

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA
SECONDA FACOLTÀ DI INGEGNERIA
CON SEDE A CESENA
CORSO DI LAUREA IN INGEGNERIA INFORMATICA

INGEGNERIZZAZIONE DI SISTEMI DOMOTICI SU UNA INFRASTRUTTURA DI COORDINAZIONE AD AGENTI

Elaborato in Sistemi Distribuiti L - A

Relatore
Prof. Andrea Omicini

Presentata da
Eugenio Pavolucci

Correlatore
Ing. Alessandro Ricci

Sessione III

Anno Accademico 2002-2003

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA
SECONDA FACOLTÀ DI INGEGNERIA
CON SEDE A CESENA
CORSO DI LAUREA IN INGEGNERIA INFORMATICA

INGEGNERIZZAZIONE DI SISTEMI DOMOTICI SU UNA INFRASTRUTTURA DI COORDINAZIONE AD AGENTI

Elaborato in Sistemi Distribuiti L - A

Relatore
Prof. Andrea Omicini

Presentata da
Eugenio Pavolucci

Correlatore
Ing. Alessandro Ricci

Sessione III

Anno Accademico 2002-2003

*“Ringrazio tutti quelli che mi sono stati vicino e che
mi hanno sostenuto per il raggiungimento
di questa tesi”*

Indice

Introduzione	1
Scopo e contesto della tesi	2
Capitolo 1 - Domotica	4
Cosa è la domotica	4
Home VS Building Automation.....	5
Tecnologie e standard attuali	6
Sviluppi futuri	10
Capitolo 2 – Ingegneria domotica.....	11
Sistemi software complessi.....	11
Ingegneria del software Agent-Oriented	12
Metodologie per sistemi multi-agente	19
Infrastruttura di coordinazione TuCSon.....	21
Capitolo 3 – D.A.S. analisi	27
Concetti generali	27
Definizione delle risorse	32
Definizione dei Gruppi	37
Definizione dei Ruoli.....	39
Capitolo 4 – D.A.S. progetto	46
Modello del sistema domotico.....	46
Coordination media	49
Classi di Agenti.....	59
Capitolo 5 - Conclusioni.....	71

Introduzione

Lo sviluppo delle nuove tecnologie informatiche elettroniche e delle telecomunicazioni che insieme vengono denominate con la sigla ICT (Information and communication technology), ha portato innovazioni in moltissimi settori. Uno di questi, nel cui ambito si focalizza il presente lavoro, è quello della domotica.

Oggi si assiste ad un utilizzo crescente di prodotti elettronici digitali all'interno di edifici di uso privato e pubblico, che sta modificando radicalmente lo stile di vita delle persone.

Prima di tutto il personal computer, ma anche TV, Hi-Fi, DVD, proiettori, senza dimenticare gli elettrodomestici "intelligenti", cioè equipaggiati con pannelli di comando digitali, così come gli impianti di riscaldamento, condizionamento, antincendio, antintrusione.

Questi dispositivi, fino a poco tempo fa, erano indipendenti tra loro, ognuno con la propria modalità di funzionamento. In questi ultimi anni il passo in avanti che si sta compiendo attraverso la domotica, è quello dell'interconnessione tra tutti i dispositivi che popolano le abitazioni, progettando sistemi di comunicazione in grado di far loro scambiare dati, servizi e di attribuire all'utente utilizzatore potenti strumenti di comando e controllo.

Scopo e contesto della tesi

L'obiettivo principale della tesi sarà quello di progettare un sistema software capace di gestire gli impianti già presenti negli edifici pubblici e/o privati, quali di riscaldamento e di rilevamento dei possibili allarmi (fughe di gas, incendio, e altro), visti non più come entità distinte e autonome, ma combinate e gestite attraverso opportuni protocolli di comunicazione.

Il progetto chiamato D.A.S. (Domotic Agent System), per la sua conformazione, sarà rivolto principalmente a edifici pubblici: università, alberghi, e in generale a tutte le strutture utilizzate da individui che hanno esigenze e stili di vita differenti.

In questo modo i servizi di ogni stanza, camera, aula, corridoio, saranno percepiti da chi vi risiede come indipendenti da quelli degli altri ambienti. Cosicché i servizi di ogni camera potranno essere impostati dall'interno secondo le esigenze più disparate, e soprattutto in maniera distinta una dall'altra. Nello stesso tempo i vari ambienti faranno parte di un sistema distribuito più ampio dotato di particolari dispositivi che avranno il compito di monitorare lo stato di ognuno, e di intervenire, programmando dall'esterno i vari impianti, in modo da erogare servizi adeguati alle esigenze degli utenti, limitando al minimo i consumi complessivi.

Il sistema può essere scalabile fino alla gestione degli impianti di un'abitazione domestica, coinvolgendo sia l'home che la building automation.

L'analisi e il progetto del sistema verranno condotti utilizzando metodologie che si discostano da quelle ormai consolidate orientate agli oggetti, perché poco adatte alla trattazione di sistemi complessi eterogenei dinamici e distribuiti, ma rientrano in quelle denominate agent-oriented, attualmente in fase sperimentale ma che in futuro avranno uno sviluppo e un campo di applicazione vastissimo nella realizzazione dei sistemi distribuiti. Inoltre verrà sfruttata la tecnologia Java, e l'infrastruttura software TuCSon, che

fornisce modelli e servizi per la coordinazione e comunicazione, anch'essa in fase sperimentale ma già fornita di utili tools per lo sviluppo.

Infine saranno citati anche gli sviluppi che potrà avere in futuro il sistema, individuando eventuali scenari possibili di funzionamento, che oltre all'organizzazione degli impianti interni interesseranno la comunicazione dell'abitazione con l'ambiente esterno e viceversa. Un sistema cioè che permetta di segnalare particolari allarmi o malfunzionamenti ad eventuali centrali di assistenza esterne o semplicemente all'utente stesso non presente nell'edificio, oppure la possibilità di impartire comandi a distanza attraverso i più disparati dispositivi di comunicazione.

Queste caratteristiche si adattano perfettamente a quelle dei sistemi distribuiti che costituiranno il modello applicato a questo progetto.

La tesi è costituita da cinque capitoli:

Capitolo 1, verrà fornita una descrizione generale sulla domotica: le origini, la storia, le tecnologie odierne e come si evolverà in futuro.

Capitolo 2, si evidenzieranno i vantaggi dell'ingegnerizzazione dei sistemi domotici (visti come particolari sistemi complessi) utilizzando metodologie Agent – Oriented, e verrà descritta l'infrastruttura software TuCSoN alla base del progetto della tesi.

Capitolo 3, verrà eseguita l'analisi del sistema domotico D.A.S (Domotic Agent System).

Capitolo 4, riguarderà la progettazione del sistema domotico D.A.S.

Capitolo 5, conclusioni rispetto allo scopo principale della tesi.

Capitolo 1 - Domotica

Cosa è la domotica

La domotica è l'insieme delle scienze e delle tecniche connesse con l'elaborazione delle informazioni, tramite i più comuni strumenti tecnologici, all'interno dell'ambiente di vita o di lavoro.

Il termine Domotica deriva dall'importazione di un neologismo francese Domotique, contrazione della parola greca Domus (Casa) e Informatique (Informatica), e sta a significare per l'appunto informatica applicata alla casa, intesa quest'ultima nel senso più ampio del termine [10 -11].

Innumerevoli sono stati i passi che hanno posto le basi della nascita di questa disciplina, ad esempio, quando nel 1891 Wisconsin William Penn Powers, noto realizzatore del primo regolatore per impianti di riscaldamento, trasferitosi a Chicago, diede vita alla Power Regulator Company, antenata di quella che oggi è la Siemens Building Technology.

Attorno agli anni '50, quando negli Stati Uniti vennero realizzati i primi grattaceli nacque l'esigenza di un sistema centralizzato di gestione impianti, ed ecco che un gruppo di Ingegneri Energetici produsse il primo dispositivo pratico di controllo multiplo degli edifici chiamato System 320 [8]. Questo nuovo sistema prevedeva già l'uso di display a cristalli liquidi in lingua inglese dai quali ricevere tutte le informazioni in tempo reale sul funzionamento degli impianti.

Le origini di questa disciplina come la intendiamo noi oggi risalgono, intorno agli anni Settanta, periodo di nascita dei personal computers e dell'informatica, in quegli anni venne sviluppato lo standard X10 (prodotto dalla Pico Electronics), ancora oggi utilizzato in domotica, che sfrutta la linea elettrica per le sue trasmissioni [8].

Con il progresso delle nuove tecnologie ICT i dispositivi e gli impianti delle abitazioni possono oggi essere adoperati in un modo prima impensabile,

per ottenere maggior sicurezza e tranquillità, migliore qualità della vita, più efficienza e tempo libero, considerevole risparmio sui consumi energetici, e rendere indipendenti quelle persone che fino ad ora non potevano, nel vivere la loro realtà domestica.

Home VS Building Automation

Il campo di applicazione della domotica si divide in due categorie, in base al tipo di edificio a cui viene fatto riferimento, allora si parlerà di Building Automation, storicamente il primo ad essersi sviluppato, per l'automazione degli edifici: ambienti lavorativi, alberghi, università, e l'Home Automation, di recente applicazione, per tutto quello che riguarda l'automazione degli ambienti domestici.

Da parte di molti operatori nell'area della Building Automation si è formata l'opinione che gli stessi prodotti, gli stessi approcci commerciali e tecnici, se pur in scala ridotta, possano essere facilmente applicati adattati al mercato dell'Home Automation.

Questo passaggio in realtà non è così banale e l'opinione è sempre più smentita dai fatti. In effetti le due aree, per quanto vicine, sono molto diverse, come alcuni hanno già potuto sperimentare. Di seguito verrà proposto un confronto tra Building e Home Automation, prendendo in esame rispettivamente un ambiente lavorativo e un appartamento [9].

	Building	Home
Decisione sulla tipologia dei servizi applicabili e la loro pianificazione	Azienda	Chi ci vive
	Building	Home

Utenza	Lavoratore	Chi ci abita
Gestione dell'ambiente in base alle decisioni prese	Building manager	Chi ci vive
Livello di utilizzo del sistema di gestione	Uso complesso	Uso semplice
Dimensione dell'area abitativa da gestire	Edificio	Abitazione singola
Gestione degli spazi	Dinamica	Statica
Motivazioni all'installazione di questi nuovi sistemi di controllo	Sicurezza, Risparmio energetico, Controllo accessi	Qualità della vita, Sicurezza, Intrattenimento

Tecnologie e standard attuali

Il costante sviluppo delle tecnologie all'interno delle abitazioni domestiche e lavorative ha portato la creazione di numerosi standard di comunicazione e di controllo per elettrodomestici e dispositivi elettronici sempre più "intelligenti" in grado di risolvere compiti impensabili fino a poco tempo fa [10].

- 3G - con questa sigla si fa riferimento all'ultima generazione di telefonini, equipaggiati di fotocamera, telecamera, stereo, radio, permettono oltre alla comunicazione vocale, la trasmissione di informazioni in moltissimi media differenti, utilizzando la radio comunicazione a banda larga(2Mbit/s), di capacità estremamente

maggior rispetto agli standard della normale rete mobile GSM per telefonini.

- Bluetooth – standard sviluppato originariamente da Ericsson, ora sotto gli auspici della Bluetooth Special Interest Group (SIG) a cui fanno parte compagnie del calibro di 3Com, Ericsson, IBM, Intel, Lucent, Microsoft, Motorola, Nokia e Toshiba, per le comunicazioni radio a breve distanza (qualche decina di metri) fra diversi dispositivi elettronici. Segna la fine dei collegamenti fisici tra cellulari, palmari, Pc portatili e persino elettrodomestici di nuova generazione.
- BACnet – Un Protocollo di comunicazione dati per Building Automation e reti di controllo. Sviluppato dall’American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE). Fornisce uno standard per la rappresentazione delle funzioni e proprietà di ogni dispositivo, assegnando codici identificativi per il loro riconoscimento all’interno di una rete locale.
- CEBus - Nel 1984, i membri dell’EIA (Electronics Industry Association) capirono la necessità di includere negli standard caratteristiche più evolute, che andavano al di là dei semplici controlli di accensione e spegnimento dei dispositivi elettronici. Così nei sei anni successivi, gli ingegneri che rappresentavano le più famose compagnie internazionali di settore, rilasciarono lo standard CEBus, un insieme di architetture e di documenti che definiscono protocolli per la comunicazione attraverso le linee elettriche, doppini, cavi coassiali, infrarossi e fibre ottiche. Offrendo anche un linguaggio per la definizione di molti controlli object oriented per realizzare sistemi domotici. E’ stato una delle basi su cui sviluppare il più famoso standard KONNEX (KNX).
- DALI – Digital Adressable Lighting Interface è un protocollo dedicato per il controllo dell’impianto di illuminazione. Questo significa che non può essere utilizzato per il controllo di altri sistemi,

ma potrebbe costituire una parte di un sistema di building automation più esteso.

- EHS – European Home System, standard che rientra nei più alti livelli del modello di riferimento OSI è stato sviluppato con la collaborazione delle maggiori compagnie elettriche e di semiconduttori, garantisce la comunicazione fra apparecchiature presenti all'interno e attorno ad un'abitazione domestica o lavorativa, Gestendo milioni di indirizzi. Attualmente fa parte dello standard di riferimento KNX
- EIB – European Installation Bus propone un sistema per il controllo e l'automazione dei dispositivi di una abitazione. Tale protocollo può essere direttamente cablato negli apparecchi da controllare oppure in una centralina di comando. Attualmente incluso nello standard di riferimento KNX.
- Ethernet – Protocollo realizzato da Xerox nel 1970 riveste il secondo livello del modello OSI, è diventato uno dei più popolari sistemi di connessione tra computer per la realizzazione di reti locali (LAN).
- HAVi - Home Audio Video Interoperability, sviluppata da otto grandi case produttrici di apparecchiature elettroniche, garantisce attraverso il canale di trasmissione i.LINK l'interoperabilità tra dispositivi elettronici digitali di consumo, come TV, videocitofono, sistemi audio, per la realizzazione di scenari innovativi all'interno delle abitazioni.
- Jini - offre una semplice infrastruttura basata sulla tecnologia Java, per la gestione di servizi in una rete e per creare interazione tra programmi che usano questi servizi, indipendentemente dalla loro realizzazione hardware/software.
- LonWorks – Creata nel 1988 da Echelon Corporation, consiste in una piattaforma multimediale per il controllo di dispositivi, è stata

una delle prime tecnologie a modificare le architetture dei sistemi di controllo migliorando la tipologia Master-Slave, con il beneficio di realizzare impianti con un numero maggiore di dispositivi controllabili.

- OSGi – Open Service Gateway initiative è una piattaforma basata sulla tecnologia Java, realizzata con la collaborazione di 15 colossi dell'informatica nel 1999. Offre agli operatori di rete, service providers, creatori di apparecchiature, e venditori uno strato di funzioni e APIs per poter integrare facilmente i nuovi protocolli e i servizi dell'Home Automation e per realizzare soluzioni di monitoraggio a distanza, attraverso internet e cellulari.
- PDA – Dispositivo che riprende le funzionalità di un personal computer, word processor, organizer, accesso a internet wireless, interfacciamento rapido a reti locali, ma con dimensioni fisiche molto ridotte, all'incirca quelle di una mano. Realizzato per la prima volta nel 1996 da Palm Computing attualmente leader nel settore.
- TCP/IP - Protocollo di rete, principe per la comunicazione in internet, con la caratteristica di racchiudere le informazioni da trasmettere in pacchetti contenenti indirizzi, facilmente inoltrabili nella rete fino a destinazione.
- UPnP - Universal Plug and Play è un architettura per la realizzazione di reti peer-to-peer in ambiente domestico o in piccoli edifici pubblici. Abilita lo scambio di dati tra più apparecchi "intelligenti" e dispositivi wireless, con la supervisione di particolari PC dedicati. UPnP è indipendente da qualsiasi sistema operativo, linguaggio di programmazione, e supporta il rilevamento automatico dei dispositivi connessi alla rete.
- X10 – standard di comunicazione Americano per dispositivi dedicati che sfrutta la normale linea elettrica, prodotto nei primi anni settanta

dalla Pico Electronics, ancora oggi utilizzato in domotica, per la gestione degli impianti di illuminazione.

Sviluppi futuri

I moltissimi standard in circolazione per l'automazione di edifici, che si differenziano in base ai dispositivi utilizzati, alle dimensioni dell'area da controllare e dal tipo di servizi scelti, possono comportare un rallentamento della crescita del settore e un aumento dell'incertezza sia da parte del consumatore che delle stesse case produttrici.

Alla luce delle considerazioni fatte la domotica potrebbe evolversi in base alle seguenti ipotesi:

- Separazione dei campi di applicazione, progettando standard indipendenti per l'Home e Building Automation, come se non ci fossero legami tra abitazioni domestiche e pubbliche.
- Utilizzo di infrastrutture e metodologie adeguate per definire un sistema domotico, scalabile e configurabile in modo da essere adattato da tutti e due le tipologie di applicazione.

Questa tesi si avvicina maggiormente alla seconda delle due ipotesi perché simile alla filosofia dell'ingegneria di sistemi complessi, che si basa sui concetti fondamentali del risparmio delle risorse e riusabilità, coprendo la più vasta area applicativa possibile.

Capitolo 2 – Ingegneria domotica

Sistemi software complessi

I sistemi software capaci di gestire impianti, elettrodomestici e dispositivi di un'abitazione si possono catalogare come sistemi complessi, e più precisamente come sistemi di controllo. La loro realizzazione quindi non è immediata ma deve essere compiuta attraverso l'adattamento delle più avanzate metodologie introdotte dall'ingegneria del software.

Lo studio della complessità di questi sistemi ha portato la scoperta di importanti regolarità [1]:

- La complessità del sistema prende la forma di una struttura gerarchica, cioè una composizione di sottosistemi tra loro correlati che compongono il sistema principale. Le relazioni tra i componenti variano nel tempo e dipendono dalla natura dei componenti stessi, quindi non possono essere conosciute a priori ma può essere identificata la loro struttura organizzativa generale.
- La definizione dei componenti primitivi del sistema è relativamente arbitraria ed è definita dall'osservazione degli scopi e degli obiettivi.
- E' possibile distinguere le interazioni tra i sottosistemi e quelle interne ai sottosistemi.
- Le interazioni interne sono più frequenti rispetto a quelle tra i sottosistemi.

Un sistema complesso quindi è decomponibile e può essere definita anche la natura di alcune(non tutte) interazioni in fase di progetto.

Prese in considerazione queste caratteristiche l'ingegneria del software ha stabilito le basi per la trattazione di questi sistemi, riassunte in tre processi fondamentali:

- **Decomposizione** – suddivisione del sistema in parti più semplici da gestire.
- **Astrazione** – definizione di un modello semplificato del sistema.
- **Organizzazione** – definizione e organizzazione delle interrelazioni tra i vari componenti del sistema e raggruppare le funzioni e i costrutti base per ottenere librerie e modelli riutilizzabili.
- **Coordinazione** - definizione di modelli di coordinazione per le interazioni di un insieme di attività concomitanti.

Avendo caratterizzato i sistemi software complessi e identificato gli approcci fondamentali dell'ingegneria del software, che aiutano nella gestione di questa complessità, si vedrà il caso particolare dell'approccio Agent-oriented.

Ingegneria del software Agent-Oriented

Prima di spiegare in dettaglio i processi fondamentali che l'ingegneria del software ha introdotto per la gestione dei sistemi software complessi secondo l'approccio Agent-Oriented, verrà introdotto il concetto d'agente che sta alla base di questa nuova metodologia.

L'**Agente** è definito come un'entità di elaborazione situato in un determinato contesto e può agire in modo flessibile e autonomo per risolvere i suoi obiettivi [1].

Caratteristiche generali:

- Sono entità di problem-solving riconoscibili e definiti con precisi limiti e interfacce.
- Situati in un contesto con una precisa funzione di controllo, cioè ricevono input attraverso sensori relativi allo stato del contesto e agiscono di conseguenza nello stesso ambiente attraverso attuatori.
- Progettati per adempiere ad uno specifico ruolo, hanno particolari obiettivi da raggiungere.
- Autonomi, hanno il controllo sul loro stato interno e sul loro comportamento.
- Capaci di essere flessibili nell'inseguimento dei loro obiettivi, essendo pronti ai cambiamenti del contesto in maniera immediata e con iniziativa propria.

Per rappresentare la natura decentralizzata che spesso caratterizza questi sistemi complessi, con numerosi luoghi di controllo, il lavoro di un singolo agente non basta ma occorre una struttura **multi agente**.

Quindi gli agenti avranno bisogno di interagire tra loro, per ottenere i loro obiettivi individuali o per gestire le dipendenze(relazioni) che sorgono nell'essere situati in un contesto comune.

Queste interazioni variano da una semplice operazione semantica di tipo client-server ad una ricca interazione sociale.

Qualunque sia la natura dei processi sociali, ci sono due sostanziali tipi di interazioni tra agenti:

- Quelle che si verificano attraverso un linguaggio di comunicazione ad alto livello. Di conseguenza le interazioni vengono condotte in termini di quali obiettivi dovrebbero essere seguiti, in quale momento e da chi.

- Gli agenti operano in un ambiente nel quale hanno solo un controllo parziale, le interazioni quindi devono essere maneggiate in maniera simile e flessibile. In questo modo, gli agenti devono avere la capacità di fare decisioni dipendenti dal contesto, sulla natura e l'ambito delle loro interazioni e per rispondere a quelle non previste in fase di progetto.

Il contesto definisce la natura delle relazioni tra agenti e le leggi che devono essere rispettate durante l'interazione [1]. Per esempio alcuni agenti potrebbero formare un gruppo che collabora con gli stessi privilegi, oppure uno di essi potrebbe essere il supervisore di altri.

Per catturare questi collegamenti i sistemi ad agenti devono avere espliciti costrutti per modellare l'organizzazione delle relazioni, che in molti casi possono evolvere molto rapidamente nel tempo.

La localizzazione temporale di queste relazioni può variare enormemente. Per capire questa varietà e dinamismo i ricercatori hanno inventato protocolli capaci di organizzare la formazione e la distruzione di un gruppo d'agenti, specifici meccanismi e strutture per assicurare la correttezza delle azioni e dei macro comportamenti del collettivo.

Adottare l'approccio Agent-Oriented nell'ingegneria del software comporta un modo ottimale per decomporre il sistema complesso, utilizzando l'agente come chiave d'astrazione ideale per modellare le dipendenze e le interazioni tra i componenti. Con l'utilizzo dell'approccio ad agenti emergono inoltre caratteristiche importanti, ci si sposta da linguaggi le cui basi concettuali sono determinate dall'architettura della macchina a linguaggi le cui astrazioni chiave sono radicate nel dominio del problema. Vengono introdotti meccanismi migliori per riusare il codice, invece di fermarsi a riusi di componenti di sottosistemi e rigide interazioni preordinate tipiche dei sistemi Object-Oriented, si possono riutilizzare interi sottosistemi e interazioni flessibili. L'adozione di questa nuova metodologia non richiede una rivoluzione dell'organizzazione dei sistemi software già

esistenti. I software che non usano agenti possono essere incorporati in una maniera precisa. La tecnica usata è denominata wrapping (avvolgente), e consiste nella creazione di un'interfaccia attorno ai componenti software già esistenti, in modo che dall'esterno risulti un agente a tutti gli effetti, invece all'interno viene compiuta una funzione di traduzione bidirezionale: prendendo le richieste esterne e trasformandole in chiamate riconoscibili al software esistente e viceversa prendendo le richieste in uscita dal software esistente trasformandolo nel giusto comando di comunicazione per gli altri agenti.

Tenendo presente i vantaggi dell'approccio Agent-Oriented, i processi fondamentali dell'ingegneria del software per la gestione dei sistemi complessi si adattano nel modo seguente:

Decomposizione - I sistemi complessi si suddividono in una gerarchia di sottosistemi più piccoli, per ogni livello gerarchico, lavorano insieme per ottenere funzionalità da quelli di livello superiore. I sottosistemi sono formati da parti che a loro volta lavorano insieme per ottenere obiettivi specifici. Quindi è naturale distinguere i componenti in base agli obiettivi raggiunti (i quali si riconoscono nella metodologia orientata agli agenti in termini di funzioni/azioni/processi al contrario di quella orientata agli oggetti dati/oggetti). I sistemi complessi non hanno un centro di controllo, ma esso è dislocato su tutto il sistema, allora i singoli componenti dovranno localizzare e incapsulare il controllo delle proprie azioni [1].

Questi componenti per adempiere ai propri e ai collettivi obiettivi hanno bisogno di interagire tra loro, perciò in un sistema complesso sarebbe impensabile in fase di progetto sapere tutti i possibili collegamenti, perché queste interazioni si possono verificare in modo imprevedibile, per ragioni altrettanto imprevedibili e tra componenti imprevedibili. Quindi è molto meglio assegnare ai componenti l'abilità di fare decisioni sulla natura e sugli scopi delle loro interazioni a Runtime. Questa tecnica risolve il problema legato all'accoppiamento dei componenti. Essi sono progettati

per rispondere e generare spontaneamente delle richieste. In più siccome queste interazioni avvengono attraverso un linguaggio di comunicazione ad alto livello, tra gli agenti vengono rimosse le preoccupazioni di natura sintattica dei tipi di errori causati da un'inaspettata interazione.

Da questa discussione risulta che l'approccio orientato agli agenti è il più adatto, perché il modo migliore per decomporre un sistema complesso avviene attraverso componenti autonomi multipli che agiscono e interagiscono in modo flessibile per raggiungere i loro obiettivi.

Astrazione - Una parte importante della fase di progetto di sistemi complessi è trovare un buon modello per rappresentare il problema. Quando si progetta il software, le migliori astrazioni sono quelle che rendono minime le differenze tra le unità di analisi, che sono intuitive e usate per concettualizzare il problema e i costrutti presenti nel paradigma(modello) dell'effettiva soluzione del problema.

Nel caso dei sistemi complessi il modello consiste nel caratterizzare i sottosistemi, i componenti dei sottosistemi, le interazioni e le relazioni di organizzazione:

- I sottosistemi corrispondono naturalmente all'organizzazione tra agenti. Essi mettono in gioco molti componenti che agiscono e interagiscono in accordo alle loro regole all'interno di una struttura più grande.
- I componenti dei sottosistemi sono gli agenti.
- Le interazioni tra i sottosistemi e tra i loro componenti sono viste in termini di interazioni sociali di alto livello. In un sistema complesso ad ogni livello d'astrazione si possono trovare dei specifici oggetti che collaborano per ottenere viste(informazioni) di più alto livello. Un sistema ad agenti è descritto come formato da entità che collaborano per ottenere obiettivi comuni, che coordinano le loro azioni o negoziano per risolvere problemi.

- I sistemi complessi sono caratterizzati da numerosi cambiamenti delle relazioni tra i loro componenti. Essi richiedono anche gruppi di componenti per essere trattati come una singola unità concettuale, quando visti da un livello differente di astrazione. Una ricca collezione di strutture è disponibile per una rappresentazione esplicita delle relazioni organizzative. Esistono protocolli di interazione per formare nuovi gruppi e distruggere quelli non più voluti. E' chiaro che cambiando le relazioni organizzative cambia facilmente l'assetto di tutto il sistema e il significato delle interazioni, da qui si può notare la flessibilità della metodologia orientata agli agenti.

Organizzazione - I costrutti organizzativi sono la prima classe di entità nei sistemi ad agenti, le esplicite rappresentazioni sono costituite da relazioni organizzative e da strutture.

In più, i sistemi orientati agli agenti hanno meccanismi computazionali per una flessibile formazione, gestione e disgregazione delle organizzazioni. Questo potere rappresentativo dei sistemi ad agenti è ottimo per rappresentare due sfaccettature della natura dei sistemi complessi:

- Il concetto di componente primitivo può essere variato secondo le necessità dell'osservatore. Ad un livello, interi sottosistemi possono essere visti come entità singole o raccolte di agenti che a loro volta possono essere visti come componenti primitivi.
- Tali strutture offrono la possibilità di creare forme intermedie stabili del sistema finale, che sono essenziali per il suo rapido sviluppo. La loro disponibilità significa che gli agenti singoli o raggruppati in organizzazioni possono essere sviluppati singolarmente senza(o quasi) considerare tutto il sistema e poi aggiunti in maniera incrementale in un secondo momento. Questo, assicura una crescita graduale in funzionalità.

Coordinazione – In termini generali, la coordinazione è l'arte di organizzare le interazioni e le dipendenze fra le attività, a questo proposito è stato definito un coordination model il quale a sua volta definisce un framework formale nel quale vengono espresse le interazioni di un insieme di attività concomitanti [3]. Nell'informatica, un coordination model è tutto ciò che riguarda la creazione, la distruzione, la comunicazione, la distribuzione e la mobilità sullo spazio, così come la sincronizzazione temporale, di un gruppo di entità software attive, siano processi, oggetti, o agenti. In altri termini un coordination model può essere definito come una struttura che tratta la comunicazione e la sincronizzazione di un sistema di agenti software autonomi.

Da quest'ultima prospettiva, un coordination model può essere pensato come costituito da tre elementi:

- **Coordinati** - le entità che hanno interazioni reciproche in base al modello del sistema, come gli agenti in un sistema multi agente.
- **Media di coordinazione** - le astrazioni che permettono le interazioni tra agenti, come se fossero centri attorno ai quali i componenti di un sistema coordinato vengono organizzati. Un esempio sono tuple center(spazi di tuple) introdotti in TuCSoN.
- **Leggi di coordinazione** - definiscono i comportamenti dei media di coordinazione in risposta ad eventi di interazione. Le leggi possono essere definite in termini di un linguaggio di comunicazione e di coordinazione, un esempio è ReSpecT.

I coordination model possono essere divisi in due classi: data-driven e control-driven [3]. Nei **control-driven** i media di coordinazione fondamentalmente uniscono lo spazio di interazione, controllando in base alle leggi di coordinazione come il flusso di eventi connette i coordinati (agenti) e come si propaga nel sistema, senza preoccuparsi dei dati

scambiati. Nei **data-driven** i media di coordinazione uniscono lo spazio di interazione in termini di interazione tra le entità, attraverso lo scambio e la sincronizzazione di dati a monte di un'occorrenza dei dati stessi.

L'infrastruttura di coordinazione TuCSon utilizzata in questa tesi si basa sul modello data-driven, prevedendo nella definizione di media di coordinazione flessibili, nei quali le leggi di coordinazione di base possono essere cambiate e adattate ai bisogni specifici di coordinazione del sistema.

Metodologie per sistemi multi-agente

L'approccio object-oriented per analizzare e progettare un sistema multi agente risulta essere non del tutto adeguato in quanto non è equipaggiato di astrazioni e modelli per la gestione delle interazioni indipendenti tra i componenti dinamici del sistema. In questo senso sono state introdotte alcune nuove metodologie tra cui SODA, che fonda le sue basi sul concetto di società di agenti, come prima classe di componenti del sistema, e di ambiente(environment), cioè lo spazio dove gli agenti vivono e interagiscono [4]. In questo senso SODA costituisce una metodologia di riferimento orientata a definire l'aspetto sociale, di coordinazione degli agenti, definendo un vero e proprio coordination model.

In particolare SODA permette all'ambiente dove vivono e interagiscono gli agenti di essere esplicitamente modellato e mappato su infrastrutture ben definite.

La metodologia in questo senso è di aiuto per la determinazione della struttura sociale richiesta, delle leggi di coordinazione, di come dovrebbero essere progettate e migliorate le regole sociali. Per ogni società, si dovrebbe essere in grado di determinare quanti comportamenti sociali

potrebbero essere incapsulati negli agenti, e quanti invece caricati sopra le infrastrutture sociali.

In oltre un agente non può essere pensato separatamente dal suo ambiente. Esso può essere eterogeneo, dinamico, aperto, distribuito e imprevedibile come ad esempio Internet. Le proprietà dell'ambiente modificano il modo nel quale gli agenti vivono e come pianificano le loro azioni.

La metodologia SODA porta a definire esplicitamente un modello di coordinazione usato per gestire le interazioni e regole sociali del sistema. In particolare tali regole sociali sono progettate come leggi di coordinazione, incorporate in un media di coordinazione e le infrastrutture sociali sono costruite sopra i sistemi di coordinazione.

Adattando i costrutti e i modelli di questa metodologia, verranno descritti di seguito i punti principali per l'analisi e il progetto del sistema domotico D.A.S. introdotto in questa tesi.

Analisi:

- Determinare le risorse, nei termini del servizio che erogano, i loro modi di accesso, e i protocolli di interazione nei termini delle informazioni richieste per erogare il servizio e quali informazioni vengono rilasciate quando viene invocato il servizio.
- Determinare i gruppi, definiti in termini delle loro operazioni sociali (gestione della temperatura), i loro permessi per accedere alle risorse, e le corrispondenti regole di interazione tra i gruppi e le risorse per raggiungere le operazioni sociali del gruppo.
- Analizzare i ruoli, definiti in termini delle loro operazioni individuali (invio della temperatura corrente), i loro permessi per accedere alle risorse e il corrispondente protocollo di interazione (tra ruolo e risorse) in termini di informazioni richieste o cedute dai ruoli in modo da completare le operazioni individuali.

La progettazione riguarda:

- Definire il modello della società definita a sua volta attorno ai coordination media e formata dai gruppi descritti nell'analisi
- Progettare le risorse del sistema, specificando la posizione che occupano nel sistema, il modo in cui vengono erogati i servizi descritti in fase di analisi e stabilendo le loro dipendenze e quali agenti sono collegati ad essi.
- Definire classi di agenti associati ai ruoli, determinare il numero, la locazione degli agenti e il protocollo di interazione(secondo il modello di coordinazione di TuCSoN) con i servizi per ogni ruolo.

Infrastruttura di coordinazione TuCSoN

TuCSoN è l'infrastruttura software che fornisce modelli e servizi per la coordinazione di agenti [2], scelta per la realizzazione del progetto di questa tesi, nata per l'implementazione di applicazioni internet, ma estendibile ad una vasta tipologia di sistemi complessi(distribuiti), basata sulla metodologia definita dal coordination model. La caratteristica principale che distingue TuCSoN da tutti gli altri, riguarda la tipologia di comunicazione tra gli agenti. Viene abbandonata la tipologia di comunicazione diretta(peer-to-peer o client-server), ritenuta inadeguata per soddisfare comportamenti dinamici, asincroni e indipendenti, in quanto implica un forte accoppiamento tra le entità interagenti in termini di nomi(chi sono gli interlocutori), luoghi(dove gli interlocutori si trovano) e tempi(quando interagiscono). TuCSoN definisce uno spazio di interazione(o di coordinazione) per la comunicazione tra agenti, costituito da una molteplicità di astrazioni indipendenti, chiamate tuple center,

estraendosi dalle dimensioni della rete su cui fisicamente le informazioni vengono trasferite [2].

Gli spazi di tuple (tuple center) sono contenitori di informazioni (tuple) con specifiche leggi di coordinazione per gestire le richieste dei vari agenti, che possono interagire tra loro senza conoscere a priori chi e dove sono gli interlocutori. Questo rende possibile inglobare le proprietà dei sistemi complessi all'interno dello spazio di definizione delle interazioni, assegnando al centro di tuple, comportamenti di eterogeneità e dinamicità, permettendo la cooperazione tra gli agenti mobili nello spazio e nel tempo e uno sviluppo incrementale delle applicazioni.

Le caratteristiche principali di TuCSoN:

Spazio di coordinazione - Lo spazio di coordinazione di TuCSoN costituito da una molteplicità di tuple center, diffusi sui nodi di Internet ed usati dagli agenti per interagire con gli altri agenti così come con la struttura di esecuzione locale. Ogni tuple center è associato ad un nodo ed è denotato da un identificatore localmente unico.

Come mostrato dall'esempio in Figura 1 (tratto dall'articolo "Coordination for Internet Application Development, A. Omicini, F. Zambonelli), ogni nodo offre la sua propria versione di TuCSoN (coordinazione) quindi il proprio media space (ovvero, l'insieme di identificatori ammissibili per il centro di tuple).

Di conseguenza, ogni centro di tuple può essere identificato o per mezzo del suo nome completo Internet (assoluto) o dal suo nome locale (relativo). Più precisamente, i tuple center (tc) offerti dal nodo Internet (node) possono essere raggiunti specificando il nome assoluto tc@node ovunque in Internet, oppure specificando il nome relativo (tc) all'interno del contesto del nodo stesso.

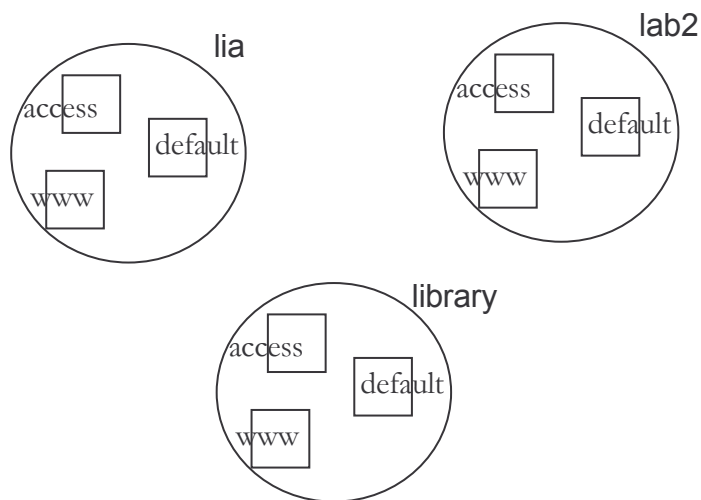


Fig. 1 – Tre nodi TuCSoN (lia, library, lab2), ognuno dei quali implementa la propria versione locale dello stesso media space TuCSoN (default, access, www).

Secondo l'esempio in Figura 1, il nome `access@lia.deis.unibo.it` denota univocamente il tuple center `access` residente nel nodo Internet `lia.deis.unibo.it`, mentre solo il nome `access` può denotare uno dei tre centri di tuple `access` residenti nei tre nodi, dipendenti dal nodo dove il nome è usato.

Corrispondentemente, lo spazio di coordinazione di TuCSoN può essere visto sia come uno spazio di interazione globale, rappresentando una raccolta di tuple center unicamente denotati (quando si fa riferimento ai nomi assoluti dei tuple center), sia come una raccolta di spazi di interazione locale, definendo lo stesso insieme di identificatori (quando si fa riferimento ai nomi relativi dei tuple center).

La sottosezione seguente dovrebbe fare chiarezza su come questo possa essere di beneficio per la coordinazione degli agenti, semplificando le interazioni.

Linguaggio di coordinazione - L'interazione tra gli agenti avviene scambiando tuple(informazioni) attraverso i tuple center, utilizzando un gruppo ristretto di primitive per la comunicazione [6]:

- out – scrive una tupla nel tuple center
- in – legge e poi elimina dal tuple center la tupla che corrisponde al modello richiesto, rimane in attesa per tutto il tempo in cui la tupla non è presente all'interno del tuple center.
- rd - legge dal tuple center la tupla che corrisponde al modello richiesto, anch'essa rimane in attesa fino a quando risulta disponibile l'informazione richiesta.
- inp – stesso comportamento della in ma non bloccante
- rdp – stesso comportamento della rd ma non bloccante

La forma generale delle operazioni ammissibili da TuCSoN, condotte da un agente su un tuple center risulta essere

tc?operation(tuple)

- tc - nome assoluto(tc@node) o relativo(tc) del tuple center a cui è riferito il comando. Se si usa il nome assoluto, tipicamente il tuple center si trova in un nodo remoto della rete rispetto all'agente, al contrario il nome relativo se l'agente si trova nello stesso contesto.
- operation – le operazioni ammissibili(out, in, rd, inp, rdp)
- tuple – template della tupla su cui viene eseguita l'operazione

I media di coordinazione - Un centro di tuple è una astrazione per la comunicazione che è percepita dalle entità che interagiscono come un contenitore di informazioni, ma i cui comportamenti, in risposta agli eventi di comunicazione possono essere definiti attraverso leggi di coordinazione. Ogni nodo di TuCSoN può contenere una molteplicità di tuple center i cui comportamenti possono essere singolarmente programmabili secondo gli specifici compiti di coordinazione. Questo può essere realizzato abilitando la definizione di reazioni agli eventi di comunicazione attraverso un linguaggio di specificazione chiamato ReSpecT [5]. Più precisamente, questo linguaggio rende possibile l'associazione tra le primitive di comunicazione di TuCSoN (out, in, rd, inp il rdp) alle specifiche reazioni. Una reazione è definita come un insieme di operazioni non bloccanti, può assumere due stati: una reazione riuscita può produrre atomicamente effetti sullo stato del tuple center, se fallisce non produce alcun risultato. Ogni reazione può liberamente aver accesso e modificare le informazioni raccolte nel tuple center, e può accedere a tutte le informazioni riferite all'evento di comunicazione provocato. Ogni evento di comunicazione può scatenare una molteplicità di reazioni: comunque, tutte le reazioni eseguite sono una conseguenza di un solo evento di comunicazione, vengono tutte eseguite in una sola transizione dello stato del centro di tuple, prima di servire qualsiasi altro evento di comunicazione. Di conseguenza, dal punto di vista degli agenti, il risultato della chiamata di una primitiva di comunicazione è la somma degli effetti di se stessa e di tutte le reazioni che ha provocato, percepite insieme come una transizione unica dello stato del tuple center.

Tuple center con ReSpecT - Un tuple center ReSpecT contiene tuple e specificazioni di tuple. Le tuple ReSpecT sono clausole logiche del primo ordine e usano la sintassi del Prolog, l'insieme delle tuple logiche di un tuple center ReSpecT, costituisce il suo spazio di tuple. Una

specificazione di tuple ReSpecT, invece è una clausola Prolog unitaria della forma `reaction(Op, R)` la quale associa un evento di comunicazione rappresentato come un termine logico `Op` alla reazione `R`.

Ogni reazione ReSpecT può avere accesso alle informazioni relative all'evento di comunicazione provocato, manipolare termini, e leggere, scrivere e consumare tuple. In particolare, predicati come `out_r`, `in_r`, `rd_r`, e `no_r` rendono possibile l'accesso e modificano lo spazio delle tuple [5].

Più precisamente:

- `out_r(T)`: inserisce la tupla `T` nell'insieme delle tuple.
- `in_r(TT)`: preleva una qualsiasi tupla che faccia matching con il template `TT`.
- `rd_r(TT)`: legge una qualsiasi tupla che faccia matching con il template `TT`.
- `no_r(TT)`: verifica la non presenza di tuple che facciano matching con `TT`.

In aggiunta vengono introdotti anche i due predicati `pre/0` e `post/0`, in modo da poter definire reazioni che avvengono solo nella fase in cui viene verificata la presenza della tupla richiesta nel tuple center (pre fase) e la fase di risposta successiva (post fase), quando la tupla richiesta viene inoltrata all'agente. Con la combinazione di questi nuovi predicati si possono realizzare anche altre primitive di comunicazione come la `in` e la `rd`.

In risposta ad un evento di comunicazione si possono associare anche molteplici reazioni processate in sequenza, il fallimento della singola, causa il fallimento dell'intera reazione composta, e la sua esecuzione non produce effetti sul tuple center.

Capitolo 3 – D.A.S. analisi

Concetti generali

La realizzazione di un sistema per l'automazione degli edifici, comporta prima di tutto l'individuazione del suo campo di applicazione, proprio per questo verrà descritta una vasta gamma di servizi, prescindendo dal tipo di abitazione su cui verranno applicati, in un certo senso quello che si intende evidenziare già dall'inizio dell'analisi è il concetto di "unificazione", correlazione tra sistemi di home e building automation.

Accessi – monitoraggio degli accessi, attraverso card magnetiche o impronte digitali, sia per poter entrare negli edifici, ma anche al suo interno, per monitorare l'accesso a determinate stanze, con la possibilità di ricostruire tutti gli spostamenti di un individuo dal momento della sua entrata fino a quello di uscita.

Illuminazione – controllo automatico ed anche a distanza dell'accensione, regolazione, e spegnimento delle luci, possibilità di realizzare un piano giornaliero, decidendo quale zona dell'edificio illuminare, quando iniziare e per quanto tempo lasciarla illuminata.

Intrattenimento – possibilità di realizzare scenari di intrattenimento, riproducendo una comunicazione telefonica o videotelefonica su qualsiasi dispositivo di riproduzione audio/video, oppure diffondere solo in determinate stanze l'immagine catturata dalle telecamere di sorveglianza, gestendo al meglio e in modo personalizzato tutti i media, TV, Hi-Fi, proiettore, vivavoce, videocitofono, presenti nell'edificio.

Sicurezza - garanzia contro il verificarsi di situazioni pericolose all'interno degli edifici, come potrebbero essere fughe di gas, principi di incendio, allagamenti, intrusioni indesiderate, tramite la gestione di unità di controllo che elaborano le informazioni provenienti da rivelatori(sensori), che

percepiscono i fenomeni, e tramite dispositivi che attuano le misure di difesa, segnalando l'accaduto anche a distanza, esempio alla polizia, ambulanze o vigili del fuoco.

Sistemi di energia – strumenti per il monitoraggio automatico dei carichi energetici, segnalando preventivamente eventuali sovraccarichi di potenza nella linea elettrica domestica.

Temperatura - gestione del confort termico, eventualmente possibile anche a distanza, consente la regolazione automatica per “zone” differenziate e scenari nei quali l'arco della giornata viene suddivisa in fasce orarie ognuna con la propria caratteristica.

Utenze elettriche - gestione di tutte le utenze elettriche lavatrici, asciugatrici, macchine per il caffè, forno elettrico, forno micro-onde, con particolare riferimento a quelle adibite alla riduzione degli sforzi fisici quali motorizzazioni per tapparelle, porte e finestre, con la possibilità di eseguire comandi singoli o multipli tramite telecomandi, pulsanti e telefoni cordless.

I servizi appena citati sono, in linea di massima, quelli più diffusi nei sistemi domotici odierni, ciò non toglie il fatto che in futuro possano subire cambiamenti o esserne introdotti nuovi, ed è proprio questa inevitabile evoluzione, accompagnata da tecnologie sempre più sofisticate e dalla ricerca di un livello di confort e benessere crescente, che fa sorgere spontanea l'idea di creare sistemi domotici complessi, aperti, eterogenei e dinamici, modellati con l'approccio Agent-Oriented.

Nell'analisi del sistema domotico D.A.S. vengono presi in esame due tipologie di servizi ben distinte:

- **gestione del confort termico**
- **sicurezza**

Sfruttando le caratteristiche di questi nuovi sistemi, l'analisi, considerando solo questi due servizi, può risultare riduttiva, ma si capirà ben presto che

per l'elevata modularità del sistema complessivo, un eventuale ampliamento delle sue funzionalità non comporta un elevato dispendio di risorse e di tempo.

Prima di analizzare i servizi offerti dal sistema D.A.S. verranno descritte le tipologie di utenze a cui è rivolto, ognuna con determinati privilegi e caratteristiche:

- **Supervisore** – chi ha la possibilità di utilizzare i tulli forniti dal sistema per il controllo della situazione dell'edificio, la configurazione e la modifica, in qualsiasi momento della giornata, del modo in cui vengono erogati i servizi. La figura del supervisore può cambiare in base al tipo di edificio su cui si fa riferimento, quindi per edifici pubblici, ad esempio per gli alberghi, sarà un addetto al monitoraggio dell'edificio che ricoprirà questa carica, oppure in ambito domestico sarà il proprietario della casa stessa.
- **Utente** – colui che usufruisce dei servizi all'interno dell'edificio, può eseguire delle configurazioni e modifiche sull'erogazione di tali servizi, in base alle proprie esigenze e ai limiti imposti dal supervisore. Volendo fare ancora un paragone tra un albergo e una casa, nel primo, l'utente alloggiando in una camera, avrà il controllo solo su quella, invece in un'abitazione il controllo sarà su tutta la casa. Il controllo non sarà totale, ma potrà avere dei cambiamenti e delle limitazioni in base alle politiche di gestione del supervisore.

Ora verranno analizzati in generale i due servizi offerti, descrivendo i compiti e i ruoli del supervisore e dell'utente.

Gestione del comfort termico - Il supervisore attraverso il sistema domotico in esame può configurare e far mantenere la temperatura in ogni stanza inibendo la regolazione manuale dell'utente, può programmare e imporre i limiti sulla temperatura per tutta la settimana, suddividendo le

giornate in fasce orarie e assegnando a ciascuna, un intervallo([temp_min – temp_max]) in modo che l'utente della stanza sia vincolato nella regolazione della temperatura, oppure assegnarli piena libertà sulla regolazione. Può modificare e controllare la potenza erogata dell'apparato di condizionamento(se tale impianto è costituito da ventilconvettori, si dovrà agire, variando la velocità del Fan Coil), imponendo anche opportuni limiti (es. limiti sulla velocità di rotazione), mantenere la potenza erogata su determinati livelli stabili nel tempo oppure lasciare la gestione in automatico al sistema.

Inoltre il sistema offre al supervisore la possibilità di sapere eventualmente in quale intervallo della giornata si trova ad operare una determinata stanza e di tale, può sapere la temperatura corrente, in quale modalità sta lavorando(calda o fredda) e a quale velocità(min,med,max). Genera allarmi se si verificano anomalie nella gestione del servizio e riconosce eventuali malfunzionamenti nei dispositivi utilizzati (pannellino utente, venticollettore e sensore temperatura).

L'utente della camera in base ai diritti concessi dal supervisore può regolare la temperatura(in estate ed inverno) e la potenza del condizionatore(velocità ventola per i ventilconvettori) in modo automatico lasciando il controllo al sistema oppure con impostazione manuale, attraverso dispositivi di comando (pannellino utente installato in camera, oppure dall'esterno con dispositivi wireless che si interfacciano al sistema).

Sicurezza - Il sistema può rilevare situazioni di pericolo o semplicemente non desiderate e malfunzionamenti dei dispositivi o impianti installati, per questo motivo le segnalazioni si raggruppano in due categorie fondamentali riferite ad ogni singola camera dell'edificio:

Warning

- Presenza di finestre aperte
- Presenza di porte aperte

Allarmi

- Fughe di gas
- Fuoriuscite di Fumo
- Malfunzionamenti dei servizi camera (gestione del comfort temperatura, sicurezza) compresi quelli dei dispositivi utilizzati per erogare i servizi stessi

Il supervisore può decidere se abilitare o meno tale servizio o parte di esso, nel senso che potrebbe aver bisogno solo di sapere il verificarsi degli allarmi e non dei warning o viceversa. Può essere aggiornato in tempo reale sullo stato di una camera e in generale il sistema è in grado di avvertire immediatamente il supervisore non appena si verificano warning o allarmi. Riceve da ogni servizio di camera(es. il gestore del confort termico o da qualsiasi altro servizio presente) eventuali allarmi di malfunzionamento, riuscendo immediatamente a rilevare rotture dei dispositivi e impianti installati per quei servizi(sensore temperatura, rilevatore di fumo, attuatore per porte e finestre,...).

Ad ogni allarme segnalato, verrà impartito in automatico o con il consenso del supervisore uno o una serie di comandi di risposta. Esempio, nel caso in cui venga impostata la modalità automatica e il sensore di fumo segnala un principio di incendio in una camera, allora il sistema è in grado di capire il tipo di problema e in quale camera si sta verificando, azionando immediatamente il dispositivo(se presente) che spruzzerà la sostanza per domare l'incendio nella camera, o avviserà prontamente il centralino della stazione dei vigili del fuoco più vicina.

Definizione delle risorse

La scelta di un approccio Agent-Oriented comporta la realizzazione del sistema domotico, come un particolare sistema multi-agente, proprio per questo anche in fase di analisi, è necessario seguire questo nuovo approccio, definendo le risorse che saranno utilizzate dagli agenti, per eseguire i loro compiti individuali e sociali. Le risorse sono definite in base al servizio che erogano, al loro modo di accesso, alle informazioni richieste per erogare il servizio e a quelle che vengono rilasciate durante l'invocazione del servizio.

Il D.A.S. per poter erogare agli utenti i servizi descritti in generale precedentemente (nel capitolo precedente), si servirà di particolari entità (agenti) indipendenti che usufruiranno delle seguenti risorse.

Setup e controllo camera

Rende disponibile le informazioni più aggiornate per la gestione dei servizi generali (gestione del comfort termico, sicurezza,...) di ogni camera dell'edificio, per ognuno di essi, le informazioni, vengono raggruppate secondo quattro categorie: comandi, parametri, configurazione e stato.

In generale i comandi, i parametri e le configurazioni dei vari servizi di una stanza vengono forniti dal supervisore e inviati alle risorse associate, viceversa lo stato dei servizi verrà fornito dalle risorse associate di camera e rimarrà nella risorsa Setup e controllo camera fino a quando il supervisore non intende leggerlo.

Elenco delle informazioni rese disponibili da questa risorsa per ogni camera:

Gestione del comfort termico

Comandi

- On Off servizio gestione comfort termico – con questo comando il supervisore può decidere se abilitare o meno il servizio.
- Temperatura impostata supervisore – il supervisore dall'esterno può fissare la temperatura di una camera, inibendo la sua regolazione dall'interno.
- On Off intervalli condizionatore – viene abilitata o meno la suddivisione temporale per la programmazione dei limiti minimi e massimi della temperatura nell'arco di una settimana.
- Obbliga modalità di lavoro condizionatore – può essere fissata la modalità di lavoro del condizionatore, inibendo la regolazione dall'interno della camera.
- Modalità di lavoro del condizionatore non consentite – con questo comando vengono proibite alcune modalità di funzionamento del condizionatore.

Parametri

- Valore degli intervalli temporali – suddividono l'arco della giornata e possono essere definiti per tutta la settimana. I limiti sul numero e sulla dimensione degli intervalli non vengono prefissati, ma possono variare in base alle esigenze del supervisore.
- Valore della temperatura per ogni intervallo temporale – ad ogni intervallo temporale vengono affiancati i limiti sulla temperatura impostabile nella camera.
- Valore della temperatura impostata dal supervisore – definisce il valore della temperatura fissata dal supervisore.
- Valore modalità di funzionamento obbligata – specifica la modalità di lavoro del condizionatore imposta dal supervisore.
- Valore modalità di funzionamento non consentita – identifica quali modalità di lavoro non possono essere raggiunte dal condizionatore.

Configurazione

- Condizionatore inverno/estate – indica se il condizionatore deve lavorare in estate o in inverno, rispettivamente rinfrescando o riscaldando il locale.
- Isteresi temperatura – valore entro il quale variazioni di temperatura non comportano variazioni della modalità di lavoro del condizionatore.
- Presenza condizionatore – stabilisce se all'interno della stanza deve esserci o meno un condizionatore funzionante.
- Presenza lettore temperatura – stabilisce la presenza di un sensore per la rilevazione della temperatura.
- Presenza pannello – stabilisce la presenza di un pannello per dare la possibilità all'utente di camera di impostare il servizio gestione comfort termico, nei limiti imposti dal supervisore.

Stato

- Modalità di lavoro condizionatore – modalità di lavoro corrente del condizionatore di camera.
- Valore della temperatura della camera – temperatura corrente dell'interno camera rilevata da un sensore specifico.
- Condizionatore inverno/estate - indica se il condizionatore rinfresca o riscalda il locale.

Sicurezza

Comandi

- On Off servizio Sicurezza - il supervisore può decidere se abilitare o meno il servizio.
- On Off warning – richiesta di segnalazione warning.

- On Off allarmi – richiesta di segnalazione allarmi.

Parametri

- Codice identificativo stanza

Configurazione

- Presenza rilevatore fughe di gas - stabilisce la presenza di un sensore per la rilevazione di fughe di gas.
- Presenza rilevatore fumo - stabilisce la presenza di un sensore per la rilevazione di fumo.
- Presenza rilevatore porte aperte - stabilisce la presenza di un sensore per la rilevazione di porte aperte.
- Presenza rilevatore finestre aperte - stabilisce la presenza di un sensore per la rilevazione di finestre aperte.

Stato

- Stato degli allarmi di camera.
- Stato dei warning di camera.

Controllo e gestione temperatura camera

Risorsa associata al servizio gestione comfort termico, fornisce i comandi per la regolazione del funzionamento del condizionatore per ottenere la temperatura desiderata nella stanza, rispettando tutti i comandi, parametri e configurazioni impartiti sia dal supervisore che dall'utente. Inoltre capisce se accadano malfunzionamenti a quei dispositivi installati in camera per erogare il servizio segnalandoli tempestivamente al servizio sicurezza che si preoccuperà di gestirli.

Elenco delle informazioni rilasciate dalla risorsa:

- Comandi, parametri, configurazioni riguardanti il servizio gestione comfort termico.
- Temperatura impostata dall'utente in camera.

- Modalità di funzionamento del condizionatore, impostata dall'utente in camera.
- Temperatura attuale.
- Eventuali malfunzionamenti del servizio e dei dispositivi installati - vengono rilevate anomalie nei dispositivi legati al servizio gestione comfort termico e segnalate al servizio sicurezza.

Elenco delle informazioni rilasciate dalla risorsa, che fungono da comandi per il condizionatore di camera:

- Modalità di funzionamento condizionatore – comanda la modalità di funzionamento del condizionatore(spento, automatico, manuale) .
- Condizionatore inverno/estate – stabilisce se il condizionatore deve riscaldare o rinfrescare l'ambiente.

Gestione Allarmi e Warning della singola camera

Risorsa associata al servizio Sicurezza, rende disponibile lo stato di quei dispositivi che segnalano eventuali allarmi o warning all'interno di una stanza. Per allarmi si intendono particolari situazioni di pericolo (fughe di gas, principi di incendi,...), in genere quelle situazioni in cui l'intervento deve essere immediato, invece i warning sono semplicemente delle avvertenze che il sistema segnala senza troppi allarmismi. Il servizio è localizzato all'interno della camera e quindi le informazioni che produce non sono ancora accessibili al supervisore, ma dovranno essere maggiormente elaborate e gestite.

Elenco delle informazioni rese disponibili da questa risorsa per ogni camera:

- Comandi, parametri, configurazioni riguardanti il servizio sicurezza.
Stato Warning

- Stato finestre – indica se le finestre sono aperte o chiuse.
- Stato porte – indica se le porte sono aperte o chiuse.

Stato Allarmi

- Stato rilevatore gas – se il sensore è in allarme o meno.
- Stato rilevatore fumo – se il sensore è in allarme o meno.
- Stato di salute dei dispositivi – per ogni dispositivo associato ai servizi presenti in camera viene rilasciato lo stato di salute.

Gestione generale Allarmi e Warning

Risorsa associata al servizio Sicurezza, memorizza gli allarmi e i warning provenienti da ogni camera dell'edificio, associando ad ognuno la data e l'ora di rilevazione, costituendo una "centrale" di allarmi. Oltre alla memorizzazione, le anomalie vengono immediatamente segnalate al supervisore, che le gestirà nel modo più opportuno.

Elenco delle informazioni richieste per erogare il servizio:

- Le informazioni richieste sono quelle rese disponibili dalle risorse Gestione Allarmi e Warning di ogni camera.

Elenco delle informazioni rilasciate dalla risorsa:

- Informazione riguardante il tipo di anomalia, la data e l'ora in cui è stata rilevata e da quale stanza proviene.

Definizione dei Gruppi

I gruppi costituiscono assieme alle risorse, una parte fondamentale del sistema domotico, sono formati da un insieme di entità indipendenti, che collaborano per l'adempimento di particolari ruoli sociali. Vengono definiti in base alle operazioni sociali che devono compiere, i loro permessi per accedere alle risorse, e attraverso le regole di interazione tra loro e le risorse (utilizzate per il raggiungimento dei ruoli stabiliti).

Gestione generale informazioni camera

Ruolo sociale – Permette al supervisore di impostare e di trasmettere i comandi, i parametri, e la configurazione di tutti i servizi alle risorse Setup e controllo camera di ogni stanza (al momento i servizi sono: gestione del comfort termico e sicurezza) . Può prelevare lo stato di ogni servizio di camera, per poter realizzare statistiche temporali sull'andamento del sistema. Tutto questo avviene attraverso lo scambio di informazioni con le risorse *Setup e controllo camera*.

Risorse associate:

- Setup e controllo camera

Gestione comfort termico interno camera

Ruolo sociale - questo gruppo ha il compito fondamentale di gestire il comfort termico all'interno di una camera, scambiando ed elaborando le informazioni prelevate dalla risorsa *Controllo e gestione temperatura camera*, in modo da poter comandare il funzionamento del condizionatore secondo le specifiche richieste. Inoltre si preoccupa di aggiornare lo stato del servizio Gestione comfort termico, reso disponibile dalla risorsa *Setup e controllo camera*.

Risorse associate:

- Controllo e gestione temperatura camera
- Setup e controllo camera

Sicurezza interno camera

Ruolo sociale – Rileva tutti gli allarmi e i warning che possono verificarsi all'interno della stanza. In base ai comandi e alle impostazioni del supervisore, riguardanti il servizio Sicurezza, messi a disposizione dalla risorsa *Gestione Allarmi e Warning della singola camera*, vengono attuate politiche di filtraggio sulle tipologie di allarmi o warning segnalati, in altre parole per ogni stanza vengono stabilite, tra tutte le situazioni di pericolo,

quelle effettivamente da segnalare. La segnalazione avviene tramite l'aggiornamento dello stato del servizio Sicurezza, reso disponibile dalla risorsa *Setup e controllo camera* e l'aggiunta di una nuova informazione nel centro di allarmi della risorsa *Gestione generale Allarmi e Warning*.

Risorse associate:

- Gestione Allarmi e Warning della singola camera
- Setup e controllo camera
- Gestione generale Allarmi e Warning

Sicurezza generale

Ruolo sociale – Gestisce gli allarmi e i warning di tutto l'edificio, resi disponibili dalla risorsa *Gestione generale Allarmi e Warning*, attuando operazioni di intervento circoscritte alle camere in cui si verificano oppure segnalando semplicemente al supervisore le anomalie rilevate.

Risorse associate:

- Gestione generale Allarmi e Warning

Definizione dei Ruoli

Verranno analizzati i ruoli in base alle operazioni individuali che compiono, alle informazioni richieste o cedute alle risorse per portarle a termine e ai gruppi di appartenenza.

Ogni ruolo del sistema fa parte di uno dei gruppi descritti nel capitolo precedente, quindi di seguito verranno suddivisi in merito a tali gruppi di appartenenza.

Ruoli individuali del gruppo *Gestione generale informazioni camera*:

- **Imposta comandi, parametri, configurazioni**

Operazione individuale – Permette al supervisore di impostare i comandi, i parametri e le configurazioni di ogni servizio che diventeranno parte delle informazioni rilasciate dalla risorsa *Setup e controllo camera*.

Risorse associate:

- Setup e controllo camera

Informazioni cedute alla risorsa:

- I comandi, parametri e le configurazioni dei servizi *Gestione comfort termico* e *Sicurezza*, in generale le informazioni necessarie per erogare ogni servizio presente in camera.

- **Aggiorna stato servizi**

Operazione individuale – Permette al supervisore di conoscere lo stato corrente di ogni servizio di una determinata camera, prelevando le informazioni dalla risorsa *Setup e controllo camera*, che si preoccuperà di mantenerle costantemente aggiornate.

Risorse associate:

- Setup e controllo camera

Informazioni prelevate dalla risorsa:

- Lo stato dei servizi *Gestione comfort termico* e *Sicurezza* localizzati nella camera selezionata dal supervisore.

Ruoli individuali del gruppo *Gestione comfort termico interno camera*:

- **Gestore Temperatura**

Operazione individuale – Preleva le informazioni relative al servizio *Gestione comfort termico*, dalla risorsa *Controllo e gestione temperatura camera*, elabora tali informazioni in modo da

comandare, secondo le impostazioni del supervisore e alle esigenze dell'utente di camera, il funzionamento del condizionatore. Inoltre segnala lo stato del servizio *Gestione comfort termico* alla risorsa *Setup e controllo camera*.

Risorsa associata:

- Setup e controllo camera.
- Controllo e gestione temperatura camera

Informazioni prelevate dalla risorsa *Controllo e gestione temperatura camera* :

- I comandi, parametri e le configurazioni esclusivamente riguardanti il servizio *Gestione comfort termico*.
- Temperatura impostata dall'utente in camera.
- Modalità di funzionamento del condizionatore, impostata dall'utente in camera.
- Temperatura attuale.

Informazioni cedute alla risorsa *Controllo e gestione temperatura camera*:

- Modalità di funzionamento condizionatore – comanda la modalità di funzionamento del condizionatore(spento, automatico, manuale) .
- Condizionatore inverno/estate – stabilisce se il condizionatore deve riscaldare o rinfrescare l'ambiente.

Informazioni cedute alla risorsa *Setup e controllo camera*:

- Stato del servizio *Gestione del comfort termico*.

- **Sensore Temperatura**

Operazione individuale – Spedisce il valore della temperatura corrente della camera, e lo stato di salute del sensore a cui è associato, alla risorsa *Controllo e gestione temperatura camera*.

Risorsa associata:

- Controllo e gestione temperatura camera

Informazioni cedute alla risorsa:

- Temperatura attuale camera
- Eventuali malfunzionamenti del sensore a cui è associato

- **Impostazioni pannellino**

Operazione individuale – Associato al pannellino utente della camera, spedisce le informazioni impostate dall'utente per il funzionamento del condizionatore e lo stato di salute del dispositivo alla risorsa *Controllo e gestione temperatura camera*.

Risorsa associata:

- Controllo e gestione temperatura camera

Informazioni cedute alla risorsa:

- Temperatura impostata.
- Modalità di funzionamento del condizionatore.
- Eventuali malfunzionamenti del pannellino utente a cui è associato

- **Condizionatore**

Operazione individuale – Imposta il funzionamento del condizionatore di camera in base alle informazioni prelevate dalla risorsa *Controllo e gestione temperatura camera*.

Risorsa associata:

- Controllo e gestione temperatura camera

Informazioni prelevate dalla risorsa:

- Modalità di funzionamento condizionatore (spento, automatico, manuale) .
- Condizionatore inverno/estate – stabilisce se il condizionatore deve riscaldare o rinfrescare l'ambiente.

Ruoli individuali del gruppo *Sicurezza interno camera*:

- **Gestore Allarmi**

Operazione individuale – Preleva tutte le informazioni rilasciate dalla risorsa *Gestione Allarmi e Warning della singola camera*, elabora tali informazioni in modo da far conoscere istantaneamente al supervisore, solo gli allarmi o warning da lui richiesti. Gli allarmi e warning vengono assegnati alla risorsa *Setup e controllo camera* come se fossero lo stato del servizio *Sicurezza* e alla “centrale allarmi”, della risorsa *Gestione generale Allarmi e Warning*.

Risorse associate:

- Gestione Allarmi e Warning della singola camera
- Setup e controllo camera
- Gestione generale Allarmi e Warning

Informazioni prelevate dalla risorsa *Gestione Allarmi e Warning della singola camera* :

- Insieme completo delle informazioni rilasciate dalla risorsa(impostazione per il servizio Sicurezza, allarmi e warning rilevati).

Informazioni cedute alla risorsa *Setup e controllo camera*:

- Stato degli allarmi di camera.
- Stato dei warning di camera.

Informazioni cedute alla risorsa *Gestione generale Allarmi e Warning*:

- l'allarme o il warning verificatosi nella stanza, associando il codice identificativo della camera.

- **Rilevatore fughe di gas**

Operazione individuale – Spedisce l'allarme dell'eventuale fuga di gas e lo stato di salute del sensore a cui è associato, alla risorsa *Gestione Allarmi e Warning della singola camera*.

Risorse associate:

- Gestione Allarmi e Warning della singola camera

Informazioni cedute alla risorsa:

- Allarme rilevato dal sensore, dell'eventuale fuga di gas in camera.
- Eventuali malfunzionamenti del sensore a cui è associato.

- **Rilevatore Fumo**

Operazione individuale — Spedisce l'allarme dell'eventuale principio d'incendio e lo stato di salute del sensore a cui è associato, alla risorsa *Gestione Allarmi e Warning della singola camera*.

Risorse associate:

- Gestione Allarmi e Warning della singola camera

Informazioni cedute alla risorsa:

- Allarme rilevato dal sensore, dell'eventuale principio d'incendio in camera.
- Eventuali malfunzionamenti del sensore a cui è associato.

- **Rilevatore porte aperte**

Operazione individuale – Spedisce il warning della presenza di porte aperte e lo stato di salute del sensore a cui è associato, alla risorsa *Gestione Allarmi e Warning della singola camera*.

Risorse associate:

- Gestione Allarmi e Warning della singola camera

Informazioni cedute alla risorsa:

- Warning rilevato dal sensore, per la presenza di porte aperte in camera.
- Eventuali malfunzionamenti del sensore a cui è associato.

- **Rilevatore finestre aperte**

Operazione individuale – Spedisce il warning della presenza di finestre aperte e lo stato di salute del sensore a cui è associato, alla risorsa *Gestione Allarmi e Warning della singola camera*.

Risorse associate:

- Gestione Allarmi e Warning della singola camera

Informazioni cedute alla risorsa:

- Warning rilevato dal sensore, per la presenza di finestre aperte in camera.
- Eventuali malfunzionamenti del sensore a cui è associato.

Ruoli individuali del gruppo *Sicurezza generale*:

- **Aggiorna allarmi**

Operazione individuale – Gestisce tempestivamente gli allarmi e i warning di tutte le camere, provenienti dalla risorsa *Gestione generale Allarmi e Warning*, avvertendo il supervisore e pianificando opportune soluzioni per rimediare la situazione di pericolo.

Risorse associate:

- Gestione generale Allarmi e Warning

Informazioni prelevate dalla risorsa:

- Informazione riguardante il tipo di anomalia, la data e l'ora in cui è stata rilevata e da quale stanza proviene.

Capitolo 4 – D.A.S. progetto

Modello del sistema domotico

Il sistema domotico D.A.S per poter erogare i servizi generali analizzati in precedenza, viene diviso in due parti fondamentali: il **Gestore Supervisore** e il **Gestore Camera**. Quest'ultimo fa riferimento al controllo dei servizi di un'unica stanza, quindi il numero di gestori camera utilizzati è pari al numero delle stanze che si intendono automatizzare. Invece il gestore supervisore è unico per tutto il sistema e ha il compito di impostare

e controllare ogni gestore camera. L'utente avrà accesso solo a quei gestori camera associati alle stanze in cui vive, invece il supervisore lavorerà sfruttando le funzionalità del gestore supervisore.

Questa suddivisione è puramente logica, in realtà il sistema domotico viene realizzato come un particolare sistema multi-agente, costituito da parti indipendenti, autonome e con determinati ruoli individuali, che collaborano tra loro attraverso lo scambio di informazioni con i media di coordinazione (le risorse). Le entità indipendenti sono a tutti gli effetti agenti, che vengono raggruppati in diverse società le quali condividono i ruoli sociali definiti in fase di analisi. In definitiva le due tipologie di gestori oltre a riassumere i ruoli sociali e le loro dipendenze con le risorse del sistema, identificano i nodi del sistema introdotti dalla metodologia TuCSoN. Più precisamente in tutte e due le tipologie di Gestori è presente un unico nodo, però non si esclude il fatto che eventuali ampliamenti del sistema, possano far nascere la necessità di introdurre più nodi in ciascuno dei Gestori.

Attualmente:

- Il Gestore Supervisore include il nodo ***supervisore***
- Il Gestore Camera include il nodo ***camera_id***
id rappresenta il codice identificativo della camera

In figura 1 viene presentato il modello generale del sistema, considerando un edificio con solo due camere automatizzate, evidenziando la posizione dei vari gestori, e i collegamenti tra i gruppi e le risorse di cui sono formati.

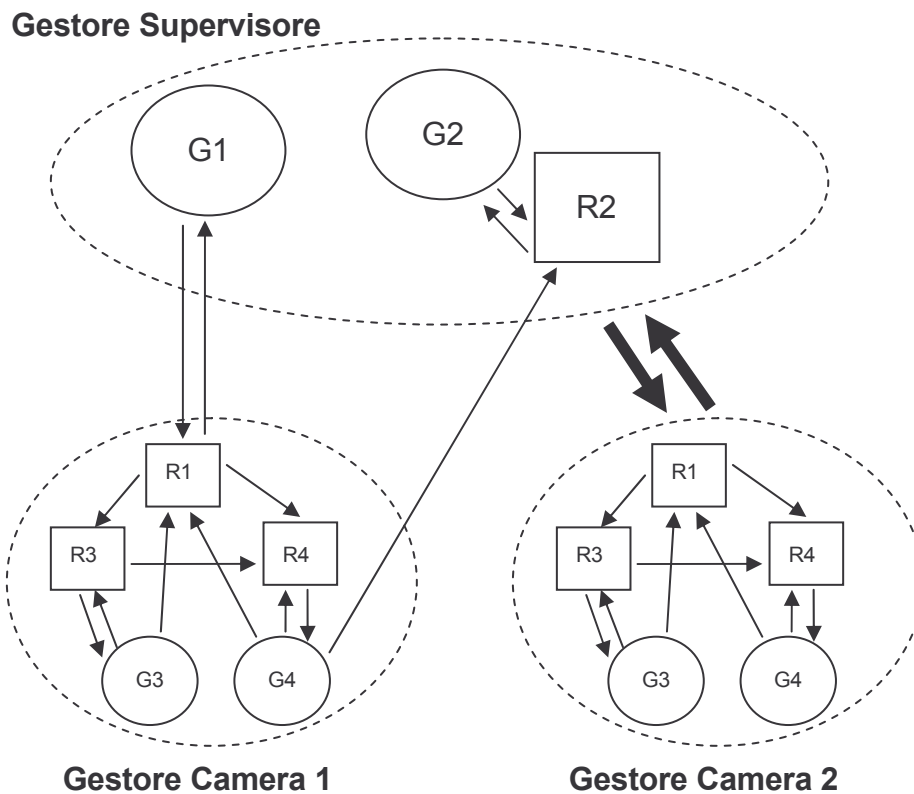


Fig.1 modello generale del sistema D.A.S.

Legenda:

Gruppi:

- G1 – Gestione generale informazioni camera
- G2 – Sicurezza generale
- G3 – Gestione comfort termico interno camera
- G4 – Sicurezza interno camera

Risorse:

- R1 – Setup e Controllo camera
- R2 – Gestione generale Allarmi e Warning

- R3 – Controllo e gestione temperatura camera
- R4 – Gestione Allarmi e Warning della singola camera

In fase di progetto vengono adottate le metodologie proposte dall'infrastruttura software di coordinazione TuCSoN, proprio per questo ora è importante definire il comportamento dei tuple center, ovvero i coordination media associati ai nodi TuCSoN, che applicano le caratteristiche delle risorse definite nell'analisi, e le classi di agenti, in base ai gruppi di appartenenza.

Coordination media

Le risorse definite in fase di analisi vengono ora associate ai tuple center che ne implementano le caratteristiche.

Tuple center che implementa le caratteristiche della risorsa Setup e Controllo camera:

Nome – **servizi_camera**

Posizione occupata nel sistema - si trova all'interno del Gestore Camera ed è associato al nodo camera_id, governa lo scambio di informazioni, relative alla gestione dei servizi, tra il Gestore Camera stesso e il Gestore Supervisore.

Come vengono implementate le caratteristiche della risorsa - mantiene le informazioni(tuple) coerenti col sistema generale, ogni informazione vecchia viene sostituita da quella nuova, dello stesso tipo, cioè si dovrà evitare il caso in cui ci siano più tuple di uno stesso tipo nello stesso istante. In questo modo, chi preleverà le informazioni sarà sicuro di aver letto quelle più aggiornate.

Inoltre smista i comandi, parametri e configurazioni di ogni servizio, ai tuple center *temperatura_camera* e *allarmi_camera*, inviando rispettivamente le tuple:

```

servizio_comfort_temp( info( cmd(...), par(...), cfg(...) ))
    cmd( OnOffServizioTemp,
        SetSpvTemp,
        OnOffIntervalli,
        CondForceSpeed,
        CondNeverSpeed )

    par( SpvTemp,
        ValForceSpeed,
        ValNeverSpeed,
        intervalli([StartTime, EndTime, StartTemp, EndTemp],
            [...],...)

    cfg( InvernoEstate,
        IsteresiTemp,
        IsOnCond,
        IsOnTempReader,
        IsOnUserPanel)

```

Comandi

OnOffServizioTemp – flag, valore 0 servizio disabilitato, 1 abilitato

SetSpvTemp – flag, valore 0 temperatura impostata dall'utente, 1 supervisore.

OnOffIntervalli – flag, valore 0 suddivisione temporale disabilitata, 1 abilitata.

CondForceSpeed – flag, valore 0 libera impostazione per l'utente, 1 velocità condizionatore fissata dal supervisore.

CondNeverSpeed – flag, valore 0 libera impostazione per l'utente, 1 non consente all'utente di selezionare determinate velocità.

Parametri

SpvTemp – valore della temperatura impostata dal supervisore
ValForceSpeed – valore della velocità(min,media,max) del condizionatore fissata dal supervisore.

ValNeverSpeed – valore della velocità non consentita.

Intervalli([...]) – suddivisione temporale della giornata, ogni intervallo è costituito dal valore della temperatura minima, massima, e l'orario di inizio e fine.

Configurazione

InvernoEstate – flag, valore 0 indica che il condizionatore lavora in inverno, 1 in estate.

IsteresiTemp – valore entro il quale variazioni di temperatura non comportano variazioni della modalità di lavoro del condizionatore.

IsOnCond – flag, valore 0 condizionatore non presente in camera, 1 presente.

IsOnTempReader – flag, valore 0 sensore temperatura non presente in camera, 1 presente.

IsOnUserPanel – flag, valore 0 pannello utente non presente in camera, 1 presente.

servizio_sicurezza(info(cmd(...), par(...), cfg(...)))

*cmd(OnOffServizioSicurezza,
OnOffWarning,
OnOffAllarmi)*

par(CodCamera)

*cfg(IsOnGas,
IsOnFumo,
IsOnPorte,
IsOnFinestre)*

Comandi

OnOffServizioSicurezza flag, valore 0 servizio disabilitato, 1 abilitato.

OnOffWarning – flag, valore 0 warning disabilitati, 1 abilitati.

OnOffAllarmi – flag, valore 0 warning disabilitati, 1 abilitati.

Parametri

CodCamera – codice identificativo di camera

Configurazione

IsOnGas - flag, valore 0 rilevatore fughe di gas non presente in camera, 1 presente.

IsOnFumo - flag, valore 0 rilevatore di fumo non presente in camera, 1 presente.

IsOnPorte - flag, valore 0 rilevatore porte aperte non presente in camera, 1 presente.

IsOnFinestre - flag, valore 0 rilevatore finestre aperte non presente in camera, 1 presente.

con i comandi:

```
out_tc(temperatura_camera, servizio_comfort_temp( info(...) ))
```

```
out_tc(allarmi_camera, servizio_sicurezza( info(...) ))
```

Classi di agenti coordinati dal tuple center:

- A1 - Imposta comandi, parametri, configurazioni
 - A2 - Aggiorna stato servizi
 - A3 - Gestore Temperatura
 - A7 - Gestore Allarmi
-

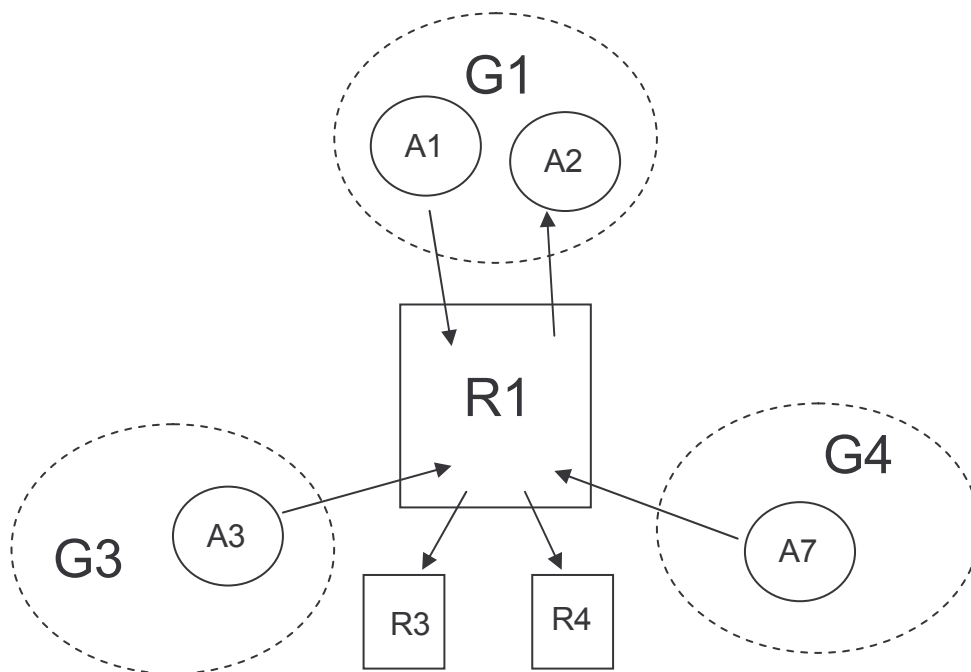


Fig. 2 Tuple Center servizi_camera e agenti coordinati

Tuple center che implementa le caratteristiche della risorsa Controllo e gestione temperatura camera:

Nome – **temperatura_camera**

Posizione occupata nel sistema – si trova all’interno del Gestore Camera ed è associato al nodo camera_id, governa lo scambio di informazioni relative al servizio Gestione del comfort termico e capisce se accadano malfunzionamenti ai dispositivi installati per erogare il servizio, segnalandoli tempestivamente al tuple center *allarmi_camera*, il quale si occuperà di gestirli.

Come vengono implementate le caratteristiche della risorsa – come il tuple center precedente, deve mantenere le informazioni(tuple) sempre aggiornate, sostituendo le tuple vecchie con quelle nuove, dello stesso tipo. Inoltre controlla regolarmente l’arrivo delle tuple che attestano il

funzionamento corretto di ogni dispositivo, se solamente una non dovesse pervenire allora segnala il malfunzionamento, inviando la tupla:

allarme(servizio_comfort_temp(CodDispositivo))

CodDispositivo - rappresenta il codice univoco del dispositivo rotto, riconosciuto dal sistema.

al tuple center *allarmi_camera*, col comando:

out_tc(allarmi_camera, allarme(servizio_comfort_temp(CodDispositivo)))

Classi di agenti coordinati dal tuple center:

- A3 - Gestore Temperatura
- A4 - Sensore Temperatura
- A5 - Impostazioni pannellino
- A6 – Condizionatore

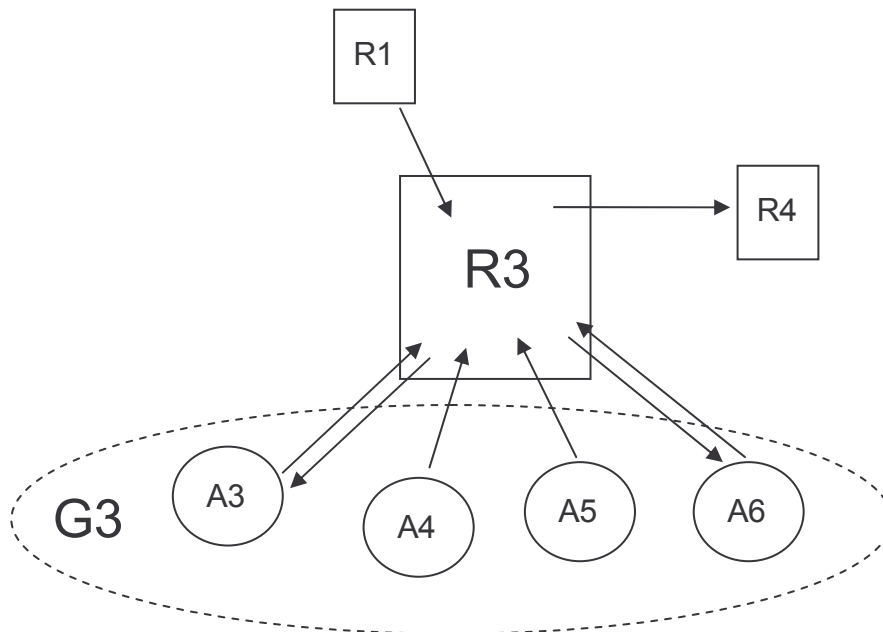


Fig. 3 Tuple Center temperatura_camera e agenti coordinati

Tuple center che implementa le caratteristiche della risorsa Gestione Allarmi e Warning della singola camera:

Nome – **allarmi_camera**

Posizione occupata nel sistema – si trova all'interno del Gestore Camera ed è associato al nodo camera_id, governa lo scambio di informazioni relative al servizio Sicurezza(Raccoglie gli allarmi e warning relativi alla singola camera) e capisce se accadano malfunzionamenti ai dispositivi installati per erogare tale servizio.

Come vengono implementate le caratteristiche della risorsa – come i tuple center precedenti, deve mantenere le informazioni(tuple) sempre aggiornate, sostituendo le tuple vecchie con quelle nuove, dello stesso tipo. Inoltre controlla regolarmente, se arrivano le tuple che attestano il

funzionamento corretto di ogni dispositivo se solamente una non dovesse pervenire allora aggiunge a se stesso la tupla che segnala il malfunzionamento:

allarme(servizio_sicurezza(CodDispositivo))

CodDispositivo - rappresenta il codice univoco del dispositivo rotto, riconosciuto dal sistema.

col comando:

out_r(allarme(servizio_sicurezza(CodDispositivo)))

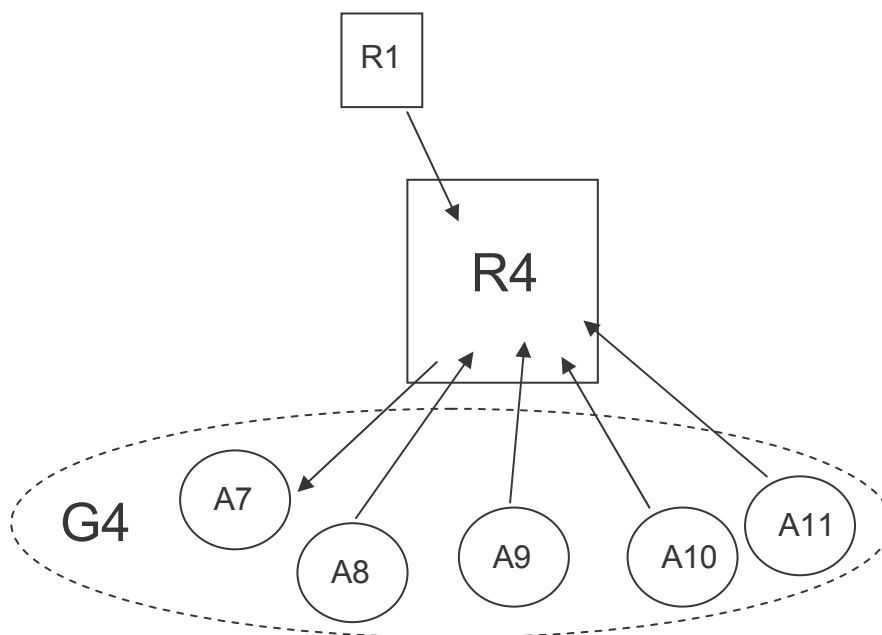


Fig. 4 Tuple Center allarmi_camera e agenti coordinati

Classi di agenti coordinati dal tuple center:

- A7 - Gestore Allarmi
- A8 - Rilevatore fughe di gas
- A9 - Rilevatore Fumo
- A10 - Rilevatore porte aperte
- A11 - Rilevatore finestre aperte

Tuple center che implementa le caratteristiche della risorsa Gestione generale Allarmi e Warning:

Nome – **centrale_allarmi**

Posizione occupata nel sistema – si trova all'interno del Gestore Supervisore ed è associato al nodo supervisore, memorizza tutti gli allarmi e warning del sistema, avvisando anche l'agente Aggiorna allarmi dell'anomalia in corso.

Come vengono implementate le caratteristiche della risorsa – ad ogni tupla di allarme o warning proveniente dalle varie camere, vengono associate informazioni sulla data e l'ora di rilevamento. Queste nuove tuple devono rimanere memorizzate nel tuple center e allo stesso tempo essere disponibili per la lettura da parte dell'agente Aggiorna allarmi. La tupla destinata all'agente ha la seguente struttura:

```
incoming_alarm(
  DataOra,
  allarmi_warning(
    CodCamera,
    CodDispositivo,
    servizio_comfort_temp(CodDispositivo),
    servizio_sicurezza(CodDispositivo) ))
```

DataOra – rappresenta l'orario in cui è stata rilevazione l'anomalia

CodCamera – codice identificativo della camera in cui si è verificata l'anomalia.

CodDispositivo – codice del dispositivo che ha segnalato l’anomalia, il sistema tramite questo codice capisce se si tratta di un allarme o di un warning.

servizio_comfort_temp(CodDispositivo) – segnala la rottura del dispositivo, identificato da CodDispositivo, facente parte del servizio Gestione del comfort termico.

servizio_sicurezza(CodDispositivo) - segnala la rottura del dispositivo, identificato da CodDispositivo, facente parte del servizio Sicurezza.

Queste ultime tre informazioni, all’interno della tupla, possono essere replicate più volte in base a quanti allarmi o warning si sono verificano contemporaneamente. Può anche capitare che ce ne sia solamente una delle tre.

E viene creata col comando:

```
out_r(incoming_alarm(...))
```

Classi di agenti coordinati dal tuple center:

- A7 - Gestore Allarmi
- A12 - Aggiorna allarmi

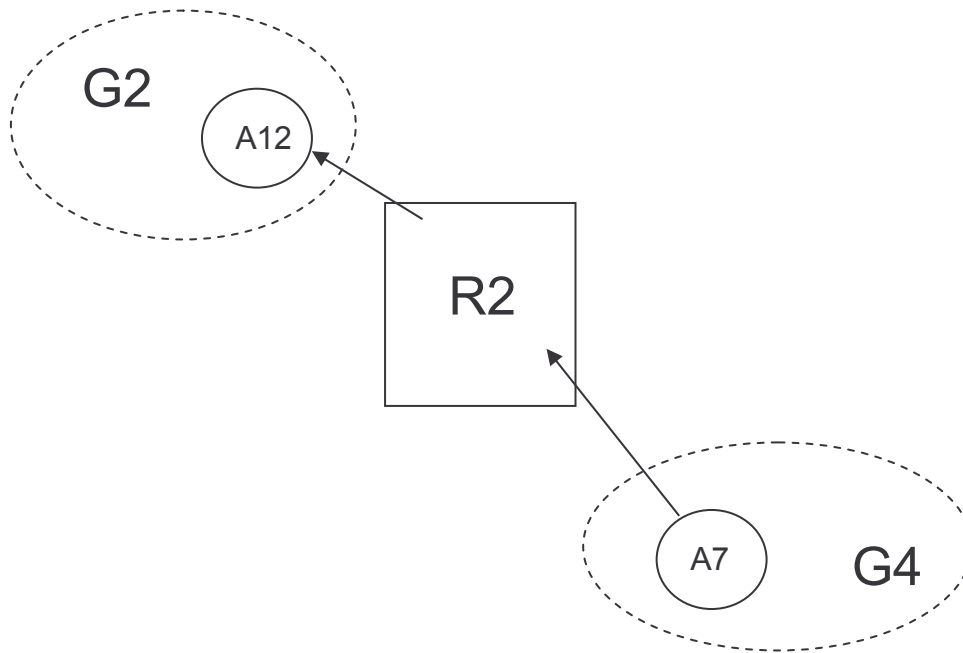


Fig. 5 Tuple Center centrale_allarmi e agenti coordinati

Classi di Agenti

I ruoli individuali definite in fase di analisi vengono ora associati a classi di Agenti che ne implementano le caratteristiche.

Le classi di agenti assumono lo stesso nome dei ruoli individuali a cui fanno riferimento.

Classe di Agenti che implementa le caratteristiche del ruolo individuale **A1-Imposta comandi, parametri, configurazioni:**

Posizione occupata nel sistema - si trova all'interno del Gestore Supervisore, più precisamente nel gruppo G1 (Gestione generale informazioni camera).

Ruolo individuale - permette al supervisore di impostare e spedire i comandi, parametri e le configurazioni di ogni servizio al tuple center servizi_camera. Il supervisore avrà a disposizione la lista delle camere automatizzate del sistema e tramite opportune interfacce potrà selezionare quella di cui vuole impostare le caratteristiche. Questo agente essendo legato ad una determinata stanza, terminerà non appena il supervisore passerà il controllo a quella successiva.

La tupla contenente le informazioni sui servizi ha la forma:

```
info_servizi( servizio_comfort_temp( info(...) ),  
servizio_sicurezza( info(...) ) )
```

Il contenuto delle tuple servizio_comfort_temp e servizio_sicurezza è già stato illustrato all'interno della definizione del tuple center servizi_camera.

e verrà spedita al tuple center servizi_camera con il comando:

```
servizi_camera@camera_id?out(info_servizi( ... ) )
```

viene utilizzato il nome assoluto perché il tuple center di destinazione risiede in un nodo diverso da quello del mittente.

Classe di Agenti che implementa le caratteristiche del ruolo individuale **A2-
Aggiorna stato servizi:**

Posizione occupata nel sistema - si trova all'interno del Gestore Supervisore, più precisamente nel gruppo G1 (Gestione generale informazioni camera).

Ruolo individuale - permette al supervisore di conoscere lo stato corrente dei servizi di ogni camera, prelevando le informazioni al tuple center servizi_camera. Il supervisore avrà a disposizione la lista delle camere automatizzate del sistema e tramite opportune interfacce potrà selezionare quella di cui vuole conoscere lo stato, a questo punto l'agente leggerà

continuamente le informazioni aggiornate, dal tuple center servizi_camera. Questo agente essendo legato ad una determinata stanza, terminerà non appena il supervisore passerà il controllo a quella successiva.

Le tuple contenenti le informazioni sullo stato dei servizi hanno la forma:

```
stato_servizi(  
    servizio_comfort_temp( ModFunzCon,  
                           TempAttuale,  
                           InvernoEstate ) )
```

ModFunzCon – modalità di lavoro corrente del condizionatore di camera:

ModFunzCon = 0 , condizionatore spento

ModFunzCon = 10, modalità manuale velocità minima

ModFunzCon = 20, modalità manuale velocità media

ModFunzCon = 30, modalità manuale velocità massima

ModFunzCon = 40, modalità automatica

TempAttuale – temperatura corrente dell'interno camera rilevata dal sensore.

InvernoEstate - indica se il condizionatore rinfresca riscalda il locale.

```
stato_servizi( servizio_sicurezza(  
    allarmi_warning([ CodDispositivo,StatoDispositivo],..)))
```

Ogni lista è associata ad un dispositivo installato in camera, favorisce il CodDispositivo e lo StatoDispositivo, il primo definisce il

tipo di dispositivo e quindi se si sta facendo riferimento ad un allarme o ad un warning e il secondo rappresenta lo stato.

I dispositivi che rilevano allarmi hanno un codice identificativo minore di 100:

CodDispositivo = 0 , sensore temperatura

CodDispositivo = 10, pannellino utente

CodDispositivo = 20, condizionatore

CodDispositivo = 30, rilevatore di gas

CodDispositivo = 40, rilevatore di fumo

I dispositivi che rilevano warning hanno un codice identificativo maggiore di 100:

CodDispositivo = 100, rilevatore di porte aperte

CodDispositivo = 110, rilevatore di finestre aperte

StatoDispositivo = 0, dispositivo rotto

StatoDispositivo = 10, dispositivo funzionante in attesa

StatoDispositivo = 20, dispositivo funzionante in allarme

e verranno lette dal tuple center servizi_camera con il comando:

```
servizi_camera@camera_id?rdp(  
    stato_servizi(  
        servizio_comfort_temp(...)))
```

```
servizi_camera@camera_id?rdp(  
    stato_servizi(  
        servizio_sicurezza(...)))
```

Classe di Agenti che implementa le caratteristiche del ruolo individuale **A3-Gestore Temperatura:**

Posizione occupata nel sistema - si trova all'interno del Gestore Camera, più precisamente nel gruppo G3 (Gestione comfort termico interno camera).

Ruolo individuale – Questo agente sarà sempre attivo e leggerà costantemente le informazioni sui comandi, i parametri e le configurazioni inerenti al servizio Gestione del comfort termico dal tuple center temperatura_camera con i comandi:

temperatura_camera?rdp(servizio_comfort_temp(info(...)))

*temperatura_camera?rdp(pannellino(TempImpostata,
ModFunzCon))*

temperatura_camera?rdp(sensoretemp(TempAttuale))

dopo aver elaborato nel modo più consono le informazioni lette, tenendo presente che le informazioni provenienti dal supervisore hanno priorità maggiore rispetto a quelle dell'utente di camera, viene spedita una tupla con l'esatto comando di funzionamento per il condizionatore al tuple center temperatura_camera:

temperatura_camera?out(fancoil(InvernoEstate, ModFunzCon))

e una tupla con l'esatto stato del servizio al tuple center servizi_camera:

servizi_camera?out(stato_servizi(servizio_comfort_temp(...)))

Classe di Agenti che implementa le caratteristiche del ruolo individuale **A4-Sensore Temperatura:**

Posizione occupata nel sistema - si trova all'interno del Gestore Camera, più precisamente nel gruppo G3 (Gestione comfort termico interno camera).

Ruolo individuale – Questa classe comprende due agenti, il primo spedisce il valore della temperatura corrente rilevata dal sensore della camera al tuple center temperatura_camera:

temperatura_camera?out(sensoretemp(TempAttuale))

il secondo spedisce ad intervalli regolari, la tupla che indica il funzionamento corretto del sensore associato:

temperatura_camera?out(device_ready(CodDispositivo))

CodDispositivo = 0, il codice corrispondente al sensore temperatura.

Classe di Agenti che implementa le caratteristiche del ruolo individuale **A5- Impostazioni pannellino:**

Posizione occupata nel sistema - si trova all'interno del Gestore Camera, più precisamente nel gruppo G3 (Gestione comfort termico interno camera).

Ruolo individuale – Questa classe comprende due agenti, il primo spedisce le impostazioni dell'utente di camera, riguardanti il servizio Gestione del comfort termico al tuple center temperatura_camera:

*temperatura_camera?out(pannellino(TempImpostata,
ModFunzCon))*

il secondo spedisce ad intervalli regolari, la tupla che indica il funzionamento corretto del pannellino utente:

temperatura_camera?out(device_ready(CodDispositivo))

CodDispositivo = 10, il codice corrispondente al pannellino utente.

Classe di Agenti che implementa le caratteristiche del ruolo individuale **A6- Condizionatore:**

Posizione occupata nel sistema - si trova all'interno del Gestore Camera, più precisamente nel gruppo G3 (Gestione comfort termico interno camera).

Ruolo individuale – Questa classe comprende due agenti, il primo rimane in attesa della tupla utile per comandare il funzionamento del condizionatore, sul tuple center temperatura_camera. Appena la nuova tupla è disponibile viene immediatamente prelevata assegnando i comandi al dispositivo:

```
temperatura_camera?in( fancoil( InvernoEstate, ModFunzCon ) )
```

il secondo spedisce ad intervalli regolari, la tupla che indica il funzionamento corretto del condizionatore:

```
temperatura_camera?out( device_ready(CodDispositivo))
```

CodDispositivo = 20, il codice corrispondente al condizionatore.

Classe di Agenti che implementa le caratteristiche del ruolo individuale **A7-Gestore Allarmi:**

Posizione occupata nel sistema - si trova all'interno del Gestore Camera, più precisamente nel gruppo G4 (Sicurezza interno camera).

Ruolo individuale – Questo agente sarà sempre attivo e leggerà costantemente, prima, le informazioni sui comandi, i parametri e le configurazioni inerenti al servizio Sicurezza, dal tuple center allarmi_camera attraverso i seguenti comandi:

```
allarmi_camera?rdp(servizio_sicurezza( info(...) ))
```

poi continuerà a leggere dallo stesso tuple center eventuali tuple che segnalano condizioni di allarme nella stanza:

```
allarmi_camera?rdp(allarme(servizio_sicurezza(CodDispositivo)))
```

```
allarmi_camera?rdp(allarme(servizio_comfort_temp(CodDispositivo)))
```

allarmi_camera?rdp(allarme(CodDispositivo))

Come accennato in precedenza il CodDispositivo identifica sia i dispositivi installati in camera, ma distingue anche le due tipologie di situazioni critiche(allarmi e warning).

Infine dopo aver letto gli eventuali allarmi, essi vengono trasferiti nuovamente, in base alle impostazioni stabilite dal supervisore, al tuple center centrale_allarmi:

```
centrale_allarmi@supervisore?out(
    allarmi_warning(
        CodCamera,
        CodDispositivo,
        servizio_comfort_temp(CodDispositivo),
        servizio_sicurezza(CodDispositivo)))
```

e al tuple center servizi_camera, per l'aggiornamento dello stato:

```
servizi_camera?out(stato_servizi( servizio_sicurezza(
    allarmi_warning([ CodDispositivo,StatoDispositivo],...)))
```

Classe di Agenti che implementa le caratteristiche del ruolo individuale **A8-Rilevatore fughe di gas:**

Posizione occupata nel sistema - si trova all'interno del Gestore Camera, più precisamente nel gruppo G4 (Sicurezza interno camera).

Ruolo individuale – Questa classe comprende due agenti, il primo ha il compito di rilevare fughe di gas nella stanza e spedire al tuple center allarmi_camera la tupla di allarme:

```
allarmi_camera?out(allarme(CodDispositivo))
```

CodDispositivo = 30, il codice corrispondente al rilevatore di fughe di gas.

solo se ci sarà una variazione rispetto a quella precedentemente inviata, cioè verrà inviata la tupla solo nei passaggi di stato: da stato di quiete a stato di allarme oppure da stato di allarme a stato di quiete.

Il secondo spedisce ad intervalli regolari, la tupla che indica il funzionamento corretto del rilevatore:

allarmi_camera?out(device_ready(CodDispositivo))

CodDispositivo = 30

Classe di Agenti che implementa le caratteristiche del ruolo individuale **A9-Rilevatore Fumo:**

Posizione occupata nel sistema - si trova all'interno del Gestore Camera, più precisamente nel gruppo G4 (Sicurezza interno camera).

Ruolo individuale – Questa classe comprende due agenti, il primo ha il compito di rilevare fuoriuscite di fumo nella stanza e spedire al tuple center allarmi_camera la tupla di allarme:

allarmi_camera?out(allarme(CodDispositivo))

CodDispositivo = 40, il codice corrispondente al rilevatore di fuoriuscite di gas.

solo se ci sarà una variazione rispetto a quella precedentemente inviata, cioè verrà inviata la tupla solo nei passaggi di stato: da stato di quiete a stato di allarme oppure da stato di allarme a stato di quiete.

Il secondo spedisce ad intervalli regolari, la tupla che indica il funzionamento corretto del rilevatore:

allarmi_camera?out(device_ready(CodDispositivo))

CodDispositivo = 40

Classe di Agenti che implementa le caratteristiche del ruolo individuale **A10-Rilevatore porte aperte:**

Posizione occupata nel sistema - si trova all'interno del Gestore Camera, più precisamente nel gruppo G4 (Sicurezza interno camera).

Ruolo individuale – Questa classe comprende due agenti, il primo ha il compito di rilevare la presenza di porte aperte nella stanza e spedire al tuple center allarmi_camera la tupla di warning:

allarmi_camera?out(allarme(CodDispositivo))

CodDispositivo = 100, il codice corrispondente al rilevatore di porte aperte.

solo se ci sarà una variazione rispetto a quella precedentemente inviata, cioè verrà inviata la tupla solo nei passaggi di stato: da stato di quiete a stato di warning oppure da stato di warning a stato di quiete.

Il secondo spedisce ad intervalli regolari, la tupla che indica il funzionamento corretto del rilevatore:

allarmi_camera?out(device_ready(CodDispositivo))

CodDispositivo = 100

Classe di Agenti che implementa le caratteristiche del ruolo individuale

A11-Rilevatore finestre aperte:

Posizione occupata nel sistema - si trova all'interno del Gestore Camera, più precisamente nel gruppo G4 (Sicurezza interno camera).

Ruolo individuale – Questa classe comprende due agenti, il primo ha il compito di rilevare la presenza di finestre aperte nella stanza e spedire al tuple center allarmi_camera la tupla di warning:

allarmi_camera?out(allarme(CodDispositivo))

CodDispositivo = 110, il codice corrispondente al rilevatore di finestre aperte.

solo se ci sarà una variazione rispetto a quella precedentemente inviata, cioè verrà inviata la tupla solo nei passaggi di stato: da stato di quiete a stato di warning oppure da stato di warning a stato di quiete.

Il secondo spedisce ad intervalli regolari, la tupla che indica il funzionamento corretto del rilevatore:

```
allarmi_camera?out(device_ready(CodDispositivo))
```

CodDispositivo = 110

Classe di Agenti che implementa le caratteristiche del ruolo individuale

A12-Aggiorna allarmi:

Posizione occupata nel sistema - si trova all'interno del Gestore Supervisore, più precisamente nel gruppo G2 (Sicurezza generale).

Ruolo individuale – Rimane in attesa, con un'operazione bloccante(in), di eventuali allarmi, provenienti dalle varie camere, sul tuple center centrale_allarmi:

```
centrale_allarmi?in(incoming_alarm(  
    DataOra,  
    allarmi_warning(  
        CodCamera,  
        CodDispositivo,  
        servizio_comfort_temp(CodDispositivo),  
        servizio_sicurezza(CodDispositivo) )))
```

La segnalazione degli allarmi di camera al supervisore risulta essere l'operazione con priorità massima del sistema.

Capitolo 5 - Conclusioni

Il punto cruciale su cui discutere le conclusioni, riguarda il fatto, che nel progetto è stata utilizzata un'ampia gamma di strumenti(Coordination Model, SODA, e TuCSon) atti alla definizione del sistema domotico D.A.S., in base alla coordinazione delle sue parti indipendenti(agenti), ma non sono stati utilizzati i corrispondenti modelli per la progettazione dei singoli agenti. La definizione degli agenti è stata fatta in base alle informazioni scambiate con i vari tuple center associati. Anche in fase di implementazione la loro realizzazione è avvenuta, adattando ed elaborando i costrutti base del linguaggio Java (object-oriented), ai concetti fondamentali dei sistemi multi-agente (agent-oriented). Le cose appena dette potrebbero essere in contraddizione, ma attualmente le tecnologie che sfruttano le metodologie agent-oriented sono in fase sperimentale e ancora necessitano miglioramenti in questo senso. Nonostante ciò, queste nuove metodologie catturano quei concetti di eterogeneità, modularità utili alla definizione di sistemi distribuiti complessi, di cui il sistema domotico realizzato in questa tesi, ne costituisce un ottimo esempio pratico.

Bibliografia

1. N. R. Jennings e S. Bussmann, "Agent-Based Control Systems: Why Are They Suited to Engineering Complex Systems?", IEEE Control Systems Magazine, Giugno 2003.
2. A. Omicini e F. Zambonelli, "Coordination for Internet Application Development", in Autonomous Agents and Multi-Agent Systems, 2, 251-269, 1999.
3. F. Zambonelli, N. R. Jennings, A. Omicini e M. J. Wooldridge, "Agent-Oriented Software Engineering for Internet Applications", in AOSE 2000, 13, 326 – 346.
4. A. Omicini, "SODA: Societies and Infrastructures in the Analysis and Design of Agent-Based Systems", in AOSE 2000, LNCS 1957, 185-193, 2001.
5. A. Omicini, E. Denti, "From tuple spaces to tuple centres", Science of Computer Programming, 41, 277-294, 2001.
6. The aliCE research team, "TuCSon Documentation", DEIS – University of Bologna/Cesena, Italy, Febbraio 5, 2002.
7. E. Denti, A. Ricci e R. Rubino, "Integrating and Orchestrating Services upon an Agent Coordination Infrastructure".
8. Lack, "Storia della domotica: Un percorso tortuoso durato un secolo", Domotrix, Ottobre 26, 2003.
9. "Confronto Home e Building Automation", www.sistamacasa.it.
10. "Tecnologie di base", www.domotica.it.
11. A. Rossi, "Domotica: tecnologie e applicazioni per la casa", PCMagazine.
12. I. Iacopini, "Il problema dell'interoperabilità", Tesi: Nuove tecnologie per l'accesso a edifici intelligenti, 2003.
13. "Sito di riferimento per l'infrastruttura di coordinazione TuCSon", www.lia.deis.unibo.it/Research/TuCSon.

