppure 2.04 (Valid).
POST: richiede che una rappresentazione, racchiusa in una richiesta, venga processata. La funzione eseguita dal metodo è determinata dal server che mette a disposizione la risorsa (origin server) e dipende dalla “risorsa target". La funzione può consistere nella creazione di una nuova risorsa o nell'aggiornamento della risorsa target esistente. Il codice di risposta è 2.01 (Created) nel caso sia stata creata una nuova, 2.04 (Changed) nel caso sia stata modificata una risorsa già presente senza

crearne una nuova oppure 2.02 (Deleted) nel caso la risorsa indicata sia stata cancellata. Il metodo POST non è né sicuro, né idempotente.
PUT: richiede che una risorsa identificata da un URI di richiesta, sia aggiornata o creata con la rappresentazione inclusa nella richiesta. Il formato di rappresentazione è specificato nell'opzione “Content-Format". Se la risorsa indicata dall'URI esiste, la rappresentazione inclusa nel messaggio è da considerare una versione modificata della risorsa stessa e si deve rispondere con un codice 2.04 (Changed) qualora la modifica sia avvenuta con successo. Se la risorsa indicata non esiste nel server, questo può crearne una nuova identificata dall'URI indicato rispondendo con codice 2.01 (Created). Qualora invece la risorsa indicata non possa essere né modificata né creata, il server deve rispondere con un codice di errore appropriato.
Il metodo PUT non è sicuro ma è idempotente
DELETE: richiede che una risorsa, identificata da un URI di richiesta, venga cancellata. Se l'operazione di eliminazione della risorsa avviene con successo o la risorsa indicata non esisteva prima della richiesta, il messaggio di risposta deve avere codice 2.02 (Deleted). Il metodo DELETE non è sicuro ma è idempotente.

Formato degli URI

Il formato di un URI di tipo CoAP si presenta come segue [CoAP11]:

"coap(s):" "//" host [":" port] path ["?" query]

- coap(s): la sintassi “coap://" si utilizza per una connessione standard, mentre la sintassi “coaps://" viene utilizzata per connessioni sicure che utilizzano il protocollo DTLS.
- host: è l'host Internet al quale il server è raggiungibile. Può essere un nome o, più comunemente, un indirizzo IP
- port: indica la porta UDP alla quale il server è in ascolto. La porta di default è la 5683
- path: identi_ca una risorsa. È costituito da un insieme di segmenti separati dal carattere / (slash)
- query: paramterizza una risorsa. È formata da una sequenza di argomenti separati dal carattere & (ampersand). La sintassi degli argomenti è del tipo “key = value".

4.2.4.2 Risposte

Dopo aver ricevuto ed interpretato una richiesta, un server risponde con un messaggio di risposta CoAP che è associato alla richiesta originaria tramite il token generato dal client e incluso nel messaggio di richiesta. Analogamente al Codice di Stato nel protocollo HTTP, una risposta CoAP è contraddistinta da un Codice Risposta (codici 64-191), presente nel campo Codice dell'header del messaggio, che indica l'esito del tentativo di comprendere e soddisfare la richiesta ricevuta.
Una risposta _e identi_cata dal campo Code nell'header del messaggio, settando tale campo con il codice della risposta.
L'elenco completo dei codici di risposta che devono essere settati nel campo Code dell'header, si trova nel “CoAP Response Code Registry" [69]. Per quanto concerne i codici delle risposte,

Codici di risposta

Gli 8 bit che formano il Codice Risposta sono suddivisi secondo la notazione **classe.dettaglio** (Figura 50):

- i 3 bit più significativi specificano la classe della risposta;
- i rimanenti 5 bit meno significativi indicano un dettaglio aggiuntivo all'interno della classe di risposta.

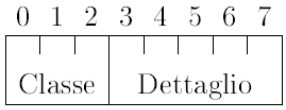


Figura 50. Struttura del Codice Risposta CoAP.

Esistono 3 classi di risposta:

- 2 - Successo: la richiesta è stata ricevuta, interpretata e accettata;
- 4 - Errore del client: la richiesta contiene errori di sintassi o non può essere soddisfatta;
- 5 - Errore del server: il server non è riuscito a soddisfare una richiesta accettata come valida.

Qualora un host riceva un codice di risposta non supportato o non riconosciuto all'interno della classe 4 (Errore del client) o 5 (Errore del server) essi vanno trattati come codici di risposta equivalenti al generico codice di risposta di quella classe (4.00 o 5.00 rispettivamente). Se si riceve un codice di risposta non supportato o non riconosciuto all'interno della classe 2 (Successo), poiché non esiste un generico codice di risposta 2.00 all'interno di tale classe che indichi esito positivo della richiesta, l'host deve interpretare tale risposta semplicemente come un riscontro della riuscita della richiesta senza ulteriori dettagli aggiuntivi.

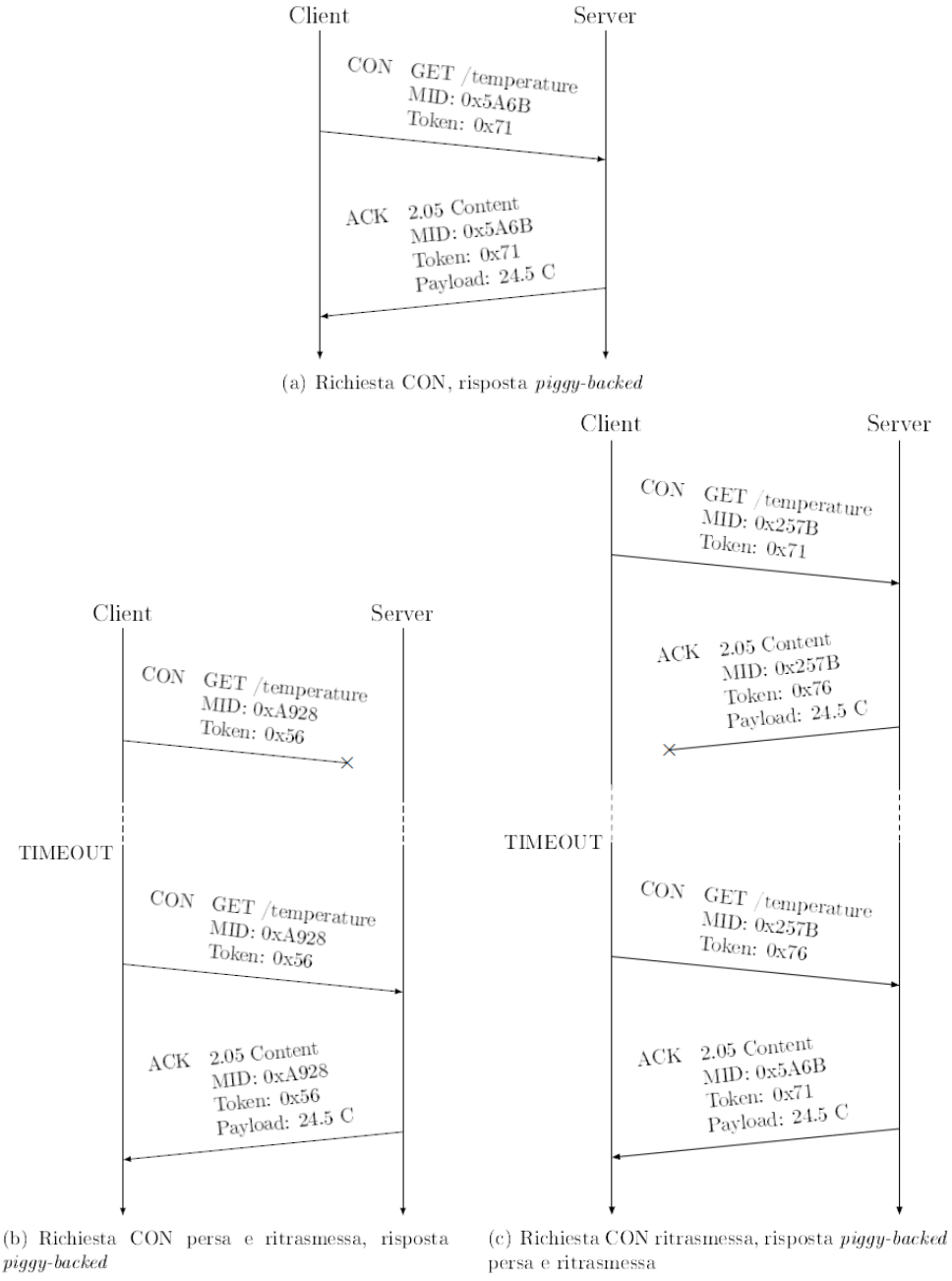
Risposta Piggy-backed

Se la risposta ad una richiesta di tipo CON è disponibile immediatamente, il server può inviarla direttamente nel messaggio di riscontro (ACK) della richiesta CON appena ricevuta (Figura 51(a)). In questo modo si elimina la necessità di dover trasmettere la risposta in un secondo messaggio distinto dal riscontro della richiesta ricevuta. Questo tipo di risposta viene detto piggy-backed e non necessita di un riscontro in quanto qualora non giungesse a destinazione, il client provvederebbe a ritrasmettere la richiesta per un numero fissato di volte. In Figura 51 sono illustrati due casi di risposta piggy-backed in cui vengono persi dei messaggi: nella Figura 52(b) viene perso il messaggio di richiesta mandato dal client e la sua ritrasmissione genera la risposta piggy-backed che giunge a destinazione; nella Figura 51(c) il primo messaggio di risposta piggy-backed viene perso ed allo scadere di un timeout il client procede alla ritrasmissione del messaggio di richiesta che giungerà a destinazione.

Risposta Separate

Può accadere che la risposta ad un messaggio di richiesta di tipo CON non sia immediatamente disponibile per varie ragioni. Ad esempio il server può impiegare per l'ottenimento della rappresentazione della risorsa richiesta un tempo maggiore del tempo di attesa del timeout di ritrasmissione del client, nel qual caso quest'ultimo procederà alla ritrasmissione del messaggio di richiesta. In questi casi è opportuno che nel server venga avviato un timer di riscontro in contemporanea all'inizio del processo di ottenimento della rappresentazione della risorsa. Allo scadere di questo timer il server procede all'invio del riscontro ACK della richiesta CON ricevuta. Se la rappresentazione della risorsa viene ottenuta prima del timeout allora si procede al suo invio piggy-backed nel riscontro (stesso Message ID e stesso token). Se invece allo scadere del timeout il server non è riuscito ad ottenere la rappresentazione della risorsa il server procede all'invio del riscontro della richiesta CON ricevuta come ACK vuoto (che verrà ritrasmesso nel caso di ricezione di duplicati della stessa richiesta).
Appena il server ottiene la rappresentazione della risorsa richiesta, la invia al client in un nuovo messaggio CON avente quindi un diverso Message ID ma lo stesso token della richiesta originaria. Tale tipo di risposta è detta separate. Con l'invio della risposta separate in un messaggio di tipo CON il server avvia a sua volta un timer di ritrasmissione allo scadere del quale in assenza di riscontro procederà al reinvio della risposta secondo un algoritmo prestabilito. Alla ricezione della risposta CON il client deve a sua volta inviare un riscontro ACK vuoto al server (un riscontro che non trasporti né una richiesta né una risposta). In Figura 52 sono illustrate due risposte separate a due richieste di tipo CON. In particolare nella Figura

52(a) è mostrato il caso in cui il primo ACK vuoto del server alla richiesta del client non giunge a destinazione: il client deve essere consapevole del fatto che qualora l'ACK vada perso può ricevere una risposta CON alla sua richiesta senza riscontro oppure può riceverlo dopo, non essendo garantito l'ordine di recapito dei pacchetti dal protocollo UDP.

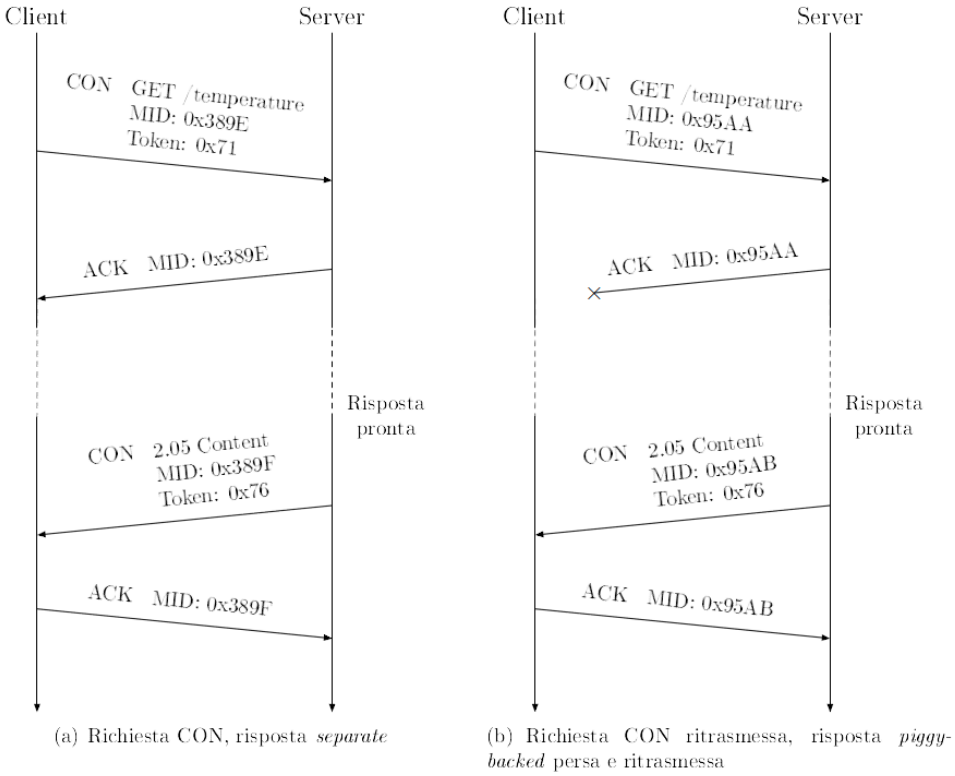


(a) Richiesta CON, risposta *piggy-backed*

(b) Richiesta CON persa e ritrasmessa, risposta *piggy-backed*

(c) Richiesta CON ritrasmessa, risposta *piggy-backed* persa e ritrasmessa

Figura 51. Richieste CoAP ti tipo CON con tecnica del piggy-backing.



(a) Richiesta CON, risposta *separate*

(b) Richiesta CON ritrasmessa, risposta *piggy-backed* persa e ritrasmessa

Figura 52. Richieste CoAP di tipo CON con risposte separate.

4.2.5 Implementazione di un caso studio con protocollo CoAP

Si può ora passare ad analizzare rapidamente il codice che è stato utilizzato e scritto col fine di raggiungere gli scopi sopra citati, sia per quanto riguarda l'aspetto implementativo della logica esecutiva del Local Gateway sia per ciò che riguarda l'aspetto funzionale del dispositivo IoT.
Partendo con l'analizzare la porzione di codice che governa e gestisce il corretto funzionamento architetturale descritto nella Figura 53, ci si può concentrare inizialmente sull'analizzare tale aspetto implementativo, dal punto di vista del Server CoAP, ovvero dal punto di vista dell'entità che all'interno dell'architettura della nostra rete, viene chiamata ed individuata col termine di Local Gateway:
la componente software più importante e principale che ritroviamo all'interno di questo nodo è sicuramente rappresentata dallo script Python server_put-coap.py

```python
import datetime
import asyncio
import aiocoap.resource as resource
import aiocoap
import requests
import json

class Sensori(resource.Resource):
    def __init__(self):
        super().__init__()
    async def render_put(self, request):
        print('DEBUG> PUT payload: %s' % request.payload)
        data=json.loads(request.payload)
        r = requests.post('http://http://iotprogettons.zapto.org/data/coap/1',data=data)
        print(r.text)
        return  aiocoap.Message(code=aiocoap.CHANGED)

def main():
    root = resource.Site()
    root.add_resource(['sensori'], Sensori())
    asyncio.Task(aiocoap.Context.create_server_context(root))
    asyncio.get_event_loop().run_forever()
if __name__ == "__main__":
    main()
```

Figura 53. Script Python server_put_coap.py

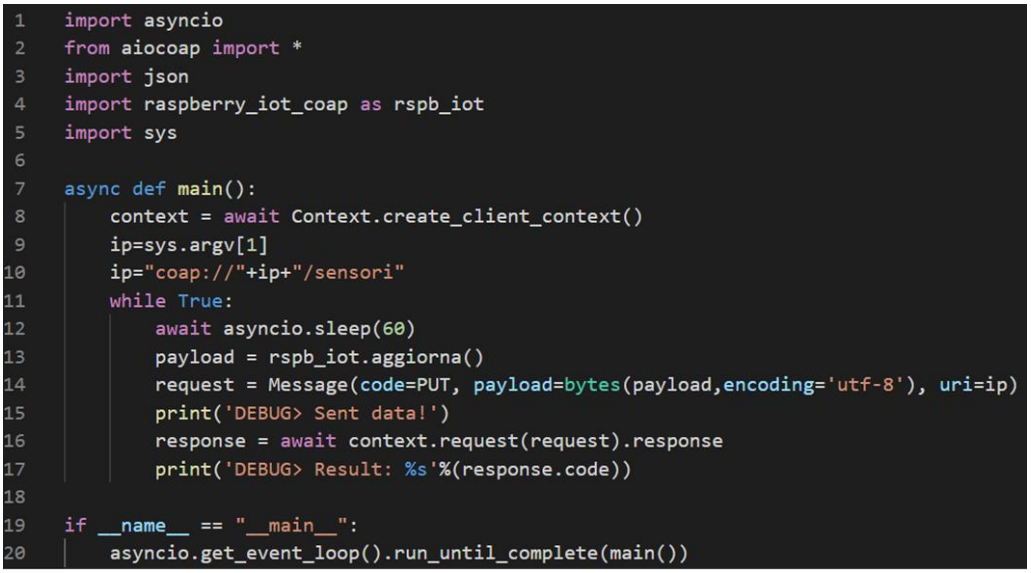
Come si può ben intuire analizzando e leggendo il codice, la situazione implementativa è molto semplice e diretta: infatti ciò che accade è che, inizialmente, il Server CoAP crea e definisce un Site, che può essere visto ed inteso come il "luogo fisico" all'interno del quale andrà ad inserire ed a nominare le risorse che vuole esporre verso l'esterno, rendendole raggiungibili ed "interagibili" con i Client CoAP presenti; una volta creato e realizzato tale spazio di risorse, viene aggiunta, tramite il metodo add_resource(…), una prima risorsa, che è la risorsa 'sensori', risorsa che viene realizzata a livello implementativo tramite la costruzione di una istanza di una classe omonima, che come si può ben vedere implementa un solo metodo (in overriding), chiamato render_put(…), che rappresenta la sequenza di istruzioni ed operazioni che saranno "triggerate" ed eseguite, quando quella risorsa riceverà una richiesta CoAP di tipo PUT; ciò che tale metodo va a compiere, molto semplicemente è quello di ricevere la stringa rappresentativa il JSON contenente i dati inviati dal dispositivo IoT, convertirla in JSON, ed inviare tale file di riferimento, all'interno del campo body di una richiesta POST HTTP, verso il Server Remoto che si occuperà successivamente di aggiornare i suoi dati interni in merito a quelli ricevuti. Fatto ciò, infine, il Server CoAP, tramite l'invocazione del metodo get_event_loop().run_forever(), si mette in ascolto per gestire la ricezione di richieste CoAP asincrone, al fine di reinoltrarle verso le differenti risorse esposte coinvolte.

Andando invece a spostare l'attenzione su ciò che il CoAP Client ovvero il dispositivo IoT, compie, la situazione implementativa è ovviamente leggermente differente: difatti troviamo in questo caso due differenti blocchi di codice di interesse da analizzare, il primo presente all'interno dello script nominato raspberry_iot_coap.py mentre il secondo posizionato all'interno del successivo script client_put-coap.py. Se si analizza più nel dettaglio il primo dei due script, si può notare, molto semplicemente, come al suo interno risulta essere definito un'unica funzione, aggiorna(), che si occupa di interagire col sensore presente e associato al Raspberry, al fine di ottenere i dati ambientali correnti misurati e da lui estratti, per poi incapsularli in un file di tipo JSON ed infine restituirli alla funzione chiamante sotto forma di stringa alphanumerica. Tale JSON generato, avrà esattamente la seguente struttura:

```
{
 "temperatura": "32",
 "data": "06-05-2020 00:10",
 "pressione_barometro": "99955",
 "temperatura_off": "25",
 "umidita": "28",
 "luce": "3",
 "temperatura_barometro":
 "33",
 "uomo": "Live body detected within 5 seconds!"
}
```

Figura 54. File JSON

La componente software che però permette effettivamente al dispositivo IoT di essere visto all'interno della rete e quindi di funzionare come un Client CoAP, così da poter entrare in contatto con la componente Server precedentemente introdotta, è situata ed implementata all'interno del file di scripting client_put-coap.py:



```
import asyncio
from aiocoap import *
import json
import raspberry_iot_coap as rspb_iot
import sys

async def main():
    context = await Context.create_client_context()
    ip=sys.argv[1]
    ip="coap://"+ip+"/sensori"
    while True:
        await asyncio.sleep(60)
        payload = rspb_iot.aggiorna()
        request = Message(code=PUT, payload=bytes(payload,encoding='utf-8'), uri=ip)
        print('DEBUG> Sent data!')
        response = await context.request(request).response
        print('DEBUG> Result: %s'%(response.code))

if __name__ == "__main__":
    asyncio.get_event_loop().run_until_complete(main())
```

Figura 55. Script Python client_put_coap.py

Come si può facilmente intuire leggendo il codice sovrastante, quello che il dispositivo IoT, visto come Client CoAP, esegue a livello operazionale ed applicativo, è molto semplice ed intuitivo da comprendere: difatti esso non fa altro che andare a definire l'URL di riferimento, rappresentante la risorsa a cui il Client vuole richiedere l'accesso (coap://<ip_local_gateway>/sensori), per poi successivamente, tramite un loop infinito, andare a generare ogni 60 secondi (sleep(60)) una nuova richiesta di tipo PUT, il quale payload che sarà inoltrato al suo interno, sarà ottenuto dal metodo

aggiorna() citato in precedenza, che permette di ottenere i valori dei dati ambientali correnti a partire da quelli recepiti dal sensore associato al dispositivo IoT.
Terminata questa analisi prettamente tecnica ed implementativa del codice Python legata al funzionamento pratico del modello architetturale associato alla Figura 56, possiamo ora passare a concentrarci sull'analizzare l'aspetto sempre tecnico e di sviluppo implementativo, relazionato però al modello architetturale definito nella successiva Figura 57.
Come accaduto nella precedente analisi, è conveniente partire col descrivere la struttura esecutiva ed implementativa del Server CoAP, andandosi a soffermare attraverso un rapido approfondimento, su ciò che concerne il suo codice applicativo interno; bisogna però tenere in considerazione come in questo caso, a differenza del precedente, il ruolo di Server CoAP non viene svolto dal Local Gateway, bensì dallo stesso dispositivo IoT considerato. Detto ciò, tralasciando il codice contenuto del file raspberry_iot_coap.py che è identico al caso precedentemente analizzato, è opportuno invece fare un rapido approfondimento sulla logica implementativa presente nel file client_observer-coap.py:

```
2  import asyncio
3  import aiocoap.resource as resource
4  import aiocoap
5  import json
6  import raspberry_iot_coap as rspb_iot
7
8  class DataResource(resource.ObservableResource):
9
10     data_state = bytes(rspb_iot.aggiorna(), encoding="ascii")
11     def __init__(self):
12         super(DataResource, self).__init__()
13         self.notify()
14     def notify(self):
15         self.data_state = bytes(rspb_iot.aggiorna(), encoding="ascii")
16         self.updated_state()
17         asyncio.get_event_loop().call_later(60, self.notify)
18     @asyncio.coroutine
19     def render_get(self, request):
20         #print('State of Data: %s', self.data_state)
21         return aiocoap.Message(code=aiocoap.CONTENT, payload=self.data_state)
22
23 def main():
24     root = resource.Site()
25     root.add_resource(['data'], DataResource())
26     asyncio.Task(aiocoap.Context.create_server_context(root))
27     asyncio.get_event_loop().run_forever()
28 if __name__ == "__main__":
29     main()
```

Figura 56. Script Python client_observer-coap.py

Come si può facilmente intuire, ciò che il Server CoAP va a compiere in questo caso, è molto similare a ciò che è già stato precedentemente descritto, se non con qualche piccola ma sostanziale differenza: difatti, come prima operazione, esso non fa altro che creare il site all'interno del quale poter definire ed esporre pubblicamente le proprie "risorse CoAP", per poi aggiungere in esso, una prima risorsa identificata col PATH '/data': tale DataResource però, a differenza della tipologia di risorsa analizzata in precedenza, come si può ben vedere leggendo la riga ove è presente la definizione dell'omonima classe, risulta essere ereditaria di una particolare tipologia di risorse, chiamate osservabili (ObservableResource);

ciò permette di capire come, in questo caso, oltre a fare l'overriding del metodo render_ get(…), che è il metodo che verrà eseguito quando la risorsa considerata riceverà una richiesta CoAP di tipo GET, bisognerà sovrascrivere anche il metodo notify(…), che non è altro che la sequenza di operazioni e di istruzioni che verrà svolta nel momento in cui quella risorsa subirà un cambiamento di stato diretto (permette di far cambiare periodicamente stato alla risorsa, per poi notificarlo a tutti gli Observable ad essa associati). Come si può facilmente leggere, la risorsa Subject in questione, avrà come stato interno, i valori corrispondenti ai dati ambientali correnti che, sottoforma di JSON, vengono ottenuti e prelevati per mezzo dei sensori associati, per poi, come descritto nel metodo notify, aggiornare tali valori in maniera asincrona, ogni 60 secondi, generando le diverse notifiche inoltrate poi ai differenti Observable.
Terminata questa rapida analisi per ciò che concerne la struttura esecutiva interna del Server CoAP, è conveniente ora spostare l'attenzione sull'altra entità di fondamentale importanza in questo contesto architetturale, che è il Client CoAP, questa volta rappresentato dal Local Gateway. In questo caso, il file contenente il codice di riferimento interessato, è nominato come server_observer-coap.py:

```python
import asyncio
from aiocoap import *
import json
import requests
import sys


if len(sys.argv) <= 1 :
    print("ERROR> Inserisci IP client COAP!")
    sys.exit(-1)

SERVER_URI  = 'coap://'+sys.argv[1]
def observe_handle(response):
    if response.code.is_successful():
        data=json.loads(response.payload)
        print(data)
        r = requests.post('http://iotprogettons.zapto.org/data/coap/1',data=data)
        print('DEBUG> Response Code: %s'%response.code)
        print(r.text)
    else:
        print('DEBUG> Error code %s' % response.code)

@asyncio.coroutine
def main():
    protocol = yield from Context.create_client_context()
    request = Message(code=GET)
    request.set_request_uri(SERVER_URI  + '/data')
    request.opt.observe = 0
    observation_is_over = asyncio.Future()
    try:
        requester = protocol.request(request)
        requester.observation.register_callback(observe_handle)
        response = yield from requester.response
        exit_reason = yield from observation_is_over
        print('DEBUG> Observation is over: %r' % exit_reason)
    finally:
        if not requester.response.done():
            requester.response.cancel()
        if not requester.observation.cancelled:
            requester.observation.cancel()
if __name__ == "__main__":
    asyncio.get_event_loop().run_until_complete(main())
```

Figura 57. Script Python server_observer-coap.py

Come si può effettivamente osservare dal codice sovrastante, ciò che il Local Gateway va a compiere, dal punto di vista di un Client CoAP, è quello di creare inizialmente una richiesta CoAP di tipo GET, selezionando come risorsa CoAP quella posizionata all'URL coap://<ip_dispositivo_iot>/data, e settando il campo Observe Option a 0, per indicare che quella richiesta di tipo GET, è una richiesta estesa di "registrazione e osservazione" nei confronti del subject 'data'.
Una volta fatto ciò, come ultimo passaggio da compiere, egli definisce, attraverso il metodo observation.register_callback(…), quella che sarà la funzione di callback da eseguire ogni volta in cui sarà ricevuto un messaggio di notifica inerente ad un cambiamento di stato relativo al subject precedentemente scelto: in questo caso, tale funzione di callback è denominata come observe_handle(…) e consente di estrarre dalla notifica ottenuta (response) le informazioni sui valori di stato del subject modificati, per poi inoltrarli, sempre come file JSON inserito all'interno del body di una POST HTTP, verso il Server Remoto di riferimento, che si occuperà successivamente di aggiornare i suoi dati interni in merito a quelli appena ricevuti.

4.3 Il protocollo MQTT

Il **Message Queue Telemetry Transport (MQTT)** è un protocollo ISO standard applicativo di messaggistica, nato proprio con lo scopo di favorire con semplicità e leggerezza, lo scambio di messaggi tra dispositivi potenzialmente ristretti e limitati al fine di minimizzare il traffico sulla rete e richiedere ben poche risorse computazionali e di memorizzazione ai loro componenti coinvolti, utilizzato sopra il protocollo TCP/IP. Tale protocollo, oltre a migliorare l'aspetto computazionale e di impiego delle risorse fisiche indotte a gestire uno scambio di messaggi tra dispositivi "constrained", permette anche di eseguire e performare in maniera efficiente la distribuzione di messaggi, da un mittente verso molti destinatari, secondo una politica "one to many". Per favorire tutto ciò, MQTT segue un paradigma implementativo differente da quello "client/server" di CoAP, che si basa invece sul meccanismo di *pubblicazione* e *sottoscrizione* asincrono ad un **Topic**, conosciuto anche col nome di *publish* and *subscribe*. L'idea di base di questo paradigma è molto semplice: un nodo A nella rete decide di pubblicare un messaggio (publish) relativo ad un generico Topic, e tale messaggio sarà ricevuto da tutti i nodi B che sono iscritti (subscribe) a tale Topic di riferimento. Inoltre il tutto viene gestito da una particolare entità, interna o esterna alla rete, chiamata **Broker**: il suo compito non è altro che quello di ricevere i messaggi dai produttori degli stessi e renderli disponibili verso gli utilizzatori, consegnando quindi il messaggio soltanto per i Topic (ovvero gli argomenti, ad esempio la temperatura) sottoscritti dal ricevente.

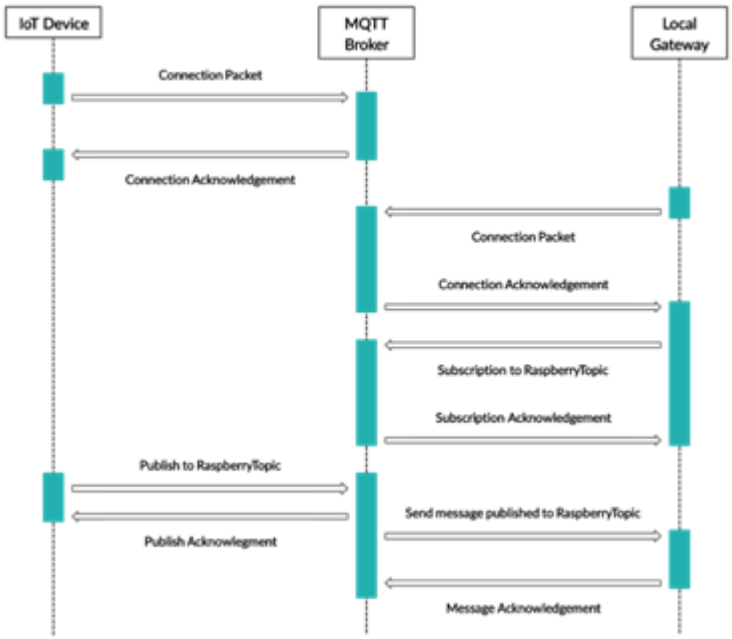


Figura 58. IoT Network-LAN-SeqDiag-MQTT

Prima però di andare ad analizzare le feature che maggiormente caratterizzano il protocollo appena descritto, è opportuno, tramite appositi diagrammi di sequenza, analizzare quella che è stata la logica funzionale da noi adottata all'interno della struttura di rete precedentemente introdotta, al fine di garantire la corretta realizzazione dello stesso protocollo.
Come si può osservare dalla Figura 58, il funzionamento applicativo di tale protocollo, all'interno della nostra architettura esecutiva di riferimento, è molto semplice ed intuitivo: difatti abbiamo da una parte, i due **MQTT Client**, che sono rappresentati in questo caso, sia dal **dispositivo IoT** che dal **Local Gateway**; dalla parte opposta invece, emerge la figura di un **Broker** esterno, intento a gestire le relazioni di interesse e di comunicazione tra i due client coinvolti. In tutto ciò, si può compiere poi un'ulteriore suddivisione di ruoli tra i rispettivi e già definiti client coinvolti: difatti il dispositivo IoT può essere considerato, nel rispetto dell'architettura di base del protocollo MQTT, come un semplice publisher, mentre il Local

Gateway, dall'altra parte, può essere classicamente associato al ruolo di subscriber. Una volta capito come i ruoli protocollari sono distribuiti tra le diverse entità presenti nella nostra rete, il meccanismo di comunicazione pratico che permette al dispositivo IoT di inoltrare messaggi (come per esempio i dati ambientali di riferimento) verso il Local Gateway è il seguente: in primo luogo, sia il dispositivo IoT, che il Local Gateway, stabiliscono una connessione diretta e persistente con il Broker di riferimento considerato; una volta che la connessione ambo i lati viene correttamente stabilita, il Local Gateway decide di sottoscriversi, tramite un messaggio di subscribing, ad un particolare Topic di riferimento (es. RaspberryTopic), notificando tale scelta al Broker con cui precedentemente era entrata in contatto. Dall'altra parte il Broker, ricevendo tale richiesta di sottoscrizione per quel determinato Topic, salverà e mapperà tutte le info necessarie a riconoscere il canale comunicativo stabilito col Local Gateway, con il nome identificato del Topic, così che, appena avverrà una nuova pubblicazione in funzione di esso, potrà inoltrare senza problemi il messaggio convenuto verso le entità sottoscritte a quel determinato Topic. Infine, il dispositivo IoT che precedentemente aveva stabilito un canale comunicativo col Broker, deciderà di inviare i dati estratti correntemente dal sensore, attraverso la publish di un messaggio nei confronti del Topic a cui il Local Gateway si è precedentemente iscritto (es. RaspberryTopic); allora, a conclusione del tutto, il Broker, una volta ricevuto il messaggio associato al Topic scelto, lo estrarrà e lo andrà a reinoltrare verso tutte le entità sottoscritte e attive rispetto quel Topic di riferimento (in questo caso, essendo il RaspberryTopic, solo il Local Gateway).
Terminata questa descrizione puramente architetturale e protocollare sulla struttura interna e generale della nostra LAN "privata" basata sull'utilizzo di MQTT, si può passare ad analizzare e ad approfondire maggiormente nel dettaglio gli aspetti implementativi e tecnici che ne hanno permesso una sua corretta realizzazione. Al fine di raggiungere lo scopo implementativo appena citato, anche qui, come è accaduto nel caso di CoAP, le componenti software che sono state progettate e sviluppate con l'obiettivo di governare l'intera logica applicativa interna sia del Local Gateway che del dispositivo IoT sono state completamente realizzate utilizzando come linguaggio di programmazione di base Python, integrando successivamente la versione basilare di quest'ultimo con un insieme di moduli e librerie di terze parti necessarie a gestire la corretta realizzazione dei diversi passi protocollari per MQTT. Al fine di realizzare ciò, è stato utilizzato un particolare package chiamato e conosciuto col termine di *paho-mqtt*, che risulta essere attualmente una delle principali e migliori librerie Python che gestiscono la corretta implementazione dei dettagli applicativi delle versioni 3.1 e 3.1.1 dello stesso protocollo MQTT.

4.3.1 Architettura del protocollo MQTT

Dal punto di vista architetturale MQTT è un protocollo message-oriented, in cui interagiscono due diverse tipologie di componenti: i **client** ed il **broker**. Ogni dispositivo che intende comunicare con un altro sfruttando il protocollo può essere visto come un client connesso via TCP ad un broker. Il compito di quest'ultimo è quello di distribuire i messaggi ricevuti su uno specifico argomento ai clienti che si erano sottoscritti a quel topic.
In Figura 59 è mostrata l'architettura del protocollo.

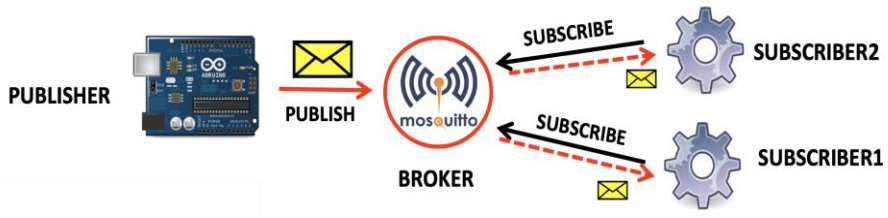


Figura 59. Architettura MQTT

Affinché un broker sia conforme alla specifica deve soddisfare le seguenti affermazioni:

1. il formato di tutti i pacchetti di controllo che il broker invia deve corrispondere ai formati descritti dal protocollo;
2. segue le regole di matching dei topic descritte dal protocollo;
3. soddisfa tutti i requisiti di livello MUST definiti nella specifica, fatta eccezione per quelli relativi ai client;

Le implementazioni di broker MQTT sono numerose, ognuna con caratteristiche diverse. Alcune di esse realizzano anche funzionalità al disopra dello standard. Tra tutte una tra le più popolari è *Mosquitto*, che recentemente è entrato a far parte dei progetti della Eclipse Foundation. Questo broker è open source, compatibile sia con MQTT che con MQTT-SN (MQTT for Sensor Networks), è leggero, scritto in C – in modo tale da poter essere eseguito anche in ambienti che non hanno le risorse necessarie per eseguire una Java Virtual Machine – ed è uno dei broker più semplici da configurare. L'attuale implementazione di Mosquitto ha un eseguibile di circa 120kB, ed un consumo di appena 3MB di RAM con 1000 client connessi. Offre inoltre la possibilità di fungere da bridge verso altri broker MQTT –comprese altre istanze di Mosquitto– consentendo la creazione di veri e propri network di broker MQTT. Ogni messaggio all'interno del protocollo è una porzione discreta di dati totalmente opaca per il broker. Questi dati verranno pubblicati ad un certo indirizzo, chiamato topic, su cui i clienti possono effettuare sottoscrizioni. Tutti i messaggi pubblicati su un certo topic verranno recapitati solo ai client che si erano precedentemente registrati ad esso. Il pattern publish-subscribe consente di disaccoppiare nello spazio mittente e destinatario dei messaggi: il publisher consegnerà il suo messaggio al broker, sarà quest'ultimo ad occuparsi della consegna a tutti i clienti che risultano connessi e che hanno sottoscritto lo specifico topic su cui il mittente ha deciso di pubblicare il messaggio, senza la necessità che il produttore li conosca

4.3.2 Caratteristiche del protocollo MQTT

MQTT, nonostante sia un protocollo leggero e adatto a lavorare in condizioni limite, riesce ad offrire una serie di funzionalità che gli hanno permesso di affermarsi come uno degli standard di riferimento per quanto riguarda l'IoT, tra cui:

- Topic matching
- Last Will and Testament (serve ad informare altri client di un client disconnesso)
- Persistenza
- Sicurezza

4.3.3 I messaggi del protocollo MQTT

La specifica MQTT è suddivisa in tre sezioni: la prima analizza le tipologie di pacchetti, la seconda definisce i dettagli di ogni tipologia di pacchetto mentre la terza descrive come i pacchetti vengono trasferiti tra client e server. Il formato del messaggio MQTT è rappresentato in Figura 60, invece in Figura 61 viene rappresentata la comunicazione tra publisher e broker.

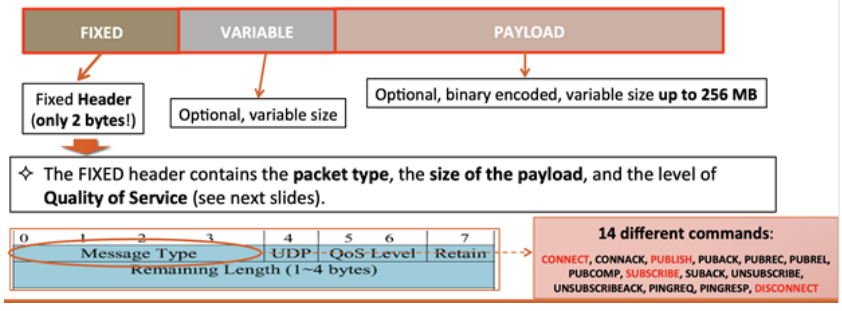


Figura 60. Formato del messaggio e differenti comandi

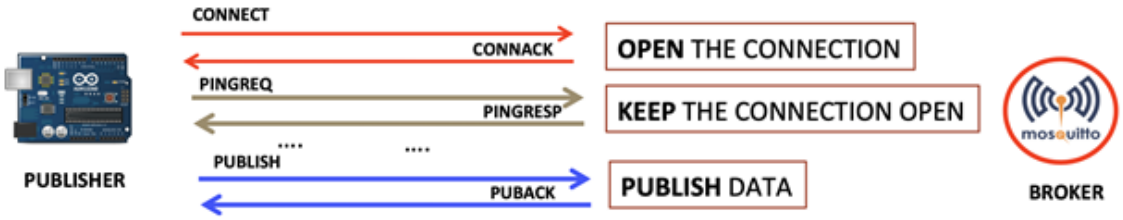


Figura 61. Comunicazione tra publisher e broker in MQTT

4.3.3.1 Sintassi

Come detto MQTT è un protocollo message-oriented, questo comporta che ogni interazione tra client e broker avverrà mediante messaggi, ognuno dei quali avrà un header di dimensione fissa, uno di dimensione variabile, ed il payload.

L'header fisso è lungo due byte e segue lo schema riportato in Figura 62:

bit	7	6	5	4	3	2	1	0
byte 1	Message Type				DUP flag	QoS level		RETAIN
byte 2	Remaining Length							

Figura 62. Fixed header

Message Type definirà il tipo di comando che si andrà ad inviare con il messaggio secondo l'enumerazione mostrata nella Tabella 7.

Tabella 7 – Tipi di messaggi

Mnemonic	Enumeration	Description
Reserved	0	Reserved
CONNECT	1	Client request to connect to Server
CONNACK	2	Connect Acknowledgment
PUBLISH	3	Publish message
PUBACK	4	Publish Acknowledgment
PUBREC	5	Publish Received (assured delivery part 1)
PUBREL	6	Publish Release (assured delivery part 2)
PUBCOMP	7	Publish Completed (assured delivery part 3)
SUBSCRIBE	8	Client Subscribe request
SUBACK	9	Subscribe Acknowledgment
UNSUBSCRIBE	10	Client Unsubscribe request
UNSUBACK	11	Unsubscribe Acknowledgment
PINGREQ	12	PING Request
PINGRESP	13	PING Response
DISCONNECT	14	Client is Disconnecting
Reserved	15	Reserved

DUP rappresenta un flag con cui è possibile specificare se il messaggio che si sta inviando è un duplicato di un precedente pacchetto di tipo PUBLISH, PUBREL, SUBSCRIBE o UNSUBSCRIBE andato perso o non confermato dal relativo ACK. Per avere un acknowledge della consegna da parte del broker il messaggio deve essere trasmesso con livello di QoS maggiore di 0, così come il suo duplicato.

QoS specifica il livello di qualità di servizio richiesta per un messaggio di tipo PUBLISH. Il campo può assumere valore da 0 a 2 come mostrato in Figura 63:

QoS value	bit 2	bit 1	Description		
0	0	0	At most once	Fire and Forget	<=1
1	0	1	At least once	Acknowledged delivery	>=1
2	1	0	Exactly once	Assured delivery	=1
3	1	1	Reserved		

Figura 63. Quality of Service

RETAIN specifica se il messaggio di tipo PUBLISH deve essere mantenuto in memoria persistente e consegnato ai clienti che si sottoscriveranno al topic in un secondo momento (valore 1), o se la consegna deve avvenire solo a chi è sottoscritto al momento della pubblicazione. Un messaggio con flag RETAIN settato rimane disponibile anche al riavvio del broker.

Il secondo byte, occupato per intero dal campo Remaining length specifica il numero di byte residui all'interno del messaggio, ossia la dimensione totale dell'header variabile e del payload.

L'**header variabile** non è una prerogativa di tutti i messaggi MQTT, ma solo alcuni di essi che lo utilizzeranno per fornire informazioni aggiuntive. Questa porzione di messaggio, quando presente, è sempre compresa tra l'header fisso ed il payload; conterrà campi ben definiti dalla specifica che saranno presenti nel seguente ordine:

Protocol Name è presente nei messaggi CONNECT, specifica il nome del protocollo (MQIsdp) codificato in UTF.

Protocol Version è composto da 8 bit. Specifica la versione di protocollo utilizzata in un messaggio di tipo CONNECT

Flag per messaggi CONNECT ha dimensione 8 bit e contiene una serie di flag che consentono di specificare la presenza di particolari parametri di connessione all'interno del messaggio. La struttura di questo campo è mostrata in Figura 64:

bit	7	6	5	4	3	2	1	0
	User Name Flag	Password Flag	Will Retain	Will QoS		Will Flag	Clean Session	Reserved

Figura 64. Connect flag

Clean Session se non è impostato, il broker conserverà le sottoscrizioni dei client dopo la loro disconnessione, in questo modo potrà ristabilire tutte le sottoscrizioni precedenti alla discon- nessione in modo automatico. Oltre alle sottoscrizioni il broker memorizzerà anche i messaggi ricevuti su quel topic dopo la disconnessione e, quando il client tornerà online provvederà a recapitarglieli.
Will specifica la presenza del messaggio *Last Will*. Se questo flag è settato devono essere presenti anche i flag Will QoS e Will Retain, il payload, inoltre, deve contenere il Will Topic ed il Will Message.
Will QoS definisce il livello di QoS per il messaggio Will
Will Retain indica al broker se deve mantenere in memoria persistente il messaggio Will

User Name e **Password** specificano se nel payload sono presenti nome utente e password che il client utilizzerà per autenticarsi.

Keep Alive timer anch'esso è disponibile per messaggi di tipo CONNECT. È un campo di lunghezza 16 bit che definisce l'intervallo massimo tra i messaggi del client, trascorso il quale viene riconosciuto come disconnesso. Si misura in secondi, e può avere un valore massimo di 18 ore. Per rimanere connesso il client dovrà inviare un PINGREQ, a cui il risponderà con PINGRESP. Trascorso il Keep Alive il broker disconnetterà il client.

Connect return code contiene il codice che viene restituito nell'header variabile di un messaggio di tipo CONNACK. Ha lunghezza 1 byte e può assumere i seguenti valori:

- 0: connessione accettata
- 1: protocolli incompatibili
- 2: identificativo respinto
- 3: server non disponibile
- 4: autenticazione fallita
- 5: autorizzazione negata
- 6-255: non ancora in uso

Topic name: è un campo disponibile (ed obbligatorio) per i messaggi PUBLISH. Contiene l'identificati- vo del canale d'informazione su cui dovrà essere pubblicato il payload. È codificato in UTF.

Il **payload** è l'effettivo contenuto del messaggio. Anche questa porzione di messaggio, come l'header variabile, non è disponibile per tutte le tipologie di messaggi, ma solo per le seguenti:

PUBLISH contiene informazioni application-specific.

CONNECT è formato da una stringa UTF-8 obbligatoria (l'identificativo del client) ed altre che conterranno, se I relativi flag sono settati ad 1, nome utente, password, Will Topic e Will Message.

SUBSCRIBE in questo caso il payload è formato da una lista di topic a cui il cliente intende sottoscriversi, specificando per ognuno di essi il livello di QoS desiderato.

SUBACK contiene la lista di livelli QoS che gli amministratori del broker consentono di utilizzare.

4.3.3.2 L'identificatore di messaggio

Alle porzioni precedenti, per quanto riguarda messaggi di tipo PUBLISH, PUBACK, PUBREC, PUBREL, PUBCOMP, SUBSCRIBE, SUBACK, UNSUBSCRIBE, UNSUBACK, è obbligatorio, nel caso in cui QoS sia maggiore di 0, inserire anche il Message ID. Questo campo –di lunghezza 16 bit– ha il compito di identificare in maniera univoca ogni messaggio trasmesso da un client in una determinata direzione: un messaggio trasmesso ed uno ricevuto, anche se presentano lo stesso ID sono considerati totalmente differenti.

4.3.3.3 L'ordine dei messaggi

La consegna ordinata dei messaggi può essere condizionata da numerosi fattori, tra cui il numero di messaggi che possono lasciare il client senza che l'arrivo dei precedenti al broker sia confermato (in-flight messages), o dal fatto che il client sia single o multi-thread. Per avere certezza che i messaggi vengano ricevuti in ordine è necessario attendere che la consegna sia portata a termine, aspettando il PUBACK o il PUBREL da parte del broker. Il numero di messaggi simultanei ha comunque effetto sulla garanzia che si può ottenere: se è possibile inviare un solo messaggio alla volta ogni flusso dovrà essere completato prima di avviare il successivo, quindi si avrà la garanzia che l'ordine sia rispettato, mentre nel caso in cui si inviino più messaggi simultaneamente si potrà fare affidamento solo sul QoS.

4.3.4 Livelli di Qualità di Servizio

MQTT consente ai client di specificare al broker il livello di qualità di servizio per ogni messaggio. Questo consente di definire come i messaggi dovranno essere consegnati. È fondamentale definire propriamente il livello di QoS, poiché determina come client e broker interagiranno. I livelli definiti dalla specifica sono tre:

Livello 0 - At most once: La consegna avviene secondo la strategia best effort del protocollo TCP/IP, vedi Figura 65.

Figura 65. **QoS 0: At Most Once**

Livello 1 - At least once: Garantisce che il messaggio venga consegnato almeno una volta, ma non esclude che ci siano duplicati, Figura 66.

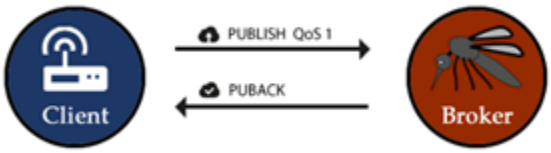


Figura 66. **QoS1: At Least Once**

Livello 2 - Exactly once: È il livello di QoS maggiore e assicura che i messaggi vengano consegnati una ed una sola volta, Figura 67.

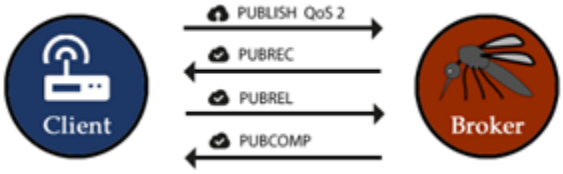


Figura 67. QoS2: Exactly Once

4.3.5 Implementazione di un caso studio con protocollo MQTT

A conclusione di ciò, si può passare ad analizzare rapidamente il codice che è stato utilizzato e scritto col fine di raggiungere gli scopi sopra citati, sia per quanto riguarda l'aspetto implementativo della logica esecutiva del Local Gateway sia per ciò che riguarda l'aspetto funzionale del dispositivo IoT e del Broker.
Concentrandosi inizialmente per quanto riguarda la figura del Broker, per semplicità implementativa e di utilizzo, si è preferito optare per sfruttare i servizi appositi che un server come Test Mosquitto metteva noi a disposizione; proprio per questo abbiamo deciso di scegliere come servizio di "broking", il seguente:
Hostname: test.mosquitto.org
Port: 1883 (per MQTT unencrypted)

Detto ciò, si può ora passare ad analizzare più nel dettaglio, l'aspetto implementativo presente all'interno dell'entità publisher presente all'interno della nostra rete locale, ovvero il dispositivo IoT. In questo caso, sono due le componenti software fondamentali che gestiscono e governano la logica esecutiva di tale nodo: da una parte troviamo lo script denominato raspberry_iot_mqtt.py mentre dall'altro emerge lo script client_publisher_mqtt.py. In realtà, da un punto di vista prettamente pratico, tali due componenti software sono strettamente legate e correlate tra loro.

```
1  import paho.mqtt.client as mqtt
2  import time
3  import datetime
4
5  def init(c_id):
6      client = mqtt.Client(client_id=c_id, clean_session=True, userdata=None, transport="tcp")
7      return client
8
9  def sending(client,br_url,br_port,msg,topic):
10     client.connect(br_url, br_port,60)
11     client.publish(topic=topic, payload=msg, qos=1, retain=False)
```

Figura 68. IoT Network-LAN-MQTT-Publisher1

```
import smbus
import client_publisher_mqtt as pb_mqtt
import json
from datetime import datetime
import time
import sys

client_id="ClientRaspberry"
topic="TopicRaspberry"
if len(sys.argv) > 2:
    client_id=sys.argv[1]
    topic=sys.argv[2]

print("DEBUG> ClientID: "+client_id)
print("DEBUG> Topic: "+topic)

client=pb_mqtt.init(client_id)
<CODE>
...
pb_mqtt.sending(client,"test.mosquitto.org",1883,json.dumps(js,sort_keys=True),topic)
print("DEBUG> Sent data!")
time.sleep(60)
```

Figura 69 IoT Network-LAN-MQTT-Publisher2

Come si evince dalle due figure, Figura 68 e Figura 69, quello che il dispositivo IoT va a compiere in veste di publisher è molto semplice ed intuitivo: egli non fa altro che, inizialmente, istanziare una nuova istanza di mqtt.Client, creando quindi un oggetto Client attraverso il quale interagire col Broker, definendo persino al suo interno l'ID attraverso il quale vuole risultare essere riconosciuto dal Broker stesso; successivamente, per tramite del metodo sending(…), sfruttando l'oggetto client precedentemente definito, non fa altro che, prima istaurare una connessione con il Broker posto all'indirizzo test.mosquitto.org sulla porta 1883, e poi, sfrutta tale canale appena creato, per eseguire una publish in riferimento al Topic chiamato come TopicRaspberry; il messaggio che sarà inoltrato tramite la publish, non sarà altro che la stringa rappresentativa del JSON, contenente tutti i differenti dati percepiti ed estratti dal sensore; tutto ciò verrà infine ripetuto periodicamente ogni sessanta secondi (nella parte in verde "<CODE>" è presente tutta quella porzione di codice che consente di estrapolare i dati dal sensore e di incapsularli in un JSON).

Terminata l'analisi per ciò che concerne il ruolo del publisher, per concludere, è opportuno andare ad analizzare la logica applicativa ed esecutiva del subscriber, ovvero del Local Gateway, vedi Figura 70.

```python
import paho.mqtt.client as mqtt
import json
import requests
import sys

serverUbuntu="ServerUbuntu"
topic="TopicRaspberry"
if len(sys.argv) <= 2 :
    print("DEFAULT> Server: "+serverUbuntu)
    print("DEFAULT> Topic : "+topic)
else:
    serverUbuntu=sys.argv[1]
    topic=sys.argv[2]
    print("DEBUG> Server: "+serverUbuntu)
    print("DEBUG> Topic : "+topic)

def on_message(client, userdata, message):
    print("DEBUG> Message Recieved: "+message.payload.decode())
    data=json.loads(message.payload)
    r = requests.post('http://http://iotprogettons.zapto.org/data/mqtt/1',data=data)
    print("DEBUG> Sent data!")


def init(c_id,br_url,br_port):
    broker_url = br_url#"test.mosquitto.org"
    broker_port = br_port#1883
    client = mqtt.Client(client_id=c_id, clean_session=True, userdata=None, transport="tcp")
    client.on_connect = on_connect
    client.on_message = on_message
    client.connect(broker_url, broker_port,60)
    return client

def subscribing(client,topic):
    client.subscribe(topic, qos=1)
    client.loop_forever()

client=init(serverUbuntu,"test.mosquitto.org",1883)
subscribing(client,topic)
```

Figura 70. IoT Network-LAN-MQTT-Subcriber

Anche qui, in maniera analoga al caso precedentemente analizzato, avviene prima la creazione di un oggetto di tipo Client, al quale viene assegnato sempre un ID identificativo, affinché tale entità possa essere riconosciuta dal Broker; successivamente, per tramite di tale oggetto, viene istanziato un canale comunicativo col Broker, sempre al solito indirizzo test.mosquitto.org sulla porta 1883, e infine viene eseguita, tramite l'ausilio del metodo subscribe(…), la sottoscrizione al Topic TopicRaspberry. Da notare come, il metodo on_message(…) viene sovrascritto in quanto è fondamentale per il corretto funzionamento di tutto il sistema, poiché è il metodo che viene invocato ogni qualvolta il subscriber riceva una nuova pubblicazione in riferimento al Topic sottoscritto: difatti si può notare come, in questo caso, ogni volta che viene ricevuto un nuovo messaggio proveniente dal Topic TopicRaspberry, il Local Gateway non fa altro che inoltrarlo, tramite l'ausilio del protocollo HTTP, verso il Server Remoto.

4.4 Valutazione di performance protocolli MQTT e CoAP

In questa sezione viene fornita una valutazione delle prestazioni dei protocolli di comunicazione IoT, MQTT e CoAP. Lo studio presenta un confronto delle prestazioni di MQTT e CoAP utilizzando un middleware comune che supporta MQTT e CoAP. Gli esperimenti vengono utilizzati per analizzare le prestazioni di MQTT e CoAP in termini di ritardo end-to-end e consumo di larghezza di banda. Sulla base delle analisi condotte in [18], presenteremo e analizzeremo l'efficienza, l'utilizzo e i requisiti di MQTT e CoAP utilizzando un Raspberry-Pi e una serie di dati energetici secondo quanto previsto dal progetto ComESTO.

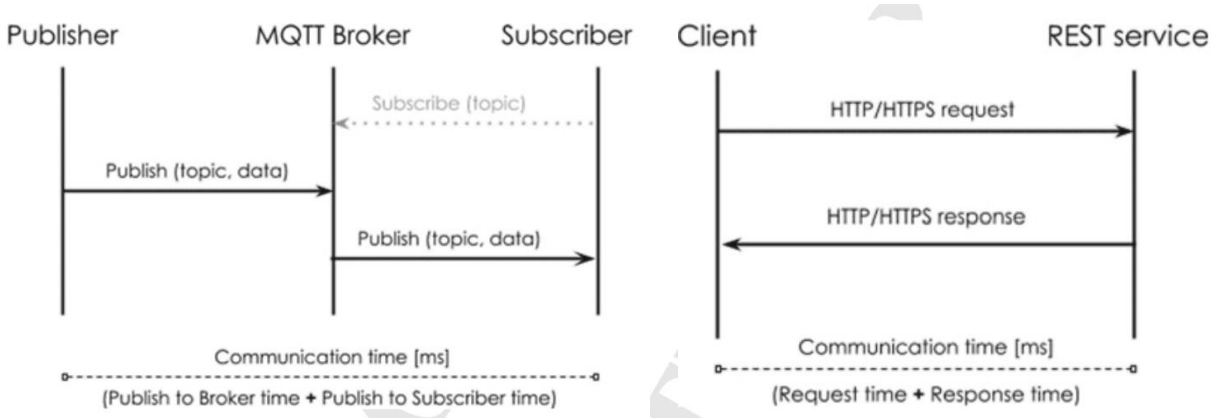


Figura 71. MQTT Publish/Subscribe timing and REST Request/Response timing

L'analisi dei tempi di comunicazione per il protocollo MQTT è legata al pattern di pubblicazione/sottoscrizione event-driven. Nei nostri test, il tempo di comunicazione è dato da Publish to Broker time + Publish to Subscriber time in millisecondi (Figura 71).
Nel classico paradigma di richiesta/risposta HTTP, utilizzato nell'approccio REST/CoAP, il tempo di comunicazione è dato da Tempo di richiesta + Tempo di risposta in millisecondi (Figura 71).
Per tutti i test di comunicazione abbiamo utilizzato lo stesso messaggio JSON che contiene:

- ID messaggio
- Timestamp
- Payload di lunghezza variabile (10, 50 o 100 byte).

L'analisi dei tempi di comunicazione del protocollo MQTT è stata effettuata per ogni livello di QoS MQTT. Il livello QoS definisce la garanzia di consegna dei messaggi MQTT:

- Al massimo una volta (0): consegna migliore senza garanzia di consegna
- Almeno una volta (1): un messaggio viene consegnato almeno una volta ai destinatari
- Esattamente una volta (2): un messaggio viene consegnato esattamente una volta ai destinatari.

Diversi livelli di QoS producono diversi tempi di comunicazione, come mostrato in Figura 72.

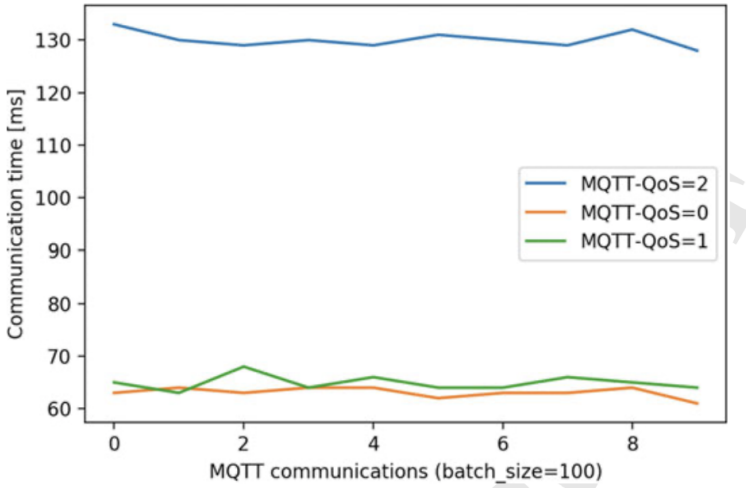


Figura 72. MQTT communication time for different QoS

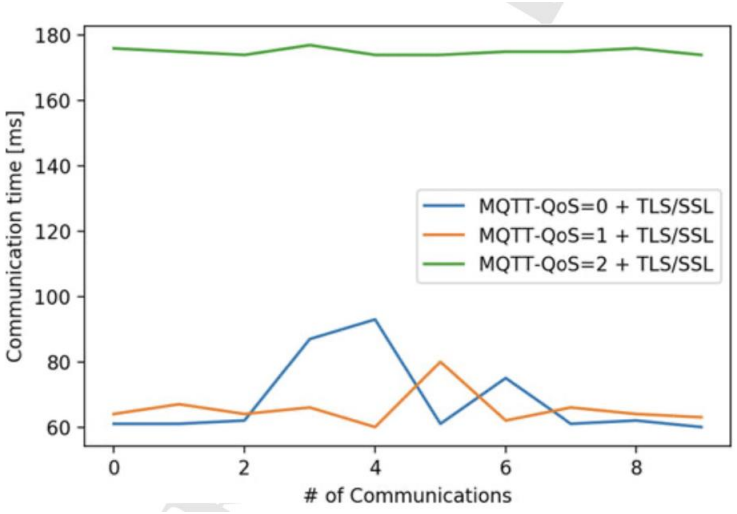


Figura 73. MQTT communication time for different QoS with TLS security

Per questo test sono state utilizzate 1000 comunicazioni MQTT per ogni livello di QoS. I dati sono stati raggruppati in 10 batch e ogni batch rappresenta il tempo medio di comunicazione di 100 messaggi MQTT consegnati.
Per i livelli QoS 0 e 1, ci sono tempi di comunicazione simili di circa 63 ms, per il livello QoS 2, il tempo medio di comunicazione è di circa 130 ms.
I risultati precedenti sono diversi applicando la QoS se viene applicato anche il Transport Layer Security (TLS) per fornire funzionalità di sicurezza. La sicurezza dell'IoT è una questione importante da affrontare. In questo capitolo abbiamo valutato le prestazioni di MQTT sotto TLS che è un protocollo di crittografia che consente una comunicazione sicura e crittografata a livello di trasporto tra client e server.
Per il livello QoS 0 e 1, ci sono tempi di comunicazione simili di circa 67 ms, per il livello QoS 2, il tempo medio di comunicazione è di circa 175 ms.

I risultati del tempo di comunicazione di MQTT con sicurezza TLS sono superiori ai risultati mostrati in Figura 72. Ciò è dovuto alla costruzione di un canale crittografato con lo scambio di chiavi e certificati di sicurezza tra client e server che consente di aumentare i livelli di sicurezza del protocollo MQTT (Figura 73).
Il risultato è influenzato dalla memorizzazione nella cache delle risposte del server che utilizzano la stessa connessione TLS stabilita. I risultati dei tempi di comunicazione MQTT sono confrontati con la richiesta REST nelle Figura 74 e Figura 75.
In particolare, sono stati effettuati test con 100 richieste REST HTTP e 100 richieste REST HTTPS raggruppate in 10 batch di 10 richieste rest ciascuno. Come illustrato nelle Figura 74 e Figura 75, i tempi di comunicazione delle richieste REST sono superiori ai tempi di comunicazione MQTT. In particolare, il tempo medio di comunicazione delle richieste REST HTTP è di circa 156 ms e la media delle richieste REST HTTPS è di circa 295 ms.

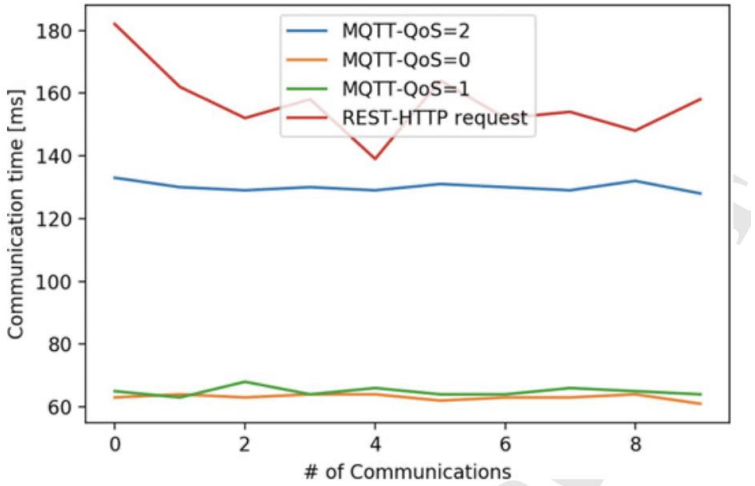


Figura 74. MQTT versus REST communication time

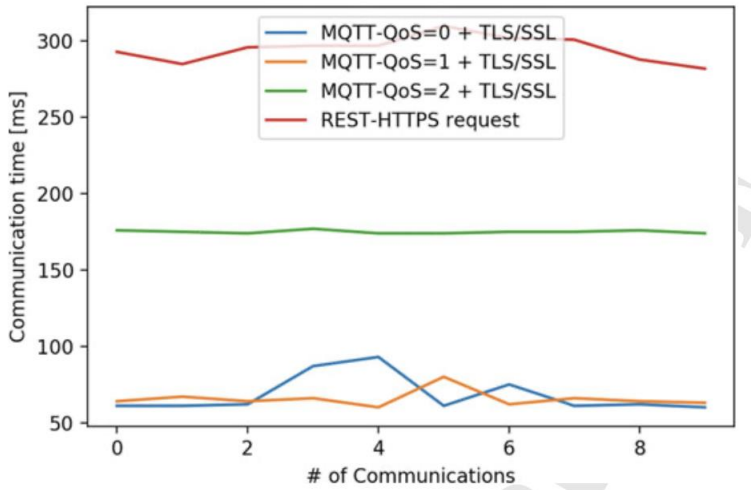


Figura 75 MQTT versus REST communication time with security

Questi risultati hanno mostrato che l'utilizzo di MQTT è migliore dell'utilizzo di riposo in termini di tempo di comunicazione con o senza sicurezza TLS per tutti i livelli di QoS. L'ambiente di test è stato lo stesso per tutti i test, il broker MQTT e i servizi

REST sono stati ospitati nella stessa macchina virtuale cloud (VM) e i client (editore MQTT, sottoscrittore MQTT e client REST) vengono eseguiti sulla stessa workstation di test.
L'analisi dei messaggi scambiati nelle comunicazioni per i diversi protocolli IoT utilizzati viene presentata in termini di dati scambiati e sovraccarico del protocollo.
Come illustrato in Figura 76, l'analisi dello scambio di messaggi per il protocollo MQTT è stata fornita per ciascun livello di QoS.

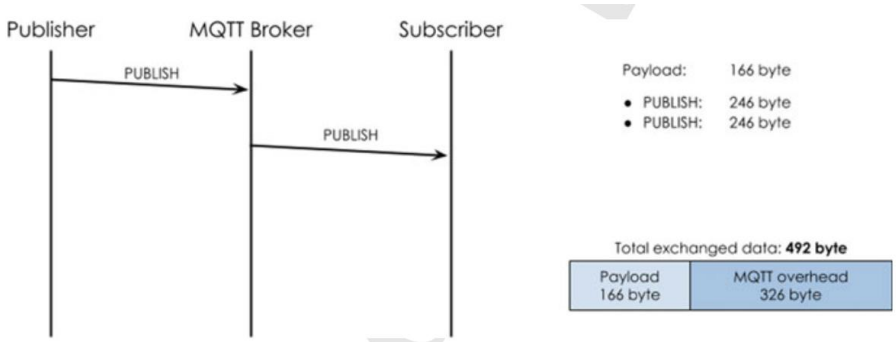


Figura 76. MQTT with QoS = 0 data exchange

Per un messaggio di 166 byte di payload, il totale dei dati scambiati è stato di 492 byte ma non è garantita la consegna. Per le comunicazioni MQTT con QoS = 1, un messaggio viene consegnato almeno una volta al destinatario. Come mostrato in Figura 77 per ogni messaggio pubblicato viene generato un ack, se un mittente non riceve questo riconoscimento invia nuovamente il messaggio, questo può generare più messaggi consegnati.

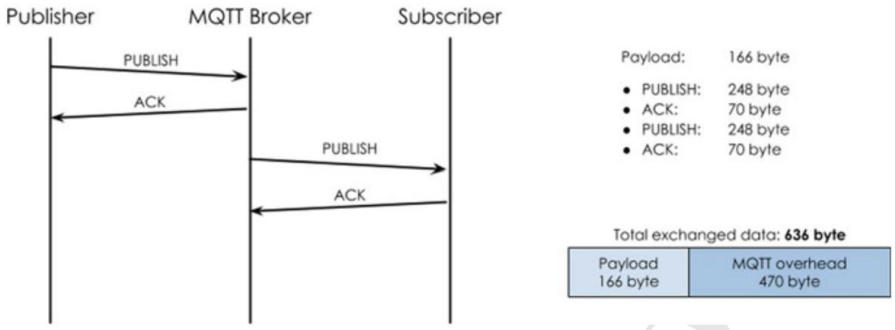


Figura 77. MQTT with QoS = 1 data exchange

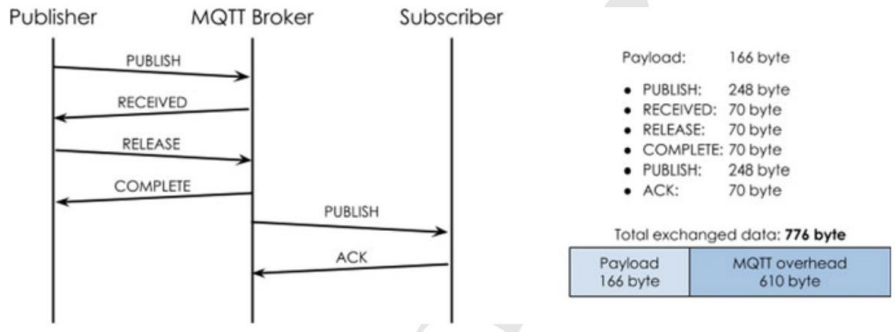


Figura 78. MQTT with QoS = 2 data exchange

I 166 byte di payload portano 636 byte di dati scambiati totali (senza più messaggi consegnati). La comunicazione MQTT con QoS = 2 è il livello QoS più sicuro e lento con una stretta di mano di quattro messaggi tra il mittente e il destinatario che confermano che il messaggio è stato inviato e che la conferma è stata ricevuta. Ciascun messaggio viene ricevuto una sola volta dai destinatari (Figura 78).

Per un messaggio con 166 byte di carico utile, il totale dei dati scambiati è stato di 776 byte. L'analisi dello scambio di dati per la richiesta HTTP/HTTPS classica è presentata di seguito.

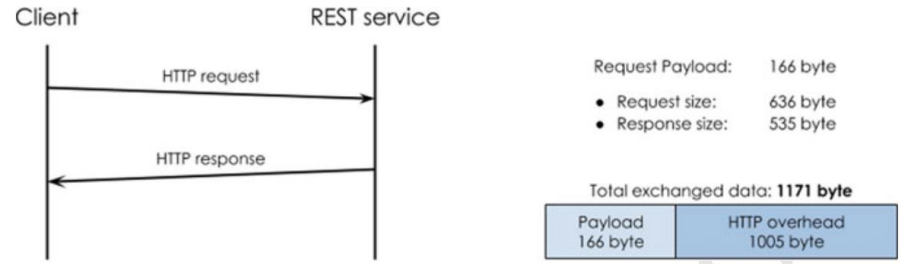


Figura 79. HTTP request data exchange

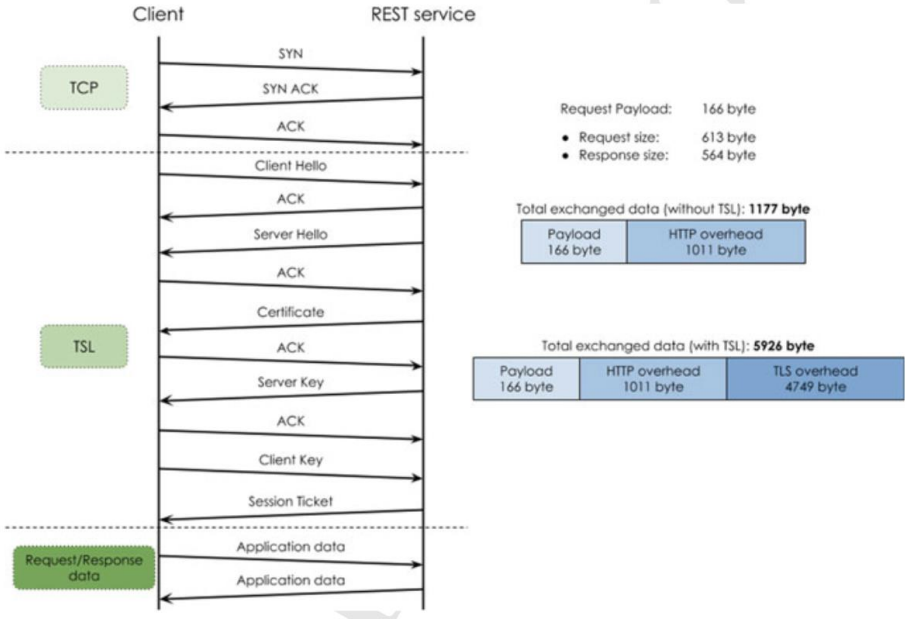


Figura 80 HTTPS request data exchange

Per una richiesta/risposta con uno scambio di 166 byte di dati di payload, ci sono 1171 byte di dati scambiati totali con un sovraccarico del protocollo di 1005 byte (Figura 79). Considerando una richiesta HTTPS, ci sono 1011 byte di sovraccarico del protocollo a cui devono essere aggiunti i dati scambiati per l'handshake TLS. Durante un handshake TLS, entrambe le parti comunicanti si scambiano messaggi per stabilire una comunicazione crittografata con crittografia asimmetrica e l'uso di una coppia di chiavi pubblica e privata. L'overhead TLS analizzato è pari a 4749 byte, in Figura 80 è rappresentato lo scambio dati completo della richiesta HTTPS. L'analisi dello scambio di dati per la richiesta/risposta CoAP è presentata di seguito per il messaggio CoAP confermabile (CON) e non confermabile (NO).

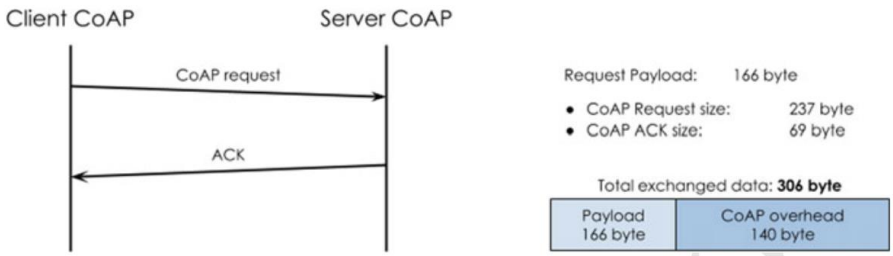


Figura 81. CoAP CON request data exchange

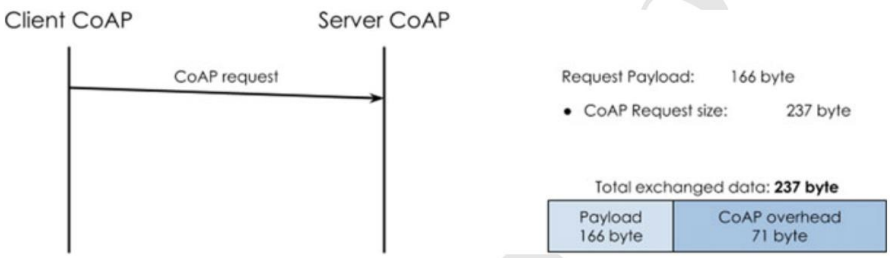


Figura 82. CoAP NON request data exchange

Nei messaggi Confermabili è richiesto un riconoscimento (ACK), per una richiesta di 166 byte ci sono 306 byte di dati scambiati totali con un sovraccarico del protocollo CoAP di 140 byte. La dimensione della richiesta CoAP è pari a 237 byte e la dimensione CoAP ACK è pari a 69 byte e contiene la risposta o il codice di errore (Figura 81).
Nei messaggi Non Confermabili non è richiesto alcun riconoscimento, per una richiesta di 166 byte si hanno 237 byte di dati scambiati complessivi (Figura 82).

Le principali differenze tra MQTT, CoAP e REST sono:

- MQTT utilizza un paradigma di pubblicazione/abbonato mentre CoAP e REST utilizzano un paradigma di richiesta/risposta
- MQTT è un protocollo molti-a-molti che utilizza un broker centrale per inviare i messaggi provenienti dall'editore ai sottoscrittori invece in CoAP e REST c'è una comunicazione client-server uno-a-uno
- MQTT è un protocollo orientato agli eventi mentre CoAP e REST sono più adatti per il trasferimento di stato
- CoAP e REST richiedono aggiornamenti sui cambiamenti di stato con richieste periodiche
- CoAP viene eseguito su UDP mentre MQTT e REST vengono eseguiti su TCP.

5 PROTOCOLLI MACHINE-TO-MACHINE A SUPPORTO DELLE NANOGRID PER LA PIATTAFORMA DEL DISTRIBUTORE

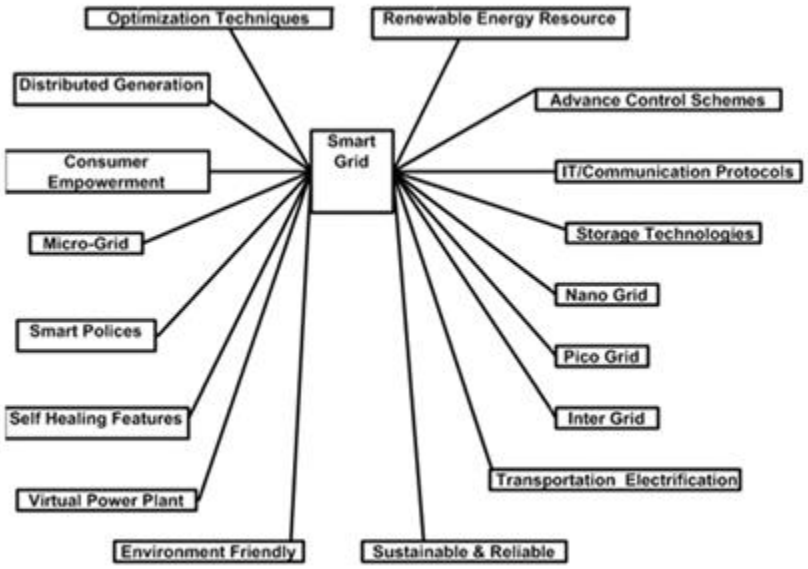


Figura 83. Generalized overview of the Smart Grid (SG).

L'evoluzione delle tecnologie dell'informazione e delle infrastrutture, così come le politiche governative, hanno un impatto sulla gestione delle reti elettriche. La crescita della domanda di energia elettrica, la responsabilità dei consumatori e le nuove tecnologie di comunicazione sono le cause principali per trasformare l'attuale rete elettrica in Smart Grid [70][71]. La rete elettrica convenzionale non è in grado di gestire le nuove tendenze e presenta la necessità di un adeguamento.

In Figura 83 viene presentata una panoramica generalizzata della Smart Grid e in Figura 84 un modello generale.

Il concetto di consumatori intelligenti può essere implementato nelle SG, migliorando le funzionalità Consumer Empowerment (CE), come DSM (Demand Side Management) e DRP (Demand Response Program). Il CE fornisce ai consumatori un modo per ottimizzare il loro profilo di utilizzo dell'energia in modo più efficace. La capacità decisionale immediata del consumatore e le funzioni di controllo diretto del carico portano al risparmio energetico. Il monitoraggio diretto e la visibilità della domanda con funzionalità DRP porteranno un impatto positivo sulla rete elettrica [72],[73].

Il modello architetturale delle Smart Grid prevede il Consumer Empowerment (CE) insieme al sistema di monitoring chiamato AMI (Advance Metering Infrastructure), vedi Figura 85.

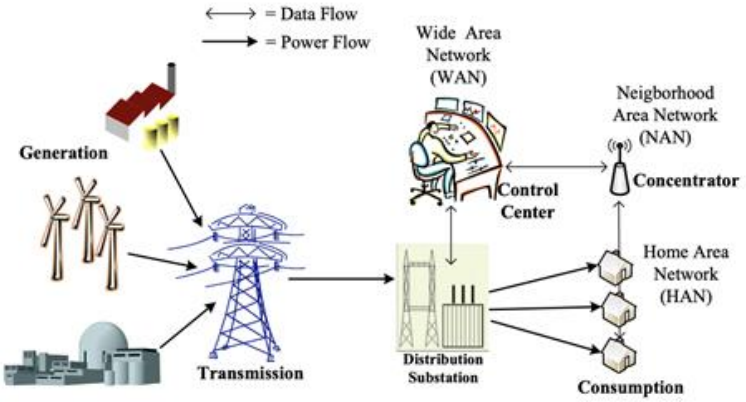


Figura 84. Modello generale di Smart Grid

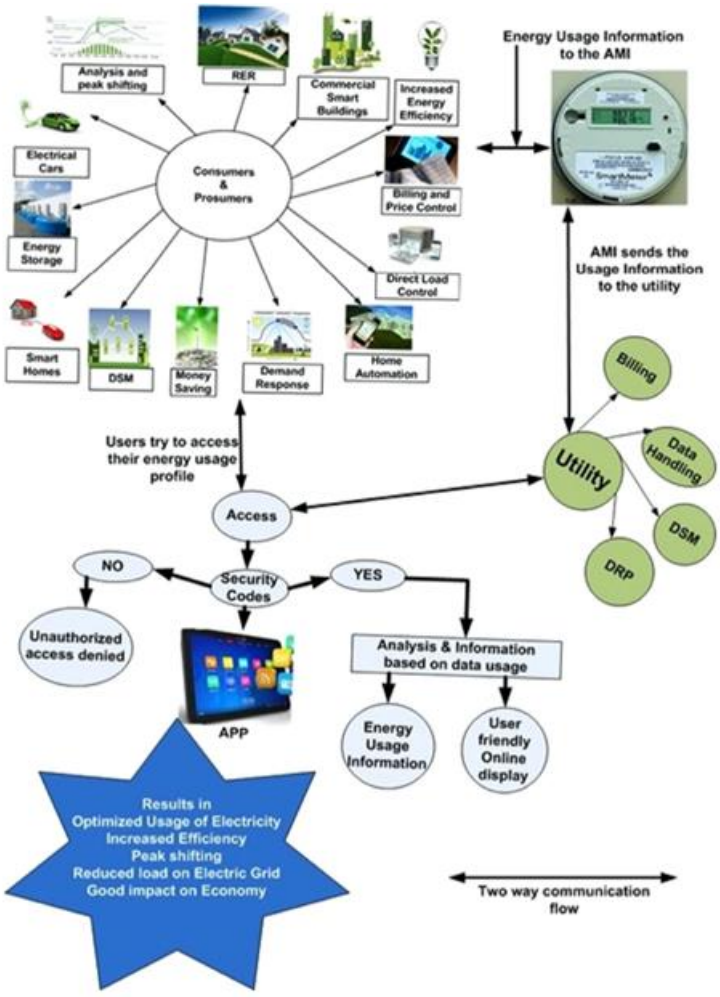


Figura 85. Modello architetturale di una SG con particolare attenzione al CE.

Il flusso bidirezionale di alimentazione e comunicazione aiuterà ad acquistare (consumer) e vendere (prosumer) energia elettrica da utility, come server e client [74]. Il server riceve una richiesta di fornitura di energia e, previa approvazione, il client riceverà energia. L'utilizzo di energia al momento giusto si tradurrà nell'interazione tra client e server. I consumatori possono supervisionare direttamente l'energia monitorandone il loro profilo di utilizzo [75]. La supervisione diretta del profilo di utilizzo dell'energia rafforzerà anche il potenziamento del sistema di distribuzione decentralizzato di prosumer e consumer. L'interazione con i prosumer è un'altra caratteristica interessante di CE che fornisce incentivi ai consumer sull'esportazione di energia elettrica prodotta in eccesso dai RER (Renewable Energy Resource), vedi Figura 86.

L'interazione tra consumatori e utenze risulta molto efficace con l'utilizzo di DSM. Il DSM nell'ambiente SG sarà in grado di soddisfare l'indicatore di sviluppo sociale e consentirà ai consumatori di utilizzare l'energia in modo efficiente. La struttura DSM migliorata aiuta le utility a monitorare, controllare e riparare la rete elettrica in tempo reale. La partecipazione attiva dei consumatori alla SG è potenzialmente possibile utilizzando un gran numero di sensori distribuiti, creando così un forte sistema di feedback. Questo sistema di feedback garantisce una risposta tempestiva nell'ambiente SG.

Figura 86. Integrazione delle Renewable Energy Resource (RER).

5.1.1 Tecnologie di comunicazione nelle Smart Grid

La gestione di enormi quantità di dati richiede una tecnologia di comunicazione sicura, affidabile e conveniente nelle SG. Le tecnologie di comunicazione consentono un flusso bidirezionale di informazioni tra le varie entità della SG. Il dibattito su "come e quale tecnologia di comunicazione dovrebbe essere implementata nella SG?" è ancora in corso [76]. Le tecnologie di comunicazione nelle SG sono classificate in due tipologie principali, Wireless e Wired. La Figura 87 presenta la classificazione delle tecnologie di comunicazione implementate nella SG. Il flusso di informazioni è classificato in due

categorie. Il primo flusso va dai sensori alle infrastrutture di misura. Il sistema ampiamente diffuso per questa connettività è il PLC (Power Line Communication) o la comunicazione wireless. Il secondo flusso di informazioni è dall'infrastruttura di misurazione al data center; ed è effettuato attraverso reti cellulari o Internet [77]. In questa sezione daremo un breve accenno alle tecnologie di comunicazione impiegate nelle SG.

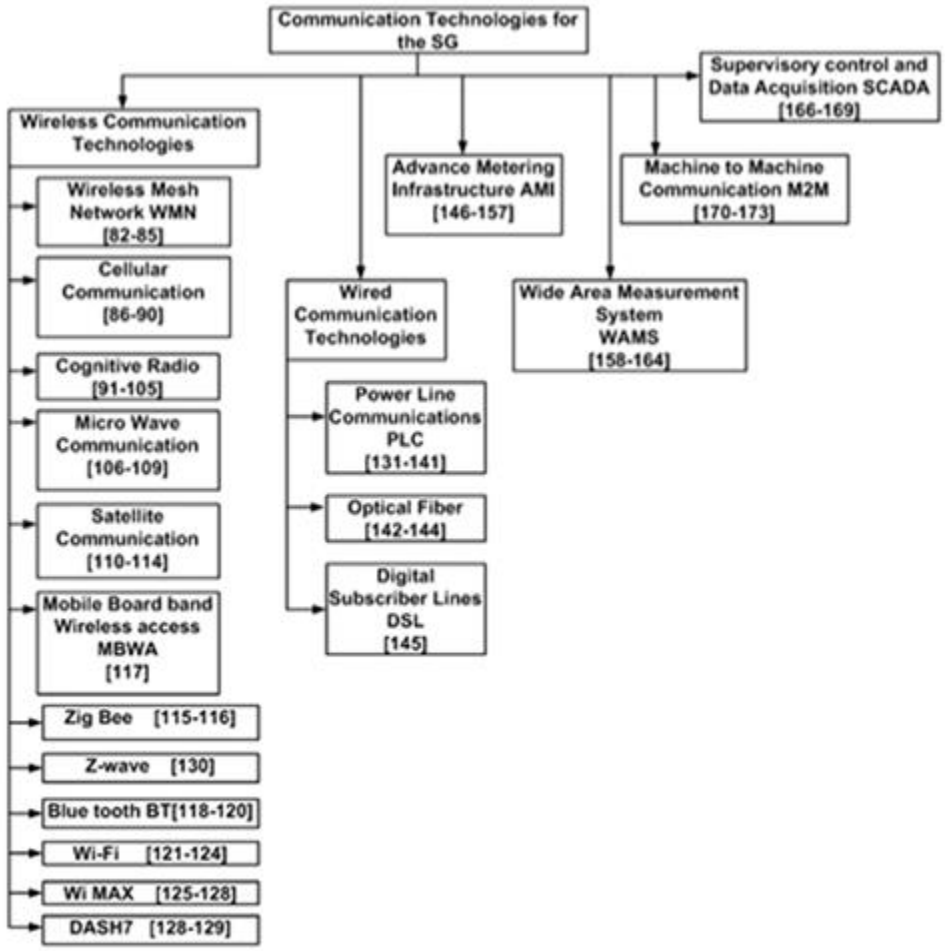


Figura 87. Classificazione delle tecnologie di comunicazione per le SG

5.1.1.1 Wireless Communication

I principali vantaggi della comunicazione wireless rispetto alla comunicazione cablata sono la mobilità, il facile rifornimento e il basso costo. La tecnologia wireless è la più appropriata per le applicazioni remote [78]. Esistono una serie di tecnologie che possono essere utilizzate per effettuare la comunicazione all'interno della SG tra i vari oggetti presenti, per elencarne qualcuna: Wireless Mesh Network (WMN), comunicazioni cellulari, tecnologia ZigBee, Bluetooth, WiFi, WiMAX, ecc.

5.1.1.2 Wired Communication

La tecnologia di comunicazione cablata (WCT) include: (a) comunicazioni su linea elettrica (PLC), (b) comunicazioni su fibra ottica (OFC) e (c) linee di abbonamento digitali (DSL).

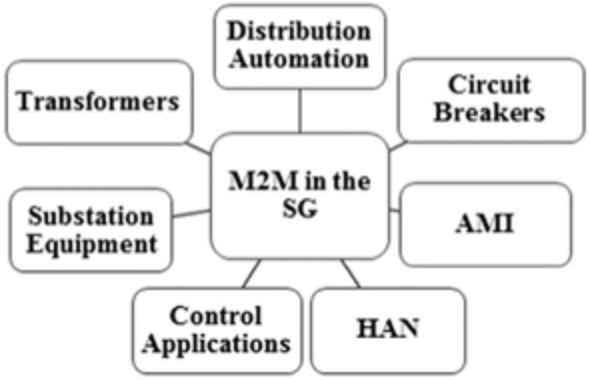


Figura 88. Aree chiave delle applicazioni M2M nelle SG.

5.1.2 Machine to Machine Communication (M2M)

La comunicazione dei dati senza interazione umana tra i dispositivi è chiamata comunicazione "M2M". Il M2M si è evoluto da SCADA negli anni '80 [79]. Le applicazioni M2M sono presenti anche in altri settori, come quello medico, industriale, della pubblica sicurezza e dei prodotti di consumo. Una delle migliori aree di crescita potenziale del M2M è nella SG. La Figura 88 mostra le aree chiave dell'applicazione M2M nelle SG [80].
M2M crea l'Internet delle cose (IoT) che viene definito "interconnessioni di dispositivi in rete". M2M stabilito dal sistema di comunicazione wireless avanzato è ampiamente preferito in tutto il mondo. Il Wi-Fi basato sullo standard IEEE 802.11 è il più interessante per l'implementazione di M2M grazie alla scalabilità e alla facilità di installazione. Lo standard IEEE 802.15.4 offre una soluzione conveniente, a basso consumo per stabilire M2M. Altre soluzioni esistenti per l'implementazione M2M sono IEEE 802.5.1, IEEE 802.15.3a e IrDA [81].

5.1.3 Machine to Machine Protocol

I protocolli Machine-to-Machine (M2M) hanno come obiettivo quello di permettere lo scambio di informazioni fra dispositivi in maniera totalmente autonoma, ovvero senza un coinvolgimento esplicito da parte dell'uomo (se non limitatamente ad operazioni di supervisione). In tale contesto l'accezione "macchina" va quindi intesa in antitesi alla parola "uomo", per riferirsi in genere a dispositivi (sensori, tag RFID, attuatori, ecc.) o anche parti di software (agenti). Questi protocolli costituiscono l'elemento fondante del paradigma dell'Internet of Things (IoT) [82] in base al quale in un prossimo futuro saranno connessi ad Internet anche oggetti comuni come documenti cartacei e contenitori per il cibo [83], [84].
Le comunicazioni M2M sono altresì alla base dei Cyber Physical Systems (CPS) ovvero di quei sistemi che vedono l'integrazione di sistemi di comunicazione, elaborazione e di controllo [85].
Machine-to-Machine (M2M), Internet of Things (IoT), Cyber Physical Systems (CPS), Industry 4.0, Industrial Internet, Internet of Everything, Big Data, sono tutti nomi di paradigmi, fra loro sovrapposti e interconnessi e di cui spesso si perdono i confini, che riflettono diversi sforzi e tecnologie mirate alla comunicazione fra il mondo dell'informazione (fatto da risorse

informatiche come database e software), il mondo cibernetico (fatto da macchine e dispositivi) e il mondo reale (fatto da uomini e oggetti di uso comune da parte dell'uomo stesso).

Nonostante i diversi possibili campi applicativi e le relative opportunità di business, notevoli sono i problemi ancora da affrontare in tale contesto legati principalmente a due fattori:

1. l'eterogeneità delle "macchine": i dispositivi M2M sono in genere dispositivi low-cost spesso basati su semplici microcontrollori e con notevoli differenze e limiti in termini di memoria, capacità di elaborazione, risorse energetiche, ecc.; in altri casi possono però essere dispositivi decisamente evoluti come smartphone e tablet di ultima generazione; non vi è quindi uniformità negli oggetti in termini di risorse hardware e software e ciò complica notevolmente la realizzazione delle applicazioni. Tutto ciò ha portato ad una notevole frammentazione del mercato;
2. la notevole differenza fra il traffico generato dall'uomo (human-based) e quello generato dalle macchine (machine-based): le comunicazioni human-based (fra uomini e macchine) sono in genere di tipo request-response, hanno cioè origine da una richiesta da parte dell'uomo, e sono spesso finalizzate al trasferimento di medie e grosse quantità di dati per singola richiesta (si pensi ad esempio al download di un file video) per cui i tempi di risposta possono essere ragionevolmente dell'ordine dei secondi. Viceversa, nelle comunicazioni M2M le comunicazioni scaturiscono da un evento e non da una richiesta, i tempi di risposta richiesti sono in genere dell'ordine dei milli-secondi (specie per applicazioni rivolte all'automazione) e le trasmissioni riguardano spesso piccole quantità di dati trasmessi in maniera sporadica.

Quest'ultima differenza permette di comprendere perché i protocolli comunemente utilizzati in Internet (TCP e HTTP) non siano idonei per le comunicazioni M2M. Da qui la necessità di nuovi protocolli e più in generale di nuovi paradigmi e modelli di comunicazioni specifici per M2M.
Gran parte delle soluzioni M2M oggi esistenti derivano dalle reti di sensori (WSN – Wireless Sensors Network) e dai sistemi di automazione industriale (SCADA). In tali contesti la comunicazione diretta M2M è però più una opzione che non la tipologia di comunicazione più rilevante. Infatti, la comunicazione fra dispositivi in tali contesti esiste, ma si limita spesso al semplice inoltro dell'informazione al fine ultimo di estendere la copertura della rete. In ambito SCADA e WSN, inoltre, tutte le comunicazioni sono dirette verso un unico punto di raccolta (il sink nelle WSN ovvero l'unità master in ambito SCADA) dove le informazioni sono elaborate per essere poi rese accessibili e/o per prendere decisioni centralizzate finalizzate all'attuazione.
Nell'accezione pura del M2M invece tutte le operazioni, comprese quelle di controllo e gestione, dovrebbero essere distribuite e lasciate alle stesse macchine, senza la necessità di un nodo centrale con funzioni di supervisione. Sebbene diversi siano stati gli sforzi in passato in tale direzione, molti dei sistemi M2M sviluppati sono "verticali", rivolti cioè ad applicazioni specifiche e spesso basati su protocolli proprietari e/o su architetture monolitiche e poco flessibili [82].
Per facilitare lo sviluppo di applicazioni M2M risulta quindi di fondamentale importanza la disponibilità di standard in grado di garantire interoperabilità fra i dispositivi e i servizi M2M.
Nel corso degli anni diversi sono stati gli organismi interessati e le attività di standardizzazione riferite al M2M. Fra le più rilevanti:

- **ETSI**: L'European Telecommunications Standards Institute (ETSI) ha fondato nel 2009 un comitato tecnico (TC) specifico per il M2M definendo una infrastruttura di riferimento standard per le applicazioni M2M [86] (la prima

release è del 2011 e una seconda del 2013). In particolare, alcuni scenari applicativi M2M nell'ambito degli smart metering sono descritti nel documento TR 102 898 [87];

- **3GPP**: sotto la sigla 3GPP (3rd Generation Partnership Project) sono raggruppati diversi organismi di standardizzazione per le telecomunicazioni (fra cui l'ETSI in Europa, ARIB/TTC in Giappone, ATIS negli Stati Uniti e CCSA in Cina). A tale partnership si devono i principali standard della telefonia cellulare (GSM, UMTS e LTE) oltre che una serie di specifiche tecniche (TS 22.X e TS 23.X) finalizzate all'integrazione delle comunicazioni M2M nelle reti cellulari. In particolare, con riferimento alla Release 13 e al documento TS 22.368 [88] sono analizzati i requisiti inerenti congestioni di rete, consumi energetici, indirizzamento, identificazione e sicurezza;
- **TIA**: Telecommunications Industry Association (TIA) è l'equivalente negli Stati Uniti dell'ETSI in Europa. Il comitato tecnico TR-50 ha identificato i requisiti delle comunicazioni M2M e sviluppato un protocollo per la comunicazione di eventi indipendente dal mezzo fisico di comunicazione [89];
- **IEEE**: L'IEEE (Institute of Electrical and Electronic Engineers) nell'ambito dello standard IEEE 802.16 (commercialmente noto come WiMax) ha pubblicato un emendamento (802.16p) che definisce una interfaccia radio a banda larga per comunicazioni M2M tramite WiMax.

5.1.4 Paradigmi di comunicazione M2M

Due sono i principali meccanismi di comunicazione M2M [90]:

- il primo meccanismo noto come **request-based**, basato sul paradigma request-response, prevede che sia un nodo o una applicazione (client) ad interrogare un altro nodo (server) il quale risponde alla richiesta ad esempio restituendo il valore di una grandezza fisica. L'interrogazione avviene in genere ciclicamente, con un meccanismo noto come polling o in alternativa mediante meccanismi simili alle query dei database.
- il secondo meccanismo noto come **event-based**, basato sul paradigma publish-subscribe, prevede che la comunicazione abbia origine da un evento (ad esempio la variazione di una grandezza fisica) a seguito del quale un nodo informa gli altri nodi interessati.

Il paradigma **request-response** è alla base delle architetture software denominate **RESTful** (ovvero REST-conformi). L'architettura **REST (Representational State Transfer)** fu ideata nel 2000 da Roy T. Fielding [91], a cui si deve anche il protocollo HTTP1.1, e prevede che una risorsa web su un server, identificata univocamente da un Uniform Resource Identifiers (URI) (e.s. http://myhost.com/...), possa essere richiamata da un client mediante appositi metodi identificabili con operazioni verbali (GET per ottenere informazioni su una risorsa, PUT per modificarla, POST per crearla e DELETE per eliminarla). Le richieste sono stateless ovvero senza stato per cui una singola richiesta deve contenere tutte le informazioni necessarie al server per eseguirla. Una risorsa può poi essere collegata ad un'altra tramite link permettendo così l'accesso ad altre risorse distribuite nel World Wide Web.

Tale meccanismo, sebbene semplice, flessibile e scalabile, risulta essere inefficiente per le comunicazioni M2M a causa dell'elevato numero di informazioni che devono essere scambiate fra client e server per singola richiesta, oltre che per la necessità di ripetere ciclicamente le richieste per sapere ad esempio quando una variabile assume un valore diverso. Ciò comporta un notevole spreco di risorse sia in termini di banda che di consumi energetici.

Una alternativa alle architetture RESTful per la realizzazione di servizi web è il SOAP [92]. Mediante l'uso del linguaggio XML il SOAP permette la definizione di servizi basati su una semantica più generale (ma anche più complessa). Tali servizi, come nel caso dell'architettura RESTful, risultano indipendenti dalla piattaforma e possono essere richiamati da nodi remoti indipendentemente dal sistema operativo o dal linguaggio di programmazione usati. Il vantaggio del linguaggio XML è quello di generare messaggi maggiormente comprensibili all'uomo (se confrontati con sequenze di bit) ma a scapito di un elevato

overhead, sia in termini di memoria che di risorse di elaborazione, per cui tale linguaggio non risulta idoneo per comunicazioni M2M.

Alcuni sforzi per ridurre tale problema hanno portato ad una versione binaria e compressa dell'XML nota come EXI [93].

Un meccanismo che permette di ridurre notevolmente il numero di messaggi scambiati è il meccanismo event-based basato sul **paradigma publish-subscribe**. Tale paradigma prevede che i client (detti subscriber) manifestino il loro interesse ad una determinata informazione/risorsa comunicandolo ad un particolare nodo detto broker; questa operazione di registrazione (detta subscription) avviene in genere una sola volta.

Quando un dispositivo o un server (più propriamente detto publisher) ha un evento/informazione da comunicare (ovvero da pubblicare, da cui il nome del paradigma) la comunica al broker il quale a sua volta avviserà i subscriber interessati trasmettendo l'informazione in multicast.

Il meccanismo **event-based** ha quindi diversi vantaggi:

1. i subscriber sono immediatamente avvertiti dell'evento non appena questo è stato prodotto riducendo al minimo il tempo per la disseminazione del dato;
2. viene eliminata la fase di richiesta, riducendo così i messaggi trasmessi e quindi i consumi energetici;
3. non occorre coordinare un polling da parte di più entità permettendo così una maggiore scalabilità del sistema;
4. il meccanismo risulta indipendente dalla topologia della rete, ciò consente di riposizionare i nodi e di gestire la loro eventuale mobilità più facilmente senza che siano necessarie informazioni di localizzazione.

BIBLIOGRAFIA

[1] “Le Tecnologie Abilitanti per l’Iot” Notiziario Tecnico Telecom Italia;
[2] www.3gpp.org
[3] B. Li, Z. Fei, and Y. Zhang, “UAV communications for 5G and beyond: Recent advances and future trends,” IEEE Internet Things J., vol. 6, no. 2, pp. 2241–2263, Apr. 2019.
[4] D. C. Nguyen et al., "6G Internet of Things: A Comprehensive Survey," in IEEE Internet of Things Journal, doi: 10.1109/JIOT.2021.3103320.
[5] W. Jiang, B. Han, M. A. Habibi, and H. D. Schotten, “The Road Towards 6G: A Comprehensive Survey,” IEEE Open Journal of the Communications Society, pp. 1–1, 2021.
[6] S. J. Nawaz, S. K. Sharma, S. Wyne, M. N. Patwary, and M. Asaduzzaman, “Quantum machine learning for 6G communication networks: State-of-the-art and vision for the future,” IEEE Access, vol. 7, pp. 46317–46350, 2019.
[7] A. Shahraki, M. Abbasi, M. J. Piran, M. Chen, and S. Cui, “A Comprehensive Survey on 6G Networks: Applications, Core Services, Enabling Technologies, and Future Challenges,” Jan. 2021, arXiv: 2101.12475.
[8] L. U. Khan, I. Yaqoob, M. Imran, Z. Han and C. S. Hong, "6G Wireless Systems: A Vision, Architectural Elements, and Future Directions," in IEEE Access, vol. 8, pp. 147029-147044, 2020, doi: 10.1109/ACCESS.2020.3015289.
[9] Shahraki, Amin & Abbasi, Mahmoud & Piran, Md & Chen, Mingzhe & Cui, Shuguang. (2021). A Comprehensive Survey on 6G Networks: Applications, Core Services, Enabling Technologies, and Future Challenges.
[10] W. Jiang, B. Han, M. A. Habibi and H. D. Schotten, "The Road Towards 6G: A Comprehensive Survey," in IEEE Open Journal of the Communications Society, vol. 2, pp. 334-366, 2021, doi: 10.1109/OJCOMS.2021.3057679.
[11] Seppo Borenius, Heikki Hämmäinen, Matti Lehtonen, Petri Ahokangas, Smart grid evolution and mobile communications—Scenarios on the Finnish power grid, Electric Power Systems Research, Volume 199,2021,107367, ISSN 0378-7796,https://doi.org/10.1016/j.epsr.2021.107367.
[12] F. Ilahi, S. Dutta, M. M. Hasan, S. Afrin Rumpa and A. K. M. Baki, "Development of a Novel UWB Antenna for 6G-IoT Based Smart Grid Device Monitoring System," 2021 International Conference on Green Energy, Computing and Sustainable Technology (GECOST), 2021, pp. 1-5, doi: 10.1109/GECOST52368.2021.9538785.
[13] Fadi Al-Turjman, Mohammad Abujubbeh, ‘IoT-enabled smart grid via SM: An overview’, Future Generation Computer Systems, Volume 96, 2019, Pages 579-590.
[14] P. Porambage, G. Gür, D. P. Moya Osorio, M. Livanage and M. Ylianttila, "6G Security Challenges and Potential Solutions," 2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), 2021, pp. 622-627, doi: 10.1109/EuCNC/6GSummit51104.2021.9482609.
[15] M. Tariq, M. Ali, F. Naeem and H. V. Poor, "Vulnerability Assessment of 6G-Enabled Smart Grid Cyber–Physical Systems," in IEEE Internet of Things Journal, vol. 8, no. 7, pp. 5468-5475, 1 April1, 2021, doi: 10.1109/JIOT.2020.3042090.

[16] J. Zhao et al., “Power system dynamic state estimation: Motivations, definitions, methodologies, and future work,” IEEE Trans. Power Syst., vol. 34, no. 4, pp. 3188–3198, Jul. 2019.
[17] S. Rasool et al., “Blockchain-enabled reliable osmotic computing for Cloud of Things: Applications and challenges,” IEEE Internet Things Mag., vol. 3, no. 2, pp. 63–67, Jun. 2020.
[18] R. Berthier, W. H. Sanders, and H. Khurana, “Intrusion detection for advanced metering infrastructures: Requirements and architectural directions,” in Proc. IEEE Int. Conf. Smart Grid Commun., 2010, pp. 350–355.
[19] S. Zonouz et al., “SCPSE: Security-oriented cyber-physical state estimation for power grid critical infrastructures,” IEEE Trans. Smart Grid, vol. 3, no. 4, pp. 1790–1799, Dec. 2012.
[20] M. Ayar et al., “Cyber-physical robust control framework for enhancing transient stability of smart grids,” IET Cyber Phys. Syst. Theory Appl., vol. 2, no. 4, pp. 198–206, 2017.
[21] L. Schenato, G. Barchi, D. Macii, R. Arghandeh, K. Poolla, and A. Von Meier, “Bayesian linear state estimation using smart meters and PMUs measurements in distribution grids,” in Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm), 2014, pp. 572–577.
[22] M. Ali, M. Adnan, and M. Tariq, “Optimum control strategies for short term load forecasting in smart grids,” Int. J. Elect. Power Energy Syst., vol. 113, pp. 792–806, Dec. 2019.
[23] M. B. Do Coutto Filho and J. C. S. de Souza, “Forecasting-aided state estimation—Part I: Panorama,” IEEE Trans. Power Syst., vol. 24, no. 4, pp. 1667–1677, Nov. 2009.
[24] C. Carquex, C. Rosenberg, and K. Bhattacharya, “State estimation in power distribution systems based on ensemble Kalman filtering,” IEEE Trans. Power Syst., vol. 33, no. 6, pp. 6600–6610, Nov. 2018.
[25] M. B. Mollah et al., “Blockchain for future smart grid: A comprehensive survey,” IEEE Internet Things J., early access, May 11, 2020, doi: 10.1109/JIOT.2020.2993601.
[26] F. Naeem, M. Tariq, and H. V. Poor, “SDN-enabled energy-efficient routing optimization framework for industrial Internet of Things,” IEEE Trans. Ind. Informat., early access, Jul. 3, 2020, doi: 10.1109/TII.2020.3006885.
[27] Farrell, S., Ed., *"Low-Power Wide Area Network (LPWAN) Overview"*, RFC 8376, doi: 10.17487/RFC8376, May 2018, <https://www.rfc-editor.org/info/rfc8376>.
[28] Lora Alliance, Technical Marketing Workgroup 1.0 *“LoRaWAN What is it? A technical overview of LoRA and LoRaWAN”*, November 2015, https://loraalliance.org/wp-content/uploads/2020/11/what-is-lorawan.pdf
[29] https://www.midomet.com/faq/lorawan-una-soluzione-per-internet-of-things
[30] A. Zourmand, A. L. Kun Hing, C. Wai Hung and M. AbdulRehman, *"Internet of Things (IoT) using LoRa technology,"* 2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS), 2019, pp. 324-330, doi: 10.1109/I2CACIS.2019.8825008.
[31] H. Huang et al., "A LoRa-Based Optimal Path Routing Algorithm for Smart Grid," 2018 12th International Conference on Sensing Technology (ICST), 2018, pp. 71-76, doi: 10.1109/ICSensT.2018.8603641.
[32] F. E. Soares e Silva, C. H. Barriquello, L. N. Canha, D. P. Bernardon and W. Seizo Hokama, "Deployment of LoRA WAN Network for Rural Smart Grid in Brazil," 2018 IEEE PES Transmission & Distribution Conference and Exhibition - Latin America (T&D-LA), 2018, pp. 1-5, doi: 10.1109/TDC-LA.2018.8511646.

[33] LoRa_IoT-Based_Architecture_for_Advanced_Metering_Infrastructure_in_Residential_Smart_Grid J. L. Gallardo, M. A. Ahmed and N. Jara, "LoRa IoT-Based Architecture for Advanced Metering Infrastructure in Residential Smart Grid," in IEEE Access, vol. 9, pp. 124295-124312, 2021, doi: 10.1109/ACCESS.2021.3110873.
[34] Gillerman, J., Falk, H., &Mackiewicz, R. (2005). Focus on IEC TC 57-power system reliability and profitability. IEEE Power and Energy Magazine, 3(4), 66-67.
[35] MCDONALD, J. Substationautomation. ied integration and availabilityof information. Power and Energy Magazine, IEEE 1, 2(March and April 2003), 22–31.
[36] KOSTIC, T., PREISS, O., AND FREI, C. Understanding and usingtheiec 61850: a case for meta-modelling. Computer Standards&Interfaces 27, 6 (June 2005), 679–695.
[37] MODBUS-IDA. Modbus application protocolspecificationv1.1b, December 2006. http://www.modbus-ida.org/.
[38] DNP USERS GROUP. The distributed network protocol. Website.http://www.dnp.org/.
[39] UCA INTERNATIONAL USERS GROUP, I. Introduction toUCA R version 2.0. Tech. rep., Institute of Electrical and Electronics Engineerings, Inc., November 1999.
[40] IEC 61850 StandardsWorking Group. Communication networks and systems in substations, part 1: introduction and overview. 2010.
[41] IEC 61850 StandardsWorking Group. Communication networks and systems in substations, part 2: glossary. 2010.
[42] IEC 61850 StandardsWorking Group. Communication networks and systems in substations, part 3: general requirements. 2010.
[43] IEEE. ” ieee 802.3 ’standard for ethernet’ marks 30 years of innovation and global marketgrowth” (press release). 2014.
[44] European FP7 Ideal Grid for All (IDE4L) project. http://ide4l:eu, 2013-2016.
[45] IEC 61850-1, “Communication networks and systems for power utility automation - part 1: Introduction and overview”, 2013.
[46] S.W. Blume, Electric Power Systems Basics for the Nonelectrical Professional. Piscataway: IEEE Press, 2007.
[47] IEC 61850-5, “Communication networks and systems for power utility automation - part 5: Communication requirements for functions and device models”, 2013.
[48] Z. Zhang, X. Huang, et al., “Modeling and simulation of data flow for VLAN-based communication in substations”, IEEE Systems Journal, vol. PP, pp. 1–12, 99 2015.
[49] IEC/TR 61850-90-4, “Communication networks and systems for power utility automation - part 90-4: Network engineering guidelines”, 2013.
[50] IEC 61850-7-1, “Communication networks and systems for power utility automation – part 7-1: Basic communication structure - principles and models”, 2011.
[51] IEC 61850-7-2, “Communication networks and systems for power utility automation – part 7-2: Basic information and communication structure - abstract communication service interface (ACSI)”, 2010.
[52] IEC 61850-7-4, “Communication networks and systems for power utility automation -part 7-4: Basic communication structure - compatiblelogicalnode classes and data objectclasses”, 2010.

[53] IEC 61850-9-2, “Communication networks and systems for power utility automation– part9-2: Specific communication service mapping (SCSM) - sampledvalues over ISO/IEC8802-3”, 2011.
[54] IEC/TS 61850-8-1, “Communication networks and systems for power utility automation - part 8-1: Specific communication service mapping (SCSM) - mappings to MMS (ISO 9506-1 and ISO 9506-2) and to ISO/IEC 8802-3”, 2011.
[55] IEC/TS 62351-6, “Power systems management and associated information exchange – data and communications security - part 6: Security for IEC 61850”, 2007.
[56] (Nov. 18, 2016). What does time synchronization mean for Sampled Values?, [Online]. Available: https://ideology.atlassian.net/wiki/spaces/AP/pages/50069508/What+does+time+synchronisation+mean+for+Sampled+Values.
[57] ISO, “ISO 9506-1:2003”, ISO, Aug. 2003.
[58] SISCO, “Overview and introduction to the Manufacturing Message Specification (MMS -revision 2)”, SISCO, 1995.
[59] J. T. Sørensen and M. G. Jaatun, “An analysis of the Manufacturing Messaging Specificationprotocol”, in Ubiquitous Intelligence and Computing, F. E. Sandnes,
[60] R. L. O’Fallon, D. A. Klas, et al., “IEC 61850 MMS SCADA network optimization forIEDs”, in DistribuTECH Conference, February 2011, Feb. 2011.
[61] IEC 61850 StandardsWorking Group. Communication networks and systems in substations - specific communication service mapping (SCSM), part 9-1, sampledvalues over serial unidirectionalmultidrop point to point link. 2010.
[62] IEC 61850 StandardsWorking Group. Communication networks and systems in substations - specific communication service mapping (SCSM), part 9-2, sampledvalues over ISO/IEC 8802-3. 2010.
[63] Qutaiba Ali and Basil Mahmood. Analysis and design of a guaranteedreal time performance enhanced industrial ethernet. Int. Arab J. e-Technol., 2(1):18–28, 2011.
[64] ThySensorCombinedsensor. http://www:thytronic:it/index:asp?menu=3&page=schedaProdotto&id=155. Accessed: 2020-07-20.
[65] European FP7 Ideal Grid for All (IDE4L) project. http://ide4l:eu, 2013-2016.
[66] David ME Ingram. Assessment of precision timing and real-time data networks for digitalsubstationautomation. 2013.
[67] Hammer, E., &Sivertsen, E. (2008). Analysis and implementation of the IEC 61850 standard (Master'sthesis, Technical University of Denmark, DTU, DK-2800 Kgs. Lyngby, Denmark).
[68] Mackiewicz, R. E. (2006, June). Overview of IEC 61850 and Benefits. In 2006 IEEE Power Engineering Society General Meeting (pp. 8-pp). IEEE.
[69] Shelby, Z., Hartke, K., & Bormann, C. (2014). The constrained application protocol (CoAP). RFC 7252.
[70] Shaukat, N., Ali, S. M., Mehmood, C. A., Khan, B., Jawad, M., Farid, U., ... & Majid, M. (2018). A survey on consumers empowerment, communication technologies, and renewable generation penetration within Smart Grid. Renewable and Sustainable Energy Reviews, 81, 1453-1475.
[71] Grijalva PS. Distributed intelligence architecture for smart grid; April 6 2010.
[72] Jing C. Study on the behaviors of electric power consumers in Smart Grid environment. In: Proceedings of the 5th international conference on electricity distribution, Paper CP0467. Shanghai, 5‑6 Sept; 2012.

[73] Roozbehani M, Dahleh M, Mitter S. Dynamic pricing and stabilization of supply and demand in modern electric power grids, Smart Grid Communications (Smart Grid Comm). In: Proceedings of the 1st IEEE international conference. Gaithersburg, MD; 2010.
[74] Fernandes RAS, Silva IN, Oleskovicz M. Identification of residential load profile in the Smart Grid context. Power and energy society general meeting. IEEE; 2010.
[75] Saad S. Electricity demand for South Korean residential sector. Energy Policy 2009; 37:5469‐74.
[76] Souryal M, Gentile C, Griffith D, Cypher D, Golmie N. A methodology to evaluate wireless technologies for the smart grid. IEEE Smart Grid Comm 2010; 10:356‐61.
[77] Wenpeng L, Sharp D, Lancashire S. Smart grid communication network capacity planning for utilities. In: Proceedings of IEEE PES, Transmission Distrib. conf. Expo; 2010.
[78] Parikh PP, Kanabar MG, Sidhu TS. Opportunities and challenges of wireless communication technologies for smart grid applications. IEEE Power Energy Soc General Meeting’ 10 2010:1‐7.
[79] Keith S et al., Guide to industrial control systems; May 2013, NIST special publication, Available at: ⟨nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST. SP.800-82r1.pdf⟩ [Accessed 2 June 2015].
[80] Siok KT. et al, M2M Communications in the Smart Grid: Applications, Standards, Enabling Technologies, and Research Challenges, International Journal of Digital Multimedia Broadcasting Volume; 2011, p. 8 (Article ID 289015).
[81] Niyato D, Xiao L, Wang P. Machine-to-machine communications for home energy management system in smart grid. IEEE Commun Mag 2011;49(4):53‐9.
[82] “The Internet of Things Architecture (IoT-A) - SOTA report on existing integration frameworks/architectures for WSN, RFID and other emerging IoT related Technologies”, Nicola Bui, 2011.
[83] “Conference Report Disruptive Civil Technologies”, in National Intelligence Council, 2008.
[84] J. Granjal, E. Monteiro e J. Sa Silva, “Security of Internet of Things: A Survey of Existing Protocols and Open Research Issues”, in IEEE Communications Surveys and Tutorials 17.3, 2015.
[85] I. Stojmenovic, “Machine-to-Machine Communications With In-Network Data Aggregation, Processing, and Actuation for Large-Scale Cyber-Physical Systems”, IEEE Internet of Things Journal, vol. 1, pp. 122-128, 2014.
[86] ETSI, “Machine-to-Machine communications (M2M); Functional Architecture”, 2013.
[87] ETSI, “Machine to Machine communications (M2M): Use cases of Automotive Applications in M2M capable networks”, 2013.
[88] ETSI, “Digital cellular telecommunications system (Phase 2+) (GSM) Universal Mobile Telecommunications System (UMTS); LTE; Service requirements for Machine Type Communications (MTC)”, 2016.
[89] “TR-50 - M2M - Smart Device Communications”, [Online]. Available: http://www.tiaonline.org/all-standards/committees/tr-50.
[90] P. Carlos e A. Ana, “Towards Efficient Mobile M2M Communications: Survey and Open Challenges”, Molecular Diversity Preservation International (MDPI), vol. 14, n. 10, pp. 19582-19608, 2014.
[91] R. T. Fielding, Architectural Styles and the Design of Network-based Software Architectures, UNIVERSITY OF CALIFORNIA, IRVINE, 2000.
[92] W3C, “SOAP Specifications”, 29 Marzo 2014. [Online]. Available: http://www.w3.org/TR/soap/.

[93] W3C, “EXI Specifications”, 11 Febbraio 2014. [Online]. Available: https://www.w3.org/TR/exi/.