

Creșterea eficienței energetice în clădirile din sectorul rezidențial și terțiar prin aplicații smart și IoT

Autori: Vladimir TANASIEV, Laura GHEORGHE

Cuprins

I.	INTRODUCERE	3
II.	CONTEXTUL ACTUAL.....	3
III.	STUDIU DE CAZ	6
IV.	APLICAȚIA SMART BUILDING CONTROLLER	13
V.	CONCLUZII.....	17

I. INTRODUCERE

Prezentul raport își propune să trateze modul în care dispozitivele și aplicațiile de tip smart își găsesc utilitatea în clădirile din sectorul rezidențial și terțiar. Rezultatele studiului au diseminate prin intermediul a cinci lucrări științifice ce au fost prezentate în cadrul conferințelor internaționale din domeniu.

II. CONTEXTUL ACTUAL

Sectorul rezidențial și terțiar este responsabil cu 39,68% din consumul total de energie primară în UE. Fiind sectorul cu cea mai mare cotă de energie are și cel mai mare potențial de economisire a energiei. Sectorul energetic, în care este inclus și consumul casnic și terțiar, generează aproximativ 37% din emisiile de gaze cu efect de seră (GES) din UE, reprezentând principala cauză a schimbărilor climatice și, în mare măsură, a poluării atmosferice.

Contextul științific. Conceptul de “smart city” este o temă relativ nouă, interdisciplinară, ce își propune să eficientizeze procese legate de: guvernare, planificare, sănătate, transport, mediu și energie. Particularizând, această problemă prezintă valențe multiple în domeniul ingineriei energetice. Cercetările în domeniu au urmărit dezvoltarea soluțiilor inteligente care să permită:

- a. Reducerea consumului de energie la utilizatorii finali (**clădiri inteligente**). În literatura de specialitate o parte din studii este dedicată modelelor matematice care aproximează cât mai fidel consumul de energie corespunzător unei clădiri exploatate după o anumită politică de operare. Soluția rezidă în utilizarea algoritmilor de tip black box. Printre cei mai performanți algoritmi care au oferit rezultate bune se numără rețelele neuronale artificiale (RNA).
- b. Sincronizarea consumului cu producția de energie (**management integrat**). În paradigma actuală, într-o rețea de energie electrică sau termică, producătorul furnizează consumatorului serviciul (adică puterea electrică sau termică) la fiecare moment. Deși variații sunt admise (mai importante pentru rețeaua termică decât pentru cea electrică), o problemă esențială pentru producător este echilibrarea producției și consumului, care necesită prognoza consumului pentru a stabili producerea necesarului și vânzarea/cumpărarea diferenței.
- c. Reducerea pierderilor asociate transportului și distribuției de energie (**smart grid**). Rețelele electrice inteligente reprezintă rețele care pot integra eficient surse regenerabile de energie, sisteme de stocare și acțiunile utilizatorilor conectați, atât din punct de vedere energetic cât și informațional. Conectarea la rețea se face asigurând eficiența economică a sistemului electroenergetic, reducerea pierderilor și creșterea siguranței în alimentarea cu energie electrică.
- d. Interpretarea datelor, identificarea patternurilor. Rețelele și clădirile inteligente cât și managementul integrat destinat sincronizării consumului cu producția de energie generează cantități uriașe de date ce necesită algoritmi performanți care să realizeze diverse operații pe setul de date. Astfel, a apărut conceptul de “**big data**”. Provocările în domeniu includ analiza, managementul, captarea, căutarea, partajarea, stocarea, transferul, vizualizarea, interogarea și caracterul datelor (public sau privat).
- e. O altă direcție ce poate fi dezvoltată este dimensionarea și configurarea resurselor de energie regenerabile (solar, eolian) astfel încât predictibilitatea acestora și implicit, calitatea energiei generate să crească. Soluțiile pot veni din includerea unor mici elemente hardware pentru achiziție și control, însoțite desigur, de aplicațiile software necesare.

Fiind vorba de o infrastructură de dimensiuni foarte mari, soluțiile inteligente nu vor fi realizate dintr-o dată. Aceste soluții fac parte dintr-un ansamblu de strategii ce vor pune în

mișcare un proces complex și perfectibil ce va evolua gradual, în ariile prioritare, în următoarele decenii. Odată ce rețeaua inteligentă va deveni matură, va putea aduce transformări comparabile cu felul în care Internetul le-a adus asupra felului în care trăim, lucrăm și învățăm.

Contextul tehnologic. Industria ITC și-a pus amprenta în aproape toate domeniile, iar din această categorie face parte și domeniul ingineriei energetice. Avansul tehnologic din majoritatea domeniilor are la bază relația triarhică dintre următorii vectori de evoluție: reducerea costurilor, reducerea dimensiunilor și creșterea performanțelor. Fiind un domeniu vast și imposibil de cuprins în câteva rânduri, noile tehnologii în domeniul ingineriei energetice aplicabile în cadrul orașelor inteligente, în continuare sunt prezentate succint câteva dintre cele mai importante realizări.

Rețelele inteligente constituie un sistem energetic în care rețeaua electrică este dublată de o rețea informatică ce realizează conectarea operatorilor de pe piața de energie electrică și prin informație. Dezvoltarea sistemelor Smart Grids permit utilizarea noilor tehnologii pentru creșterea eficienței, securității și fiabilității tuturor componentelor sistemelor electroenergetice.

În ceea ce privește protocoalele de comunicare, există o serie de standarde care specifică formatul pachetelor de date, modul de transmitere a informațiilor (semnale electrice, accesul la rețea, compresia și codarea). Printre cele mai populare standarde se numără BACnet (ISO 16484-5), KNX (ISO/IEC 14543), AMX, ZigBee (IEEE 802), LonTalk, RS232, comunicarea fără fir (IEEE 802.11).

Contextul socio-economic. Conceptul de „oraș inteligent” reprezintă unul dintre rezultatele eforturilor Uniunii Europene în cadrul „Europe’s Digital Agenda” și cuprinde numeroase aspecte interdisciplinare: guvernare, planificare, sănătate, transport, ecologie și energie.

Conform Enerdata prețul energiei în sectorul rezidențial a crescut cu peste 64% din 2004 până în prezent. În cazul în care s-ar implementa soluții de eficientizare energetică a tuturor clădirilor din sectorul rezidențial și terțiar (eurima, 2015) acest proces ar genera următoarele beneficii economice și de mediu: 1) economii care ar totaliza 270 miliarde € în fiecare an; 2) reducerea cu 460 de mil de tone a dioxidului de carbon emis în atmosferă și 3) crearea a peste 530.000 de locuri de muncă.

Conform Iai-tech, 2009, managementul inteligent poate reduce costurile de întreținere cu până la 50% pentru clădirile vechi și cu până la 20% pentru clădirile noi, traducându-se în facturi mai mici deci, economii importante la nivelul întregii Uniuni Europene.

O planificare inteligentă a producției și consumului de energie coroborată cu soluții inteligente de reducere a consumului de energie la utilizatorii casnici și terțieri ar aduce nu numai beneficii economice și sociale, ci ar permite orașelor să evolueze în orașe inteligente unde luarea deciziilor de dezvoltare ulterioare vor fi armonioase și fundamentate pe analize solide.

Elementele de dificultate ale problemei. Sistemele de monitorizare prezente oferă informații de înaltă calitate și permit gruparea în rețele avansate de senzori fără fir și rețele de senzori autonome ce permit totodată autocalibrarea și procesarea datelor. De fapt, fiecare flux de energie poate fi monitorizat și măsurat, precum și funcționarea și performanțele sistemelor și componentelor. Cu toate acestea, în ceea ce privește problema diferenței dintre prognoza consumului și performanța energetică reală încă este nevoie de o mai bună înțelegere a: I) ce măsurăm; II) cum se măsoară; III) cum se procesează datele obținute; IV) modul de utilizare a datelor astfel încât informația procesată să fie relevantă.

Mai mult decât atât, atunci când se dorește stocarea unor cantități mari de date, este importantă proiectarea bazei de date multidimensionale de natură să proceseze rapid datele pentru a furniza un răspuns prompt. Pentru a putea realiza un demonstrator care să simuleze un oraș inteligent, într-o primă etapă, datele procesate și achiziționate nu vor fi stocate într-o bază de date relațională, ci într-un cub MOLAP (multi-dimensional online analytical processing).

Limitările abordărilor curente. Problema identificării modelului optim care să aproximeze cât mai fidel consumul și confortul utilizatorilor este intens dezbătută de comunitatea științifică specializată în acest domeniu. Spre exemplu: algoritmi ce se bazează pe rețele neuronale, metode probabilistice de raționament incert, metode de învățare statistică, cunoscuți și sub denumirea de algoritmi de tip “black box”, permit rezolvarea problemelor neliniare prin identificarea funcției de regresie a consumului (aproximarea consumului de energie cu o precizie ridicată), însă acești algoritmi nu pot detecta acțiunile utilizatorilor (deschiderea geamului, deschiderea ușii) care duc la o scădere/creștere (după caz) a temperaturii interioare și implicit la o creștere a necesarului de energie. În aceste situații eroarea prognozei consumului poate depăși chiar și 200%.

Automatizarea cladirilor. Procesul de digitizare a instalatiilor termice a inceput la inceputul anilor 80 dupa aparitia controlerelor bazate pe microprocesoare. Odata cu cresterea necesitatii de a interconecta echipamentele termice in retea, majoritatea producatorilor au inceput sa adauge functii de comunicare si odata cu aceste necesitati au aparut si cateva protocoalele ce au dominat industria cladirilor automatizate.

Seccesul aplicatiilor orientate pe obiect in domeniul calculatoarelor si comunicatiilor a permis aparitia protocoalelor de comunicare orientate pe obiect. Ideea de baza era gruparea mai multor obiecte intr-o forma ierarhica intr-un alt obiect.

In prezent, odata cu dezvoltarea tehnologiilor ce utilizeaza internetul, serviciile web joaca un rol din ce in ce mai important. Serviciile web permit o integrare facila a aplicatiilor industriale in cladirile automatizate, fiind independente de platforma de programare utilizat. Baza serviciilor web este construita pe standarde ce includ XML, WSDL, SOAP, UDDI. Arhitectura serviciilor web este conceputa in jurul a trei entitati: furnizorul, solicitantul si agentul de servicii web.

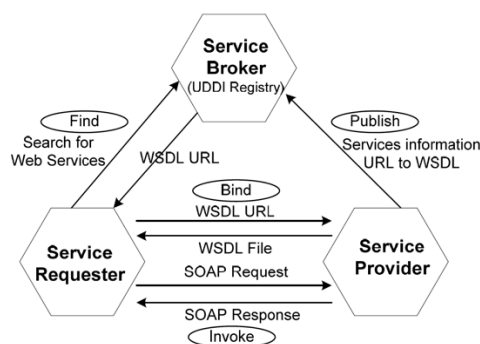


Fig. 1. Fluxul de date al serviciilor web

Furnizorul de servicii web este proprietarul serviciului. Din aceasta perspectiva arhitecturii acesta detine o platforma care gazduieste si ofera acces la serviciile web.

Solicitantul de servicii web este entitatea care necesita ca anumite functii sa fie executate. Practic solicitantul initiaza interactiunea cu furnizorul de servicii web.

Agentul de servicii web este un registru care ofera servicii de cautare a serviciilor web disponibile. Majoritatea registrelor au si un clasificator care permite utilizatorilor sa caute mai usor un anumit serviciu web.

Dispozitive IOT. „Internet of Things-IoT” reprezintă interconectarea, prin intermediul internetului, a unui număr mare de "dispozitive". În prezent există 5 miliarde de obiecte inteligente conectate la Internet, și este de așteptat că aceste vor depăși 25 de miliarde până în 2020. Aplicatiile IoT se adreseaza cladirilor inteligente, monitorizare medicală, oraș și utilități inteligente, agricultură inteligentă, control industrial, transport inteligent.

Pe măsură ce domeniul IOT se extinde, atacurile împotriva sistemelor IOT sunt în creștere în număr și complexitate. Atacurile împotriva sistemelor IOT urmăresc furtul de date sensibile, introducerea de informații false sau funcționalitatea anormală a rețelelor și a serviciilor. Prin

urmare, este important să se asigure o securitate sporita sistemelor critice prin asigurarea protecției IoT împotriva atacurilor. Principala provocare în materie de securitate este de a preveni atacatorii IoT de la obținerea controlului asupra sistemului IoT.

III. STUDIU DE CAZ

Casa pasiva Politehnica este un duplex pe doua nivele pentru uz rezidential. Imobilul este situat in campusul Universitatii Politehnica din Bucuresti, pe o suprafata plana usor umbrita cu orientare N-S.



Fig. 2. Vedere de sud a Passivhaus POLITEHNICA (Vest si Est).

Casa pasiva Politehnica (figura 2) este o cladire de test situata la urmatoarele coordonate: latit. 44.4380N, longit. 26.0470E si 76,6 m altit .

Cladirea se compune din doua case, "Casa de Est" si "Casa de Vest", care sunt incluse intr-o anvelopa termica comuna. In tabelul 1 sunt prezentate suprafetele cladirii.

TABEL 1. Suprafetele Caselor pasive Politehnica

Aria	Casa Est	Casa Vest
Suprafata utila	139,95 m ²	139,95 m ²
Suprafata vitrata nord	2.80 m ²	2.80 m ²
Suprafata vitrata est	9.13 m ²	0
Suprafata vitrata vest	0	9.13 m ²
Suprafata vitrata sud	17,94 m ²	17,94 m ²
Usa	2.19 m ²	2.19 m ²
Peretele exterior	182.52 m ²	182.52 m ²
Acoperis	96 m ²	96 m ²
Amprenta la sol	94.40 m ²	94.40 m ²
Volum	435 m ³	435 m ³

Conceptul de casa pasiva a fost dezvoltat de Institutul Passivhaus din Germania si isi propune sa reduca la minimum necesarul de energie termica printr-o izolare etansa a cladirii, prin eliminarea punctelor termice si utilizarea unui recuperator de caldura aer-aer pentru incalzirea cladirii.

Pentru ca o casa sa obtina certificatul de casa pasiva, aceasta trebuie sa indeplineasca trei criterii de baza:

- necesarul specific de energie pentru incalzire $\leq 15 \text{ kWh/m}^2\text{an}$
- schimbul de aer intre interior si exterior la o presiune de 50 Pa $N_{50} \leq 0,6 \text{ h}^{-1}$
- necesarul de energie primara (pentru toate serviciile) $\leq 120 \text{ kWh/m}^2\text{an}$.

În plus, sunt facute urmatoarele recomandari, in functie de clima:

- Coeficientul global de schimb de caldura $U \leq 0,8 \text{ W/m}^2\text{K}$ pentru ferestre;

- Coeficientul global de schimb de caldura $U \leq 0,15 \text{ W/m}^2\text{K}$ pentru peretii exteriori;
- Eficienta recuperatorului de caldura $\geq 75\%$;
- Puncti termice $\leq 0,01 \text{ W/mK}$

În tabelul 2 sunt descrise caracteristicile termice pentru toate elementele de construcție: λ conductivitatea termică, coeficientul pierderilor de caldura U și grosimea fiecărui strat.

3.1 Sistemul de monitorizare

Soluția software a sistemului de monitorizare este compusă din trei proiecte software distincte: DataLayer, DataLogger și DataView. Aceasta structurare a soluției software permite o întreținere mai ușoară a codului atunci când trebuie realizate noi modificări.

Proiectul software DataLayer

În prima instanță a fost creată o bază de date utilizând programul Microsoft SQL Server Management Studio 2012. Fără de varianta clasică de conectare la baza de date, s-a optat pentru transpunerea modelului fizic al bazei de date într-un model conceptual numit DatabaseModel. Practic, modelul conceptual se bazează pe crearea entităților separate și descrierea relațiilor dintre acestea (fig.3).

Cu alte cuvinte, fiecare tabel existent în baza de date este mapat într-o clasă C# care, odată instanțiată, devine purtătoare de informații. Pentru crearea modelului conceptual a fost folosit o tehnologie Entity Framework.

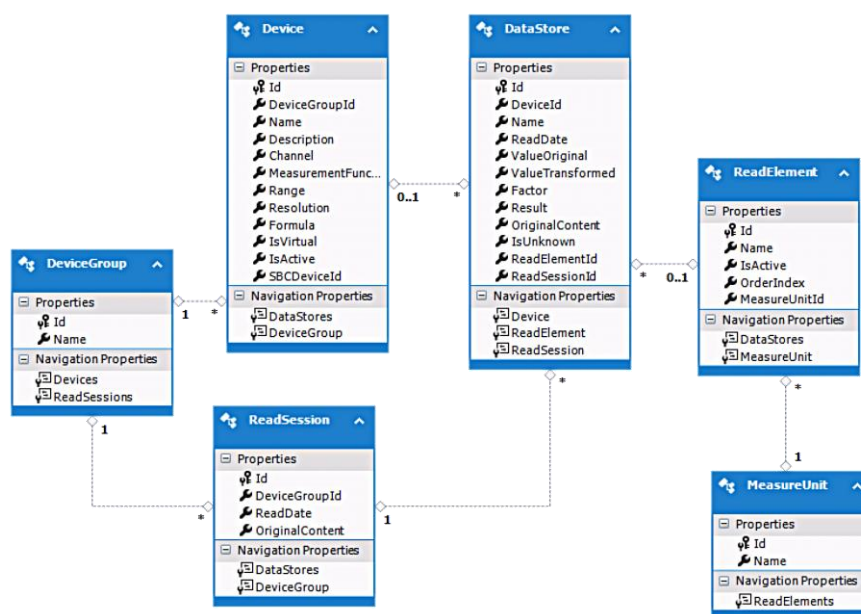


Fig. 3. Diagrama conceptuală a bazei de date pentru sistemul de monitorizare a casei pasive Est

Entity Framework (EF) permite dezvoltatorilor să lucreze cu seturi de date structurate, fără să cunoască întreaga arhitectură a bazei de date. Cu Entity Framework, dezvoltatorii pot lucra la un nivel mai ridicat de abstractizare, și pot crea și menține aplicații orientate spre date cu mai puțin cod scris decât în aplicațiile tradiționale.

Modelul de stocare a datelor este compus din 6 tabele SQL. Principalele atribute ale fiecărui tabel sunt prezentate în tabelul 2.

TABEL 2. Modelul de stocare a bazei de date.

	Device	ReadSession	MeasurmentUnit	DeviceGroup	DataStore	ReadElement
Baza de date	Retine numele senzorului; numarul canalului daca este cazul; descriere senzor, intervalul de masurare, rezoluția si groupingID	Fiecare citire de date este identificata printr-un id de sesiune unic (sessionId). Aceasta modalitate permite identificarea tuturor masuratorilor realizate indiferent de platforma hardware sau sistemul de monitorizare	Retine unitatea de masura pentru fiecare dispozitiv atasat	Retine numele platformei hardware in cazul in care sunt mai multe platforme hardware disponibile	Retine de datele colectate într-o singura sesiune.	Pentru fiecare element de citire (senzor) este stocat un status curent, în cazul în care valoarea este în afara domeniului sau a unui canal transmite valori nule

Proiectul software DataLogger

Proiectul DataLogger este cel mai important deoarece concentreaza atat fluxurile de date transmise de senzori, cat si fluxurile de date rezultate in urma operatiilor CRUD (Create, Read, Update si Delete).

Fluxurile de date sunt gestionate prin intermediul a doua fire de executie si un sistem de gestiune a evenimentelor ce pot aparea la un anumit interval de timp. Firele de executie au urmatorul rol:

1. gestioneaza schimbul de date cu baza de date
2. gestioneaza colectarea informatiilor primite de la senzori

Sistemul de monitorizare inregistreaza informatiile primite de la senzori utilizand un pas de discretizare de 10 minute. Platforma software ofera posibilitatea de a inregistra date utilizand un pas de discretizare de ordinul secundelor.

Una din problemele intalnite in timpul dezvoltarii acestui modul a fost aceea ca platforma .Net 4.5 nu ofera clase suport optimizate pentru aplicatii care necesita utilizarea firelor de executie. In urma unor teste, s-a constatat ca intr-un interval de 24 de ore apare un defazaj de 3-4 secunde intre parametrii masurati.

Pentru a elimina acest defazaj, a fost utilizata biblioteca open source Quartz. Bucla de timp este realizata din 3 componente: **StdSchedulerFactory**, **JobBuilder** si **TriggerBuilder**.

StdSchedulerFactory ofera mediul in care pot fi create un numar de Job-uri ce sunt lansate de Triggeri.

JobBuilder este componenta ce se afla întotdeauna într-o stare valida si este utilizata pentru a invoca pornirea unui serviciu sau realizarea unei actiuni; în cazul nostru Job-ul creat va realiza citirea datelor de la senzorii instalati in cladire.

TriggerBuilder lanseaza executia job-urilor ce sunt programate sa înceapa, in mod repetat, la un anumit interval de timp.

Prin urmare, Job-ul lansat parcurge urmatoorii pasi:

1. Initializeaza DataLogger-ul. Programul goleste buffer-ul si seteaza un timer pentru citire.
2. Citirea datelor. Citirea datelor este realizata de protocolul de comunicare RS232 si USB. Mesajele transmise de data logger sunt apoi prelucrate utilizand librariile puse la dispozitie de furnizor.
3. Inserarea datelor in baza de date. La sfarsitul fiecărei bucle, timer-ul declanseaza un eveniment. Acest eveniment consta in crearea unui ID pentru sesiunea de citire ce va fi stocat in tabela ReadSession. Id-ul de sesiune permite identificarea

întregului esanțion de măsurători. După parsarea datelor colectate, acestea sunt introduse în baza de date în tabelul DataStore.

Platforma hardware a sistemului de monitorizare

Cu ajutorul aplicației software sunt monitorizați 5 senzori ce comunică wireless cu nodul central. Toate datele sunt transmise server-ului prin intermediul unui controller cu conexiune tip USB. Toate componentele sunt dotate cu antene radio pentru a spori eficiența transmiterii de date. Caracteristicile tehnice ale senzorilor sunt prezente în tabelul 3.

TABEL 3. Caracteristicile tehnice ale nodului wireless Sparrow cu antena externă

Senzor UPB WI-FI Marimi:	Caracteristici:	
Umiditate	Interval masura: 0 ... 100 % Eroare de masura: $\pm 2\%$ RH	
Luminozitate	Interval masura: 0...1000	
Temperatura	Interval masura: $-40 \dots 100^{\circ}\text{C}$ Eroare de masura: $\pm 0.5^{\circ}\text{C}$	
PRR	Interval masura: 0...1000	

În figura 4 se poate observa distribuția senzorilor wireless în Casa Pasivă Est. Nodul central are un triplu rol:

- colectarea parametrilor măsurati de la nodurile wireless;
- transmiterea datelor prin interfața USB către aplicația software ce gestionează fluxurile de date;
- colectarea parametrilor măsurati de senzorii atașați nodului central;

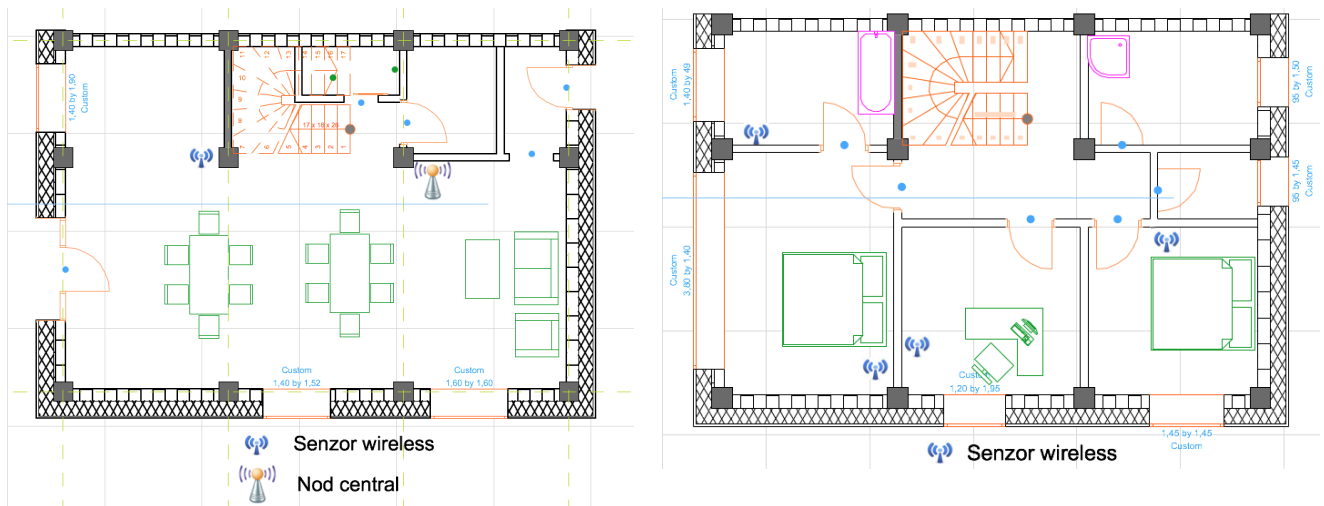


Fig 4. Dispunerea senzorilor în Casa Pasivă Est, parter și etaj.

Nodul wireless Sparrow este compus dintr-un sistem multi-controller. Această metodă permite distribuția eficientă a sarcinilor în cadrul dispozitivului. Astfel, nodul conține un controller cu stivă USB, specializat în transferul datelor prin intermediul interfeței USB, iar controller-ul radio este responsabil pentru transferul datelor fără fir.

Având în vedere că platformele Sparrow pot avea antene externe, sistemul permite comunicația pe raza mai mare între nodurile echipate cu astfel de antene. Raza de comunicație variază cu dimensiunile antenei externe.

Modulul sparrow foloseste tehnologie de comunicatie radio in frecventa 2.4GHz veche de doua generatii, forțând o comunicare simplificată pentru compatibilitate. Modulul Sparrow permite functionalități avansate, precum transmisie la 2Mbps (fata de 256kbps).

Capabilitatile senzoriale sunt extinse de senzorii de temperatura, umiditate, luminozitate, masurare a indicelui UV si a radiatiei incidente in spectrul IR, vizibil si UV, precum si un barometru/altimetru de precizie.

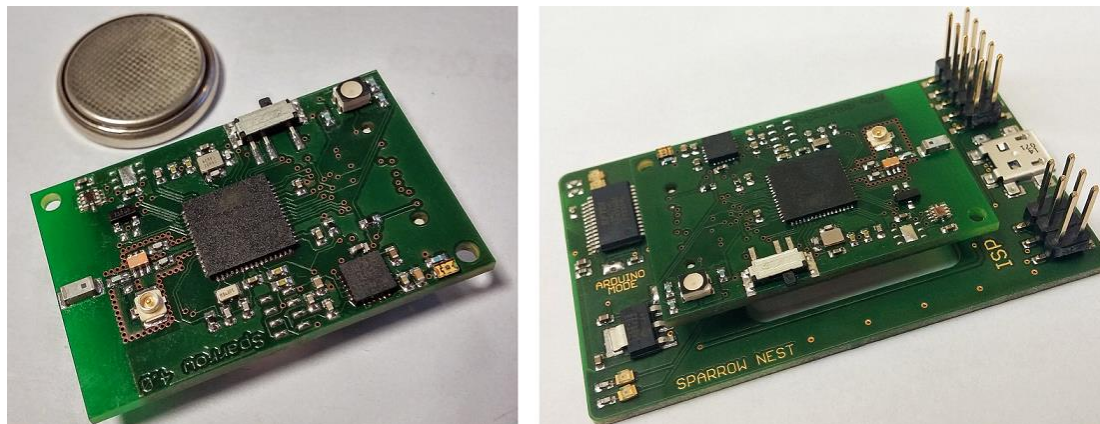


Fig 6. Nodul senzorial Sparrow.

Dupa cum se poate observa in figura 21, in centrul nodului senzorial se afla microcontroller-ul Atmega128RFA1.

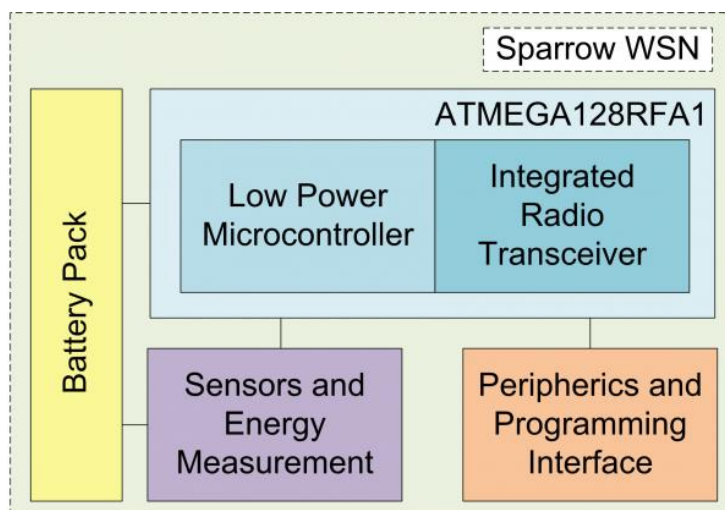


Fig 7. Arhitectura nodului senzorial Sparrow

ATmega128RFA1 este un microcontroller cu o arhitectura pe 8 biți. Ca urmare, registrele și magistrala interna de date sunt pe 8 biți.

ATmega128RFA1 are urmatoarele caracteristici:

- 128KB memorie Flash, sau dimensiunea maxima a programului care poate fi scris in microcontroller
- 4KB memorie EEPROM
- 16KB memorie RAM
- transceiver radio IEEE 802.15.4 în banda de 2.4GHz încorporat
- frecvența maxima de lucru de 16Mhz
- tensiune de alimentare între 1.8 si 3.6V
- timere cu canale PWM multiple
- canale de ADC multiple, precizie de 10 biți (situate pe portul F)
- cinci porturi de I/O digitale (de la portul B la G)
- interfețe seriale: USART, SPI, TWI (two-wire serial interface)

- interfața programare JTAG.

Pentru a folosi transceiver-ul radio cu care sunt dotate nodurile senzoriale este utilizata biblioteca numita ZigduinoRadio scrisa pentru dispozitivele Arduino si este scrisa in limbajul C. Aceasta modalitate de transfer a datelor este utilizata atunci cand nodul senzorial Sparrow transmite nodului central date privind parametrii monitorizati.

Pentru programarea nodurilor Senzoriale si realizarea unui debugging este utilizat protocolul de comunicare pe seriala.

Nodul senzorial Sparrow utilizeaza senzorul de temperatura si umiditate Si7020-A20.

Senzorul **Si7020** este compus dintr-un circuit integrat monolitic ce utilizeaza tehnologia CMOS. Senzorului de umiditate si temperatura integreaza urmatoarele componente: un convertor analog-digital, procesor de semnal, datele de calibrare, precum si o interfața I²C (comunicare pe doua fire).

Senzorii de umiditate si de temperatura sunt calibrati din fabrica, iar datele de calibrare sunt stocate în memoria nevolatila. Acest lucru incheama ca senzorul este interschimbabil, fara recalibrare sau modificare software.

In tabelul 4 sunt prezentate specificatiile tehnice ale senzorului de temperatura si umiditate Si7020.

TABEL 4. Specificatiile tehnice pentru senzorul de temperatura si umiditate Si7020.

Caracteristica	Specificatii
Precizie pentru umiditate relativa	± 4% RH (max), 0-80%
Senzor de temperatura de mare precizie	±0.4 °C (max), -10 pana la 85 °C
Gama de operare umiditate relativa	0 to 100% RH
Gama de operare temperatura	-40 to +125 °C
Tensiunea de lucru	1.9 - 3.6 V

Pentru monitorizarea prezentei utilizatorilor in cladire si a fluxului luminos, nodul senzorial Sparrow utilizeaza senzorul de mica putere Si1145/46/47. Acest senzor touchless, include un convertor analog-digital, de mare sensibilitate, fotodiode, procesor de semnal digital, si unul, doua, sau trei LED-uri în infrarosu.

In tabelul 5 sunt prezentate specificatiile tehnice ale senzorului de prezenta Si1145/46/47.

TABEL 5. Specificatiile tehnice pentru senzorul de prezenta si lumina ambientala Si1145/46/47.

Caracteristica	Specificatii
Setarea nivelului de detectare a prezentei	de la 1 cm la > 50 cm
Senzor integrat de masurare a fluxului luminos	Rezolutie 100 mlx
Tip comunicare	seriala
Temperatura de operare	-40 si +85 °C

Pentru monitorizarea consumului de energie electrica este utilizat contoare de uz industrial și comercial E550.

Totodata, pentru ca sistemul de monitorizare sa permita SBC-ului (nodului central) posibilitatea de a colecta si stoca fluxul de date generat de nodurile Sparrow, intr-un format cunoscut (csv,txt, sau xlsx), este necesara dezvoltarea unei aplicatii software care sa fie lansata in mod automat la pornirea sistemului de operare.

Dinamica citirii datelor, generate de sistemul de monitorizare, este importanta atunci cand sunt necesare realizarea de analize comparative, identificarea unor evenimente sau identificarea altor schimbari ce pot aparea pentru un interval scurt de timp.

Se recomanda utilizarea pasului de discretizare cuprins intre un minut si zece minute pentru dispozitivele Sparrow. Pentru monitorizarea consumului de energie electrica se recomanda un pas de discretizare cuprins intre o secunda si un minut.

Utilizarea unui pas de discretizare mic va genera suficiente rezultate care sa ofere un spectru larg asupra activitatilor realizate in spatiile monitorizate.

Componentele ce țin de transmisia radio de pe stick-ul USB sunt identice cu cele de pe platformele Sparrow, permitand cod similar si ameliorand diferente de timing (datorita faptului ca ambele parti ale comunicatiei dispun de un Real-time Clock bazat pe acelasi cristal de cuarț).

Controller-ul folosit pentru comunicatia USB este ATmega32U4, un controller cu capabilitati USB full-speed, cu periferic incorporat dedicat comunicatiei USB.

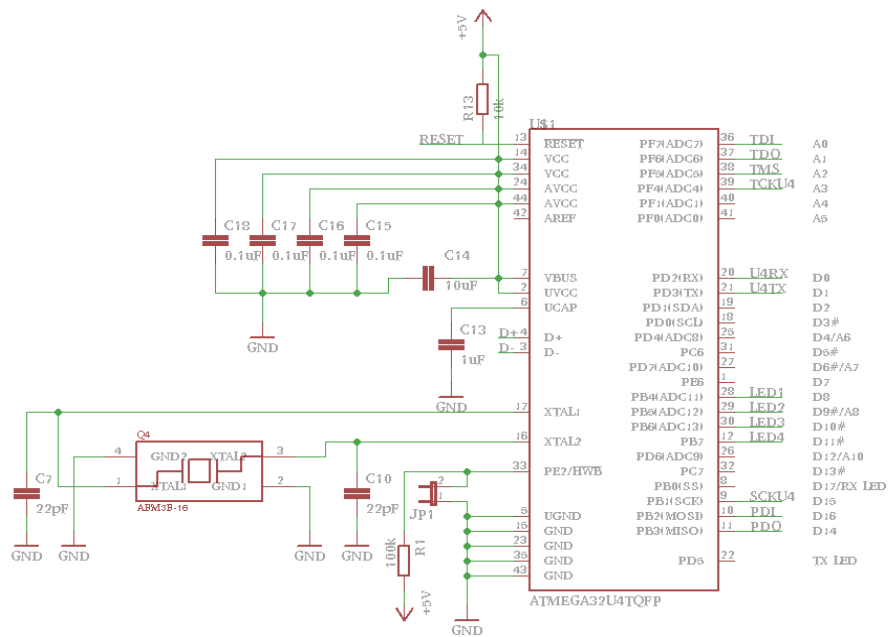


Fig. 5. Schema controller USB

Controller-ul Radio folosit pe platforma Sparrow este ATmega128RFA1. Schema controller-ului este prezentat in figura 6.

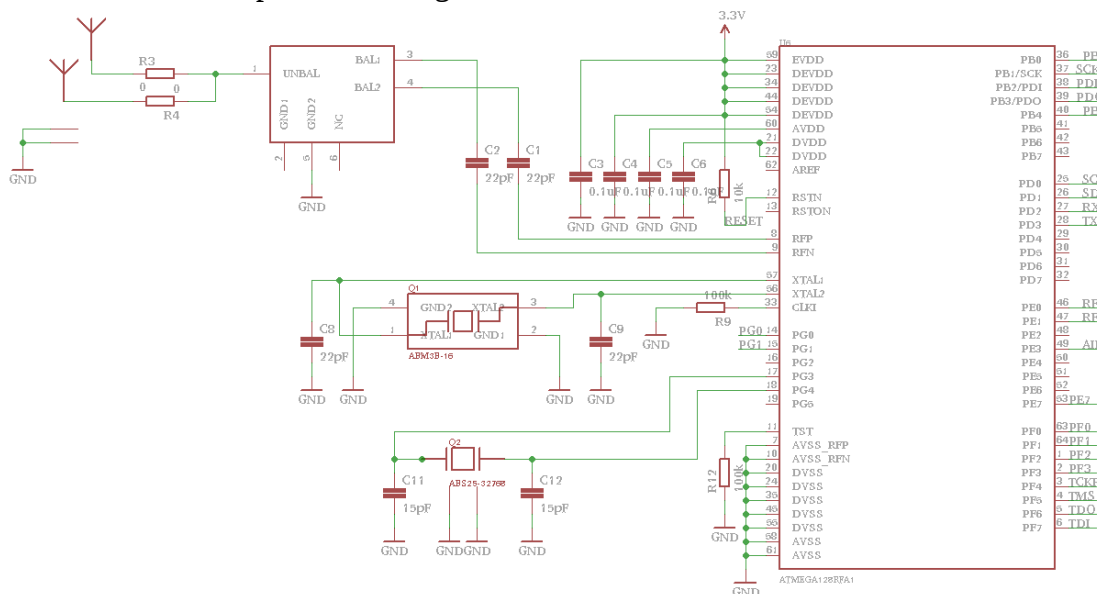


Fig. 6. Schema controller radio ATmega128RFA1

Platforma Sparrow dispune de două interfețe de programare/debugging pentru fiecare controller. Selecția se face cu ajutorul unor jumperi pe linii de clock ale conexiunilor către dispozitivele de programare. Astfel, pentru ISP este multiplexata linia SCK, iar pentru JTAG este multiplexata linia TCK. Ambele sunt legate cu rezistente pull-up pentru a nu fi influențate de semnale învecinate atunci când nu sunt selectate.

Soluția de monitorizare permite centralizarea parametrilor masurați pe un server (figura 7). Numărul de senzori existenți pentru monitorizarea caselor pasive asigură colectarea suficientă de informații privind activitatea utilizatorilor atât în cazul activității de birou, cât și în cazul activităților realizate în sectorul rezidențial.

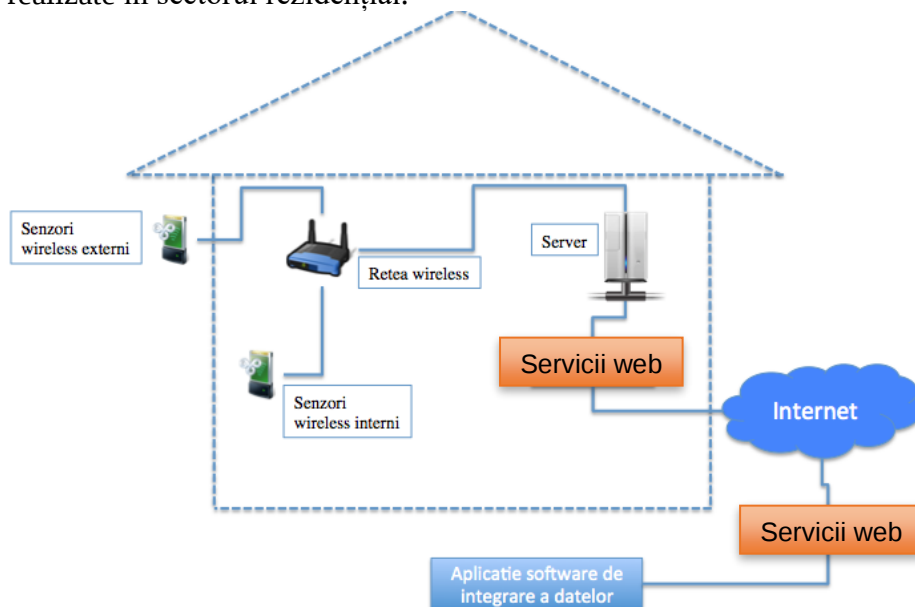


Fig. 7. Fluxul de date ale sistemului de monitorizare

Unitățile de tip server pastrează un istoric al datelor, în format CSV, pentru o săptămână. Monitorizarea parametrilor electrici este realizată prin utilizarea calculatorului single-board Raspberry Pi 2 Model B (figura 8) cu modul wi-pi ce permite centralizarea datelor într-un singur nod central.



Fig 8. Raspberry Pi 2 Model B

Pentru monitorizarea temperaturii, umidității, fluxului luminos și prezentei în imobil este utilizată **platforma hardware Sparrow**.

IV. APLICAȚIA SMART BUILDING CONTROLLER

Componenta SBC este o aplicație desktop destinată editării și gestiunii politicilor de control ale componentei Smart Building Controller asupra dispozitivelor și serviciilor instalate în

sistem (cladire). In sensul general al utilizarii sistemului (figura 9), utilizarea Editorului de politici intervine ulterior configurarii sistemului (obligatoriu) si eventual dupa etapa de definire a orariilor de functionare (etapa optionala) (figura 9).

Desi un orar de functionare este o forma simple de politica – politica temporala, ne vom referi la politici in sensul larg la notiunea ce se refera la multimi de reguli (complexe) de control al cladirii. Editarea politicilor de control al cladirii ramane o sarcina optionala, deoarece pot fi cazuri in care utilizatorul doreste doar actionarea unor dispozitive la anumite ore, bazandu-se doar pe orarii (schedules). Pot exista insa situatii in care orariile de functionare sa nu ofere suficienta logica program pentru a indeplini functia dorita de utilizator intr-o cladire, in anumite scenarii. Pentru implementarea unor algoritmi de control mai sofisticati pe baza de reguli, se foloseste editorul de politici si mecanismul intern al SBC-ului pentru executia acestora.

In contextul cladirilor inteligente, pentru usurinta utilizarii unui astfel de sistem de definire, gestiune si executie a politicilor, s-au facut unele simplificari ale notiunii de program de control al unei cladirii:

- In cadrul SBC-ului, o politica este un set de reguli ce formeaza total sau partial programul de control. Pentru aceasta ipoteza simplificatoare s-a plecat de la premiza ca utilizatorul nu detine cunostinte de programare imperativa intr-un limbaj de nivel mediu sau scazut pentru a implementa comportamentul dorit si prin urmare ii este mai simplu daca scrie un set de reguli pe care SBC-ul sa le execute in secventa.
- Intuitia unei politici poate fi gandita ca setul de reguli care se refera la un anumit aspect al functionarii cladirii, oferind modularitate (exemplu: Politica aere conditionate, politica lumini, politica instalatie de incalzire, etc.).
- Fiecare regula dintr-o politica are un set fix de campuri pe care utilizatorul trebuie sa le completeze intr-un editor pentru a forma corpul regulii. Astfel utilizatorul poate gandi programul de lucru in termeni de reguli.
- Fiecare regula poate contine un numar restrans de expresii, principale tipuri de expresii fiind: **in conditiile regulii**- expresii pentru citirea unor valori de la dispozitive sau pentru preluarea directa a rezultatului apelurilor unor servicii web, **iar in partea de actiuni** – expresii pentru actionarea dispozitivelor, grupurilor, locatiilor sau apelarea directa a unor servicii web de uz general, instalate in SBC. Editorul de politici suporta si alte tipuri de expresii auxiliare ce ajuta la formarea expresiilor principale (de exemplu: `[[DaysAgo(nDays)]]` pentru specificarea datelor calendaristice relative la data curenta, ca argumente ale unor expresii din conditii).

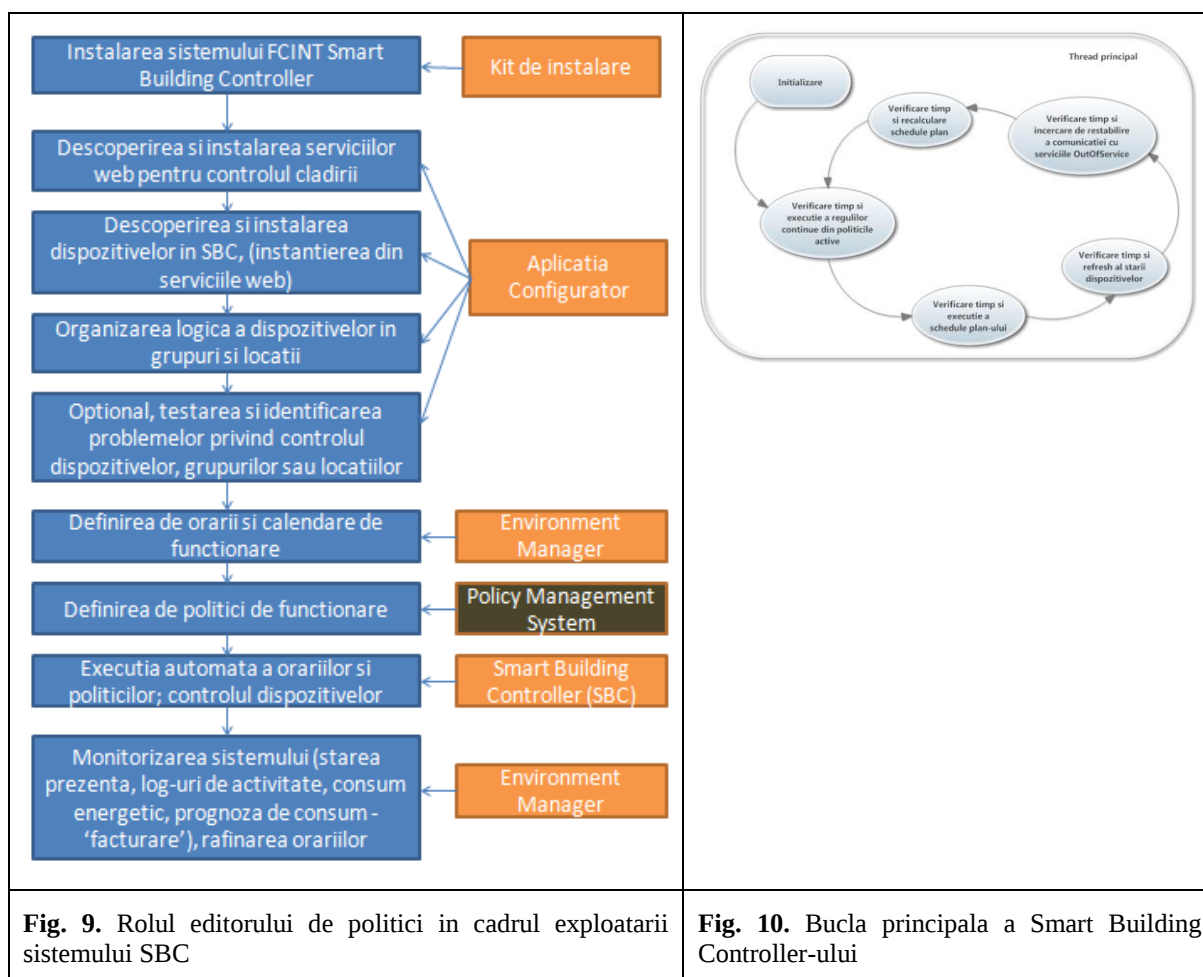
Abordarea bazata pe un sistem de reguli are un alt avantaj. Impartind politica de control a cladirii in reguli, acestea pot fi separate temporal la delay-uri predefinite, astfel incat procesorul si SBC-ul per ansamblu sa poata da prioritate, in bucla principala, si altor sarcini, prin executia lor intermitenta (exemplu la fiecare 200 ms), avand asadar un impact redus asupra performantelor intregului sistem de operare (ocupare CPU) si in acelasi timp indeplinindu-si functia. Se permite astfel configurarea de catre utilizator a frecventei cu care SBC-ul verifica anumite conditii, in functie de prioritate.

Executia regulilor se poate face izolat de alte reguli, sau prin comunicarea cu alte reguli ale altor politici, pe baza unei memorii de fapte. In acest sens, mecanismul de executie al politicilor permite citirea, stergerea, scrierea si interogarea existentei anumitor fapte in memoria de fapte, astfel incat, dupa caz, utilizatorul sa poata defini o logica mai complexa de executie, care sa

poata simula si bucele de program din programarea imperativa sau sa permita implementarea unor algoritmi de tip masina de stari finite (Finite State Machine), in functie de necesitati.

Arhitectura SBC-ului si a mecanismului de executie a politicilor a fost gandita si ca o arhitectura condusa de evenimente (Event-Driven Architecture). In acest sens, SBC-ul poate receptiona evenimente din mediul extern. La sosirea unui eveniment in sistem, utilizatorul poate specifica logica de control sub forma unor politici si reguli asociate *tipului de eveniment*. Astfel, in sistem utilizatorul poate defini doua tipuri de reguli

- Reguli continue, care sunt executate sub forma IF ... THEN ... ELSE periodic de SBC (figura 10)
- Reguli declansate de un anumit tip de eveniment (figura 11), care pot referentia in clauza IF ... THEN ... ELSE instanta evenimentului respectiv. Astfel se permite compunerea serviciilor care se executa in arhitectura SOA a SBC-ului si care trimit evenimente SBC-ului, pe baza politicilor si regulilor declansate de evenimentele respective. Si in acest caz se poate utiliza memoria de fapte, oferind posibilitati nelimitate de definire a politicilor pentru un anumit scenariu.



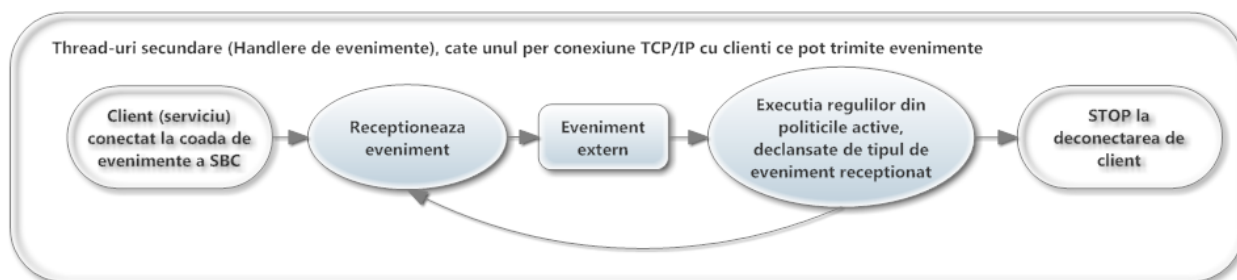


Fig 11. Thread-uri secundare pentru clientii conectati la coada de evenimente a SBC-ului (server TCP).

La receptionarea unui eveniment recunoscut de SBC, se executa toate regulile din politicile active asociate tipului de eveniment respectiv.

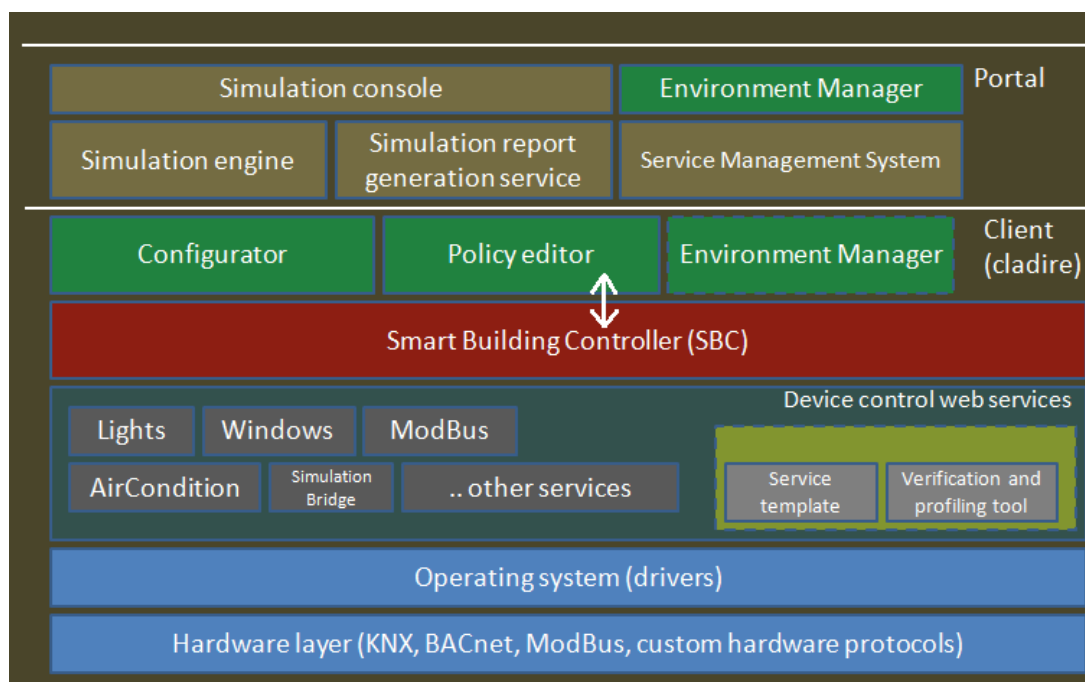


Fig 12. Arhitectura sistemului SBC si rolul editorului de politici

În cadrul arhitecturii sistemului (figura 12), editorul de politici se afla în comunicare directă și exclusivă cu un Smart Building Controller, rolul său fiind ca după autentificarea în SBC să gestioneze politicile definite pe acel SBC. Fiind o arhitectura orientată pe servicii, nu este obligatoriu, ca în cazul Configuratorului, ca editorul de politici să funcționeze pe același calculator cu SBC-ul. În acest sens, editorul de politici este clientul, iar SBC-ul este serverul.

V. CONCLUZII

Cercetările din domeniul utilizării inteligente a energiei în mediul urban s-au concentrat până în prezent în două direcții importante: **eficiența energetică a consumatorilor**, prin minimizarea consumului sau a costului facturii energetice și **rețele inteligente de distribuție**, care înglobează atât producerea, cât și distribuția de energie electrică și termică. În aceste paradigme, consumatorul și rețeaua sunt entități separate care nu colaborează, iar energia electrică și cea termică sunt tratate separat chiar dacă sunt produse prin cogenerare.

În concordanță cu studiul realizat ne propunem să realizăm, în premieră, un demonstrator ce va permite studierea răspunsului comun al unui grup mare de consumatori - echivalentul unui cartier, studiind totodată în premieră națională și sinergia între producător și consumator.

Se creează astfel premisele realizării în premieră europeană a unui campus inteligent, precursorul unor cartiere sau orașe inteligente. Proiectul viitor va reuni un grup de cercetători tineri cu preocupări științifice interdisciplinare, creându-se premisele realizării unui puternic centru de cercetare în domeniul orașelor inteligente care va colabora sinergic cu alte centre de cercetare europene în domeniu. Proiectul va reprezenta un nou punct de referință în ceea ce privește confortul și calitatea aerului clădirii însoțite de costuri reduse.

Efectele economice ale viitorului proiectului sunt legate de scăderea consumului de energie însoțită de reducerea cheltuielilor. Impactul asupra mediului se va reduce prin scăderea emisiilor de gaze cu efect de seră, datorită reducerii consumului de energie electrică și termică, realizând astfel pași importanți și eforturi susținute pentru reducerea consumurilor de energie și creșterea confortului în clădiri.