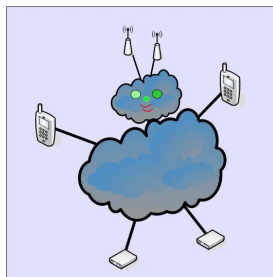


# **Sistema per al control de sensors mitjançant P2P**



**Joan Josep Garrido Sànchez**

**Nom del Consultor: Marc Domingo Prieto**

**Àrea: PFC–Xarxes de Computadors**

**Estudis: Enginyeria Informàtica**

**Universitat Oberta de Catalunya**

**7 de Gener del 2013**

A Almudena, la meva dona, pel seu suport i coratge en arribar allà on jo no he pogut durant aquest temps. A Marina, la nostra filla d'un anyet, per les estones que no li he pogut dedicar.

## Resum

L'aparició de les aplicacions peer-to-peer (P2P) ha obert el plantejament d'alguns paradigmes tradicionals en les arquitectures de computadors i les seves aplicacions. El desenvolupament d'aquest tipus d'arquitectures ha derivat en sistemes distribuïts més complexos i que obren la possibilitat a que altres dispositius puguin participar de recursos de programari o de maquinari allà on siguin.

JXTA (juxtapose) neix com l'opció de Sun Microsystems i Java a aquest tipus d'arquitectures. JXTA juntament amb JXME, la versió abreviada de JXTA per dispositius mòbils, possibiliten el desenvolupament d'aplicacions P2P. JXTA com a arquitectura de software permet integrar dispositius de tota mena, des dels clàssics ordinadors, portàtils, servidors, dispositius de mobilitat com PDA's o telèfons mòbils de poca capacitat a dispositius de mobilitat de darrera generació com a Smartphones dins d'una mateixa xarxa independentment de la topologia física d'aquesta.

El projecte de Monitorització i Control de Sensors aprofita les possibilitats que ofereix JXTA per crear un sistema distribuït de motes sensores que es poden trobar distribuïdes per arreu i que poden ser monitoritzades o controlades per una aplicació mòbil en un dispositiu smartphone o simplement en un dispositiu mòbil de baixa capacitat. El projecte desenvolupat inclou principalment els següents elements:

- Les motes sensores són simulades per una aplicació que permet que aquestes puguin oferir els seus serveis a la xarxa JXTA.
- La monitorització i control de les motes sensores s'ha desenvolupat aprofitant JXME. Es tracta d'una aplicació per a Android que es connecta a la xarxa JXTA a través dels recursos que JXME ofereix.

# Índex

Resum .....	3
Índex.....	4
Índex d'il·lustracions.....	7
Índex de taules .....	9
1. Introducció .....	11
1.1. Justificació i context del projecte.....	11
1.2. Objectius .....	12
1.3. Planificació .....	14
2. Requeriments del Sistema .....	17
2.1. Sensors .....	17
2.2. Monitorització de sensors .....	18
2.3. Control de sensors .....	19
3. Anàlisi.....	20
3.1. Sistema de sensors .....	20
3.1.1. Definició del sistema de sensors .....	20
3.1.2. Tipus de sensors .....	20
3.1.3. Estats dels sensors i les motes sensores.....	23
3.2. Definició dels missatges.....	25
3.3. Protocol d'intercanvi de dades .....	28
3.4. Anàlisi funcional del sistema .....	29
4. Disseny .....	32
4.1. Arquitectura del sistema .....	32
4.1.1. Definició del sistema .....	32
4.2. Arquitectura de software.....	36
4.2.1. Breu descripció de l'arquitectura de software JXTA .....	36
4.2.2. Descripció de l'arquitectura de software Android.....	37
4.2.3. Descripció de l'arquitectura de software del sistema desenvolupat...	39
4.3. Disseny del sistema.....	41
4.3.1. Diagrama de disseny estàtic .....	41
4.4. Disseny de l'entorn gràfic.....	50
4.4.1. Disseny entorn gràfic de la gestió de sensors .....	50

4.4.2. Disseny entorn gràfic Monitorització i Control de Motes Sensores.....	52
4.4.3. Entorn gràfic de monitorització.....	53
4.4.4. Entorn gràfic de control.....	53
4.4.5. Intercanvi de missatges entre Mota Sensora i Proxied Peer.....	55
4.4.6. Funcionament del procés d'intercanvi de missatges entre Proxied Peer i Mota Sensora .....	61
5. Implementació .....	62
5.1. Eines i tecnologies utilitzades .....	62
5.2. Consideracions durant el procés d'implementació.....	64
6. Manual d'ús del Sistema de Monitorització de Motes Sensores.....	72
6.1. Aplicació de Gestió de Mota Sensora.....	72
6.1.1. Inici de l'aplicació de Gestió de Motes Sensores.....	72
6.1.2. Creació d'una mota sensora.....	76
6.1.3. Posada en marxa de la mota sensora .....	81
6.1.4. Seguiment de l'execució d'una mota sensora .....	81
6.1.5. Aturada de la mota sensora .....	83
6.2. Aplicació de Monitorització i control de Mota Sensora.....	84
6.2.1. Inici de l'aplicació .....	84
6.2.2. Pantalla principal de l'aplicació .....	84
6.2.3. Connexió a la xarxa de l'aplicació .....	85
6.2.4. Cerca de les motes sensores disponibles .....	87
6.2.5. Monitorització i control d'una mota sensora .....	87
6.2.6. Monitorització i control d'un sensor.....	89
6.3. Funcionament trivial de l'aplicació .....	90
6.3.1. Configuració i execució del Relay Peer.....	91
7. Canvis i millores de futur del sistema .....	95
8. Conclusions .....	97
9. Glossari.....	99
10. Bibliografia .....	102
10.1. Referències a JXTA.....	102
10.2. Referències a JXME.....	103
10.3. Referències a Android .....	104

10.4. Referències a sensors .....	104
10.5. Referències eines utilitzades .....	104
10.6. Altres referències .....	105
11. Annexos .....	106
11.1. Annex I – Joc de proves realitzat.....	107
Prova 1: Creació de la mota sensora i posada en marxa .....	107
Prova 2: Visualització del Group i de la Mota Sensora en el Relay Peer .....	109
Prova 3: Connexió de l'aplicació Android Proxied Peer a la xarxa JXTA.....	110
Prova 4: Monitorització de les dades de la mota sensora creada .....	112
Prova 5: Control de la mota sensora creada (modificació de les dades).....	117
Prova 6: Monitorització d'un sensor de la mota sensora .....	125
Prova 7: Control d'un sensor de la mota sensora (modificació de les dades) .....	130

# Índex d'il·lustracions

Figura 1: Disseny breu del sistema proposat.....	12
Figura 2: Descripció d'alt nivell dels processos del sistema: definició i gestió de dades de sensors .....	29
Figura 3: Descripció d'alt nivell dels processos del sistema: gestió de serveis ..	30
Figura 4: Equivalència xarxa JXTA amb xarxa física .....	32
Figura 5: Comunicació del Proxied Peer amb el Relay Peer .....	33
Figura 6: Arquitectura del sistema de monitorització i control de sensors .....	34
Figura 7: Arquitectura de software del sistema .....	37
Figura 8: Arquitectura d'Android .....	38
Figura 9: Diagrama disseny estàtic aplicació Gestió Mota Sensora .....	44
Figura 10: Diagrama disseny estàtic aplicació Monitorització i Control de Mota Sensora .....	45
Figura 11: Diagrama de seqüència Procés de Connexió aplicació de Gestió de Mota Sensora.....	47
Figura 12: Diagrama de seqüència Procés de Connexió Proxied Peer a Relay Peer .....	48
Figura 13: Aplicació gestió Motes Sensores – Pestanya Bàsics.....	50
Figura 14: Aplicació gestió Motes Sensores – Pestanya Bàsics.....	51
Figura 15: Aplicació de gestió de Motes Sensores – Pestanya Starting Services	51
Figura 16: Aplicació de gestió de Motes Sensores – Pestanya Mota Sensora Execution Information.....	52
Figura 17: Monitorització i control de Motes Sensores – Interfície de connexió .	52
Figura 18: Vistes de l'entorn gràfic de monitorització .....	53
Figura 19: Vistes de l'entorn gràfic de control .....	54
Figura 20: Intercanvi de missatges Proxied Peer – Mota Sensora .....	61
Figura 21: Procés de connexió JXME Proxied i Relay Proxied Peer .....	65
Figura 22: Arquitectura trivial del sistema de Motes Sensores.....	66
Figura 23: Format de la Mota Sensora Advertisement .....	68
Figura 24: Format del Peer Advertisement personalitzat.....	69
Figura 25: Format del Peer Advertisement final.....	70
Figura 26: Format del Pipe Advertisement final .....	70

Figura 27: Execució aplicació Gestió de Motes Sensores sense GUI.....	72
Figura 28: Sortida execució aplicació Gestió de Motes Sensores .....	73
Figura 29: Execució aplicació Gestió de Motes Sensores amb GUI.....	74
Figura 30: Funcionament GUI – Pestanya Basics .....	74
Figura 31: Funcionament GUI – Pestanya Sensor Manager .....	75
Figura 32: Funcionament GUI – Pestanya Starting Services .....	75
Figura 33: Funcionament GUI – Pestanya Mota Sensora Execution Information	76
Figura 34: Creació d’una mota sensora – Pestanya Bàsics .....	77
Figura 35: Creació d’una mota sensora – Pestanya Sensor Manager .....	78
Figura 36: Posada en marxa d’una mota sensora – Pestanya Starting Services	81
Figura 37: Seguiment execució d’una mota sensora – Pestanya Mota Sensora Execution Information.....	82
Figura 38: Seguiment execució d’una mota sensora – Pestanya Mota Sensora Execution Information.....	83
Figura 39: Inici del servei d’una mota sensora – Pestanya Starting Services ....	83
Figura 40: Localització de l’aplicació de Monitorització i control al menú principal Android.....	84
Figura 41: Aplicació Monitorització – Pantalla de benvinguda .....	85
Figura 42: Aplicació Monitorització – Pantalla de configuració de la connexió....	86
Figura 43: Aplicació Monitorització – Seguiment Connexió .....	86
Figura 44: Aplicació Monitorització – Seguiment Connexió .....	87
Figura 45: Aplicació Monitorització – Monitorització de les dades de la mota sensora.....	88
Figura 46: Aplicació Monitorització – Control de les dades de la mota sensora..	88
Figura 47: Aplicació Monitorització – Monitorització de les dades del sensor.....	89
Figura 48: Aplicació Monitorització – Control de les dades del sensor .....	89
Figura 49: Arquitectura trivial de funcionament de l’aplicació .....	90
Figura 50: Configuració Relay Peer – Basic settings .....	91
Figura 51: Configuració Relay Peer – Advanced settings.....	92
Figura 52: Configuració Relay Peer – Rendezvous/Relays settings .....	93
Figura 53: Iniciar Relay Peer.....	93
Figura 54: Relay Peer - Shell de JXTA .....	94



# Índex de taules

Taula 1: Sensors geogràfics – Sensor de posició.....	21
Taula 2: Sensors ambientals – Sensor de temperatura.....	22
Taula 3: Sensors ambientals – Sensor de pressió .....	22
Taula 4: Sensors ambientals – Sensor de llum .....	22
Taula 5: Sensors ambientals – Sensor de soroll.....	22
Taula 6: Sensor de presència.....	23
Taula 7: Estats de les motes sensores.....	23
Taula 8: Estats dels sensors .....	24
Taula 9: Incorporació al Peer Group.....	25
Taula 10: Missatge d’anunci d’incorporació al Peer Group .....	25
Taula 11: Missatges de monitorització de sensors.....	26
Taula 12: Missatges de control de sensors.....	27
Taula 13: Contingut del missatge de petició.....	56
Taula 14: Exemple de missatge de petició de consulta mota sensora .....	56
Taula 15: Exemple de missatge de petició de consulta de sensor.....	56
Taula 16: Exemple de missatge de petició de modificació de dades de mota sensora .....	57
Taula 17: Exemple de missatge de petició de modificació de dades de sensor ..	57
Taula 18: Contingut del missatge de resposta .....	58
Taula 19: Exemple de missatge de resposta del tipus Mota Sensora .....	58
Taula 20: Exemple de missatge de resposta del tipus sensor .....	58
Taula 21: Exemple de missatge de resposta a la petició de totes les dades d’un sensor .....	60
Taula 22: Exemple de missatge de resposta a la petició de totes les dades d’un sensor .....	60
Taula 23: Interfícies de definició de les dades d’un sensor .....	80



# 1. Introducció

## ***1.1. Justificació i context del projecte***

El present Projecte de Final de Carrera es troba en el context de l'Àrea de Coneixement de Xarxes de Computadors i en la temàtica de sistemes peer-to-peer (en endavant P2P) basats en JXTA.

Es tracta d'un sistema que possibilita la monitorització i control de sensors distribuïts geogràficament. Es dota al sistema d'una aplicació de mobilitat que permet realitzar les operacions necessàries per a la seva monitorització i control a través de dispositius mòbils de baixa capacitat. Davant les característiques del sistema que es proposa al Projecte es planteja la creació d'un sistema basat en P2P que utilitza com a plataforma d'implementació JXTA.

El sistema està format per un conjunt de sensors que es basen en el concepte de peer. Aquests sensors o motes sensores tenen la capacitat de unir-se i interaccionar amb els serveis que ofereix la xarxa JXTA com si d'un peer més es tractés. Les motes sensores en el seu conjunt formen un sistema P2P que permeten la medició de diferents magnituds, depenen de la configuració de la mota sensora. Al mateix temps, aprofitant les possibilitats que ofereix la plataforma JXTA a través de la seva versió abreujada de mobilitat JXME el sistema permet la possibilitat de que pugui ser gestionat de forma remota.

## 1.2. Objectius

L'objectiu del projecte és la creació d'una aplicació Android que permeti monitoritzar i controlar l'estat de sensors. El projecte ha de definir l'arquitectura del sistema complet tot i que es desenvolupa únicament l'aplicació d'Android i la comunicació peer-to-peer. L'ús de sensors és simulat i queda fora de l'abast d'aquest projecte.

El sistema es basa en una xarxa P2P, la qual utilitza protocols JXTA per permetre la comunicació entre els diferents peers de la xarxa que comparteixen recursos. Dintre de la xarxa JXTA existeixen una sèrie de peers especials (Relay Peer) que han de possibilitar la participació en la xarxa a altres dispositius que no tenen capacitat per comunicar-se amb altres peers de la xarxa JXTA. Aquests dispositius, de recursos molt limitats (o que no volen implementar els protocols JXTA) anomenats Proxied Peer, intercanvien missatges amb el Relay Peer que actua com a super-peer. D'aquesta forma es proveeix l'accés dels Proxied Peers amb la xarxa JXTA.

El següent gràfic descriu breument els objectius fixats al projecte:

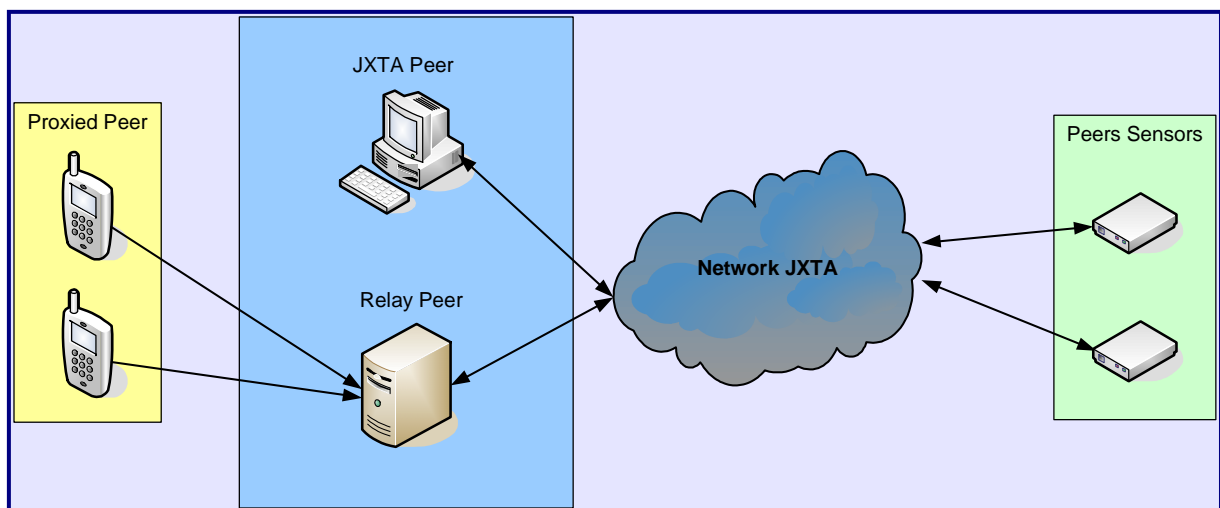


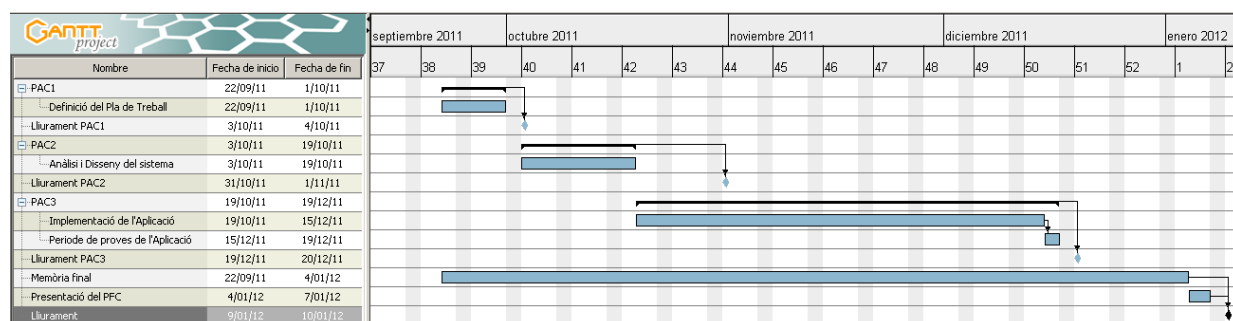
Figura 1: Disseny breu del sistema proposat

Els sensors en el projecte s'organitzen per conjunts de sensors o motes sensores. Aquestes motes sensores seran un peer i ofereixen una sèrie de serveis. El peer té capacitat de comunicació amb la xarxa JXTA. L'aplicació d'Android que es comporta com a Proxied Peer, mitjançant una interfície gràfica, ha de permetre la monitorització i control de l'estat dels serveis que ofereix la mota sensora i s'ha de comunicar amb la xarxa JXTA a través d'un Relay Peer.

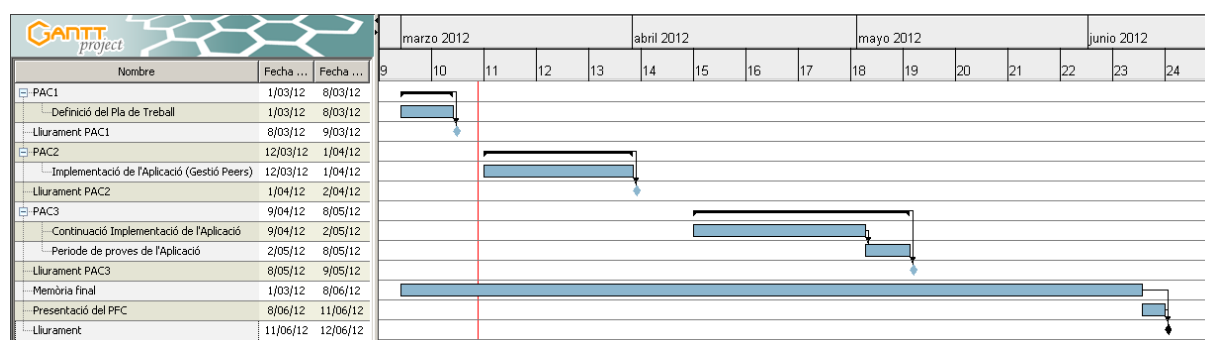
### 1.3. Planificació

Aquest projecte s'ha anat realitzant durant diferents semestres. Aquest fet ha condicionat que el projecte sigui planificat trimestre a trimestre d'acord a l'estat en què el projecte havia quedat en el període anterior. Així l'evolució de la planificació es pot observar a continuació:

En el següent cronograma es pot veure com es van distribuir les tasques al calendari en el moment d'inici del projecte. La planificació es va seguir fins a al lliurament de la PAC 2.



Durant el segon semestre la planificació que es va plantejar és la que es pot veure seguidament. El punt de partida era la fase d'implementació i tenia com a objectiu la finalització del projecte.



Al calendari presentat que s'ha plantejat durant aquest semestre s'han identificat algunes dates que corresponen a l'assoliment d'algunes fites en la realització del projecte. Aquestes són les que es descriuen tot seguit:

1 d'Octubre – PAC 1: Definició del pla de treball del projecte on s'especifiquen els objectius, el llistat de funcionalitat, l'estat actual del projecte, el calendari, la tecnologia utilitzada i els possibles riscos que detecten en aquest moment.

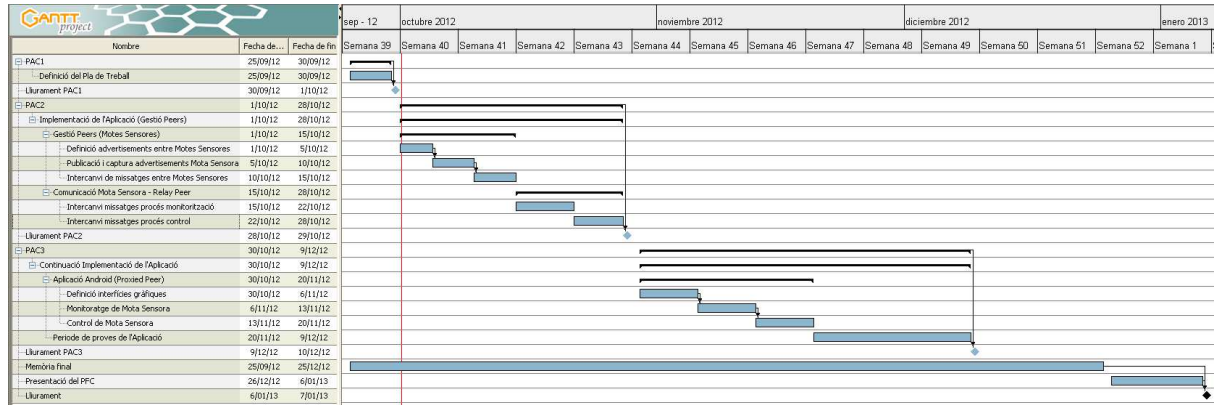
29 d'Octubre – PAC 2: En el lliurament de la PAC 2, l'etapa d'implementació ha d'haver completat la gestió de peers o motes sensora i la gestió de la comunicació entre ells. Les interfícies de comunicació entre la mota sensora i el Relay Peer han d'estar pràcticament finalitzades a fi de permetre el funcionament dels processos de monitorització i control.

10 de Desembre – PAC 3: L'etapa d'implementació ha d'estar completada. Les proves del projecte s'han d'haver executat a fi de comprovar que es compleixen els requeriments de funcionalitat esperats.

D'altra banda, s'hauran començat les activitats relacionades amb la entrega de la memòria, ja avançada ja que la documentació s'ha anat realitzant durant les diferents etapes del projecte, i la preparació de l'entrega del producte final i la seva presentació.

7 de Gener – Lliurament final: El lliurament final correspon a la finalització del projecte. En aquest moment es lliurarà el projecte complet amb el producte final assolit, la memòria corresponent al projecte i la presentació del PFC.

El nou cronograma pretén planificar la part restant del projecte, tenint com a punt de partida l'etapa en què es va deixar anteriorment. El cronograma de l'actual pla de treball comença per tant a l'etapa d'implementació en estat avançat i queda com segueix:



S'ha considerat un període de proves ampli amb la finalitat d'ajustar possibles desviacions de temps acumulades durant el que resta de l'etapa d'implementació.



## 2. Requeriments del Sistema

El sistema definit requereix les funcionalitats que es descriuen a continuació:

- Ha de permetre el monitoratge i control remot de les motes sensores distribuïdes a la xarxa des d'altres peers.
- El funcionament del sistema ha de poder ser un conjunt on els peers de sensors dispersos per la xarxa treballin com si fossin un mateix sistema i no simplement un conjunt de peers de sensors dispersos.
- Finalment, el conjunt funcionarà com a un grup restringit de peers i ha de garantir uns nivells de seguretat que permetin el funcionament correcte del sistema.

Les funcionalitats que requereixen els diferents components del sistema es defineixen en el següent apartat.

### 2.1. Sensors

Els sensors o motes sensores permeten obtenir diferents dades de l'exterior d'acord a magnituds o altres propòsits.

Els peers que implementen els sensors o motes sensores han de tenir les següents funcionalitats a fi de mantenir la seva autonomia i la del conjunt del sistema a implementar:

1- Han de permetre la gestió de les dades dels diferents sensors que formen la mota sensora i del seu estat de connexió i servei, alhora que possibiliten la monitorització o control remot quan aquests són requerits per altres peers del sistema.

2- S'han de poder gestionar els sensors de forma òptima a fi de millorar el rendiment dels seus propis recursos. Han de possibilitar l'emmagatzematge de les dades durant un cert període de temps fins que es produeixi la propera mesura, la qual es realitzarà amb la freqüència necessària a fi de complir amb un

marge de precisió determinat pel sensor. D'aquesta forma s'aconsegueix un estalvi d'energia que permet optimitzar el funcionament dels sensors o de les motes sensores.

3- Les motes sensores s'han de comportar com un peer més dintre d'una xarxa P2P i han de possibilitar totes les funcionalitats associades a un sistema d'aquestes característiques. Aquesta consideració es fa per simplicitat ja que les motes sensores també podrien ser Proxied Peers, els quals delegarien la seva integració a la xarxa P2P al seu super peer (Relay Peer).

Tot i que ja s'ha comentat amb anterioritat, cal considerar que els sensors o motes sensores seran implementats o considerats com a un peer habitual en una xarxa P2P.

## **2.2. Monitorització de sensors**

La monitorització de sensors es realitza des d'una aplicació realitzada en Android. A través d'un dispositiu mòbil es podrà comprovar l'estat de diferents sensors o motes sensores distribuïdes per la xarxa P2P. A fi d'ampliar el ventall de dispositius que poden utilitzar aquesta aplicació, s'ha considerat que la monitorització pugui funcionar sobre dispositius que no tenen capacitat per implementar els requeriments de connexió a la xarxa P2P com un peer més o simplement perquè no els volen implementar.

L'aplicació permetrà tenir la monitorització de qualsevol peer sensor o mota sensora i així tenir coneixement del seu estat, de l'estat d'alguns dels sensors o de qualsevol de les dades mesurades pel sensor. A fi de facilitar una capacitat de monitorització del conjunt del sistema més amplia, la monitorització ha de possibilitar consultes referents al conjunt del sistema.

### **2.3. Control de sensors**

El control de sensors permet realitzar operacions sobre qualsevol peer sensor o mota sensora modificant el seu estat si és necessari, posant-lo fora de servei o, simplement, canviant la parametrització d'alguns dels sensors si això es requereix. D'aquesta forma s'aconsegueix tenir el control remot dels diferents sensors o motes sensores.

Per al control de sensors s'ha considerat el mateix tipus de dispositiu que té la possibilitat de realitzar la monitorització.

## **3. Anàlisi**

### **3.1. Sistema de sensors**

#### **3.1.1. Definició del sistema de sensors**

Definirem el concepte de sensor com aquell dispositiu creat amb el propòsit de rebre informació d'una magnitud exterior que es pot transformar en una altra magnitud, habitualment elèctrica, la qual es pot mesurar o quantificar.

D'acord amb la definició prèvia podem dir que una mota sensora és un conjunt de sensors. El sistema dissenyat en el present projecte pretén la simulació de motes sensores, les quals es considera que tenen el funcionament i les capacitats de comunicació d'un peer en una xarxa JXTA. El sistema considera cada peer com una mota de sensors, el què significa que cada peer pot contenir diversos sensors i, per tant, pot informar de les diferents magnituds depenent de les característiques dels sensors que el compona.

En la realització d'aquest projecte s'ha considerat la classificació de les motes sensores en dos grans grups motes sensores dinàmiques i motes sensores estàtiques:

*Les motes sensores dinàmiques* tindran entre tots els seus sensors un sensor de posició que indicarà on es troba localitzada la mota sensora.

*Les motes sensores estàtiques* són aquelles en què no necessitem conèixer la seva posició i que, per tant, no tindran entre els seus sensors el sensor de posició.

#### **3.1.2. Tipus de sensors**

El correcte funcionament del sistema exigeix que es defineixin quins tipus de sensors podem trobar distribuïts als peers al llarg de la xarxa JXTA a fi d'assegurar un funcionament del sistema apropiat.

Els sensors que mesuren magnituds tenen associada una precisió que defineix com de fiables són les dades que el sensor està oferint quan se li consulta. El motiu d'aquesta precisió és que els sensors no hagin de capturar les dades en aquell instant i puguin donar les dades de la darrera consulta que hagin fet prèviament com a vàlides. D'aquesta forma s'aconsegueix que els sensors optimitzin el consum d'energia.

D'altra banda, podem trobar motes sensores que per la seva situació o característiques el consum d'energia no és un factor crític i que, per tant, poden realitzar la lectura dels diferents sensors de la mota sensora "per demanda".

Els diferents tipus de sensors que hi ha al sistema es defineixen a continuació.

## SENSORS GEOGRÀFICS

Sensors de posició: Els sensors de posició són elementals pel sistema. El fet de tenir una xarxa JXTA que distribueix els peers arreu vol dir que aquests possiblement es trobin distribuïts de forma geogràfica. Les magnituds recollides per a aquest tipus de sensor per al posicionament són latitud i longitud. Aquestes magnituds tenen associades un nivell de precisió, que és l'error màxim de les dades obtingudes pel sensor.

Paràmetre	Tipus	Unitat
Latitud	Decimal	Coordinate (x <sup>0</sup> y' z'' N)
Longitud	Decimal	Coordinate (x <sup>0</sup> y' z'' E)
Precisió	Decimal	Metres (m)

Taula 1: Sensors geogràfics – Sensor de posició

## SENSORS AMBIENTALS

Sensors de temperatura: Els sensors de temperatura ens permetran conèixer la temperatura a la posició remota on es troba el peer i determinar la precisió del sensor.

Paràmetre	Tipus
Temperatura	Decimal
Unitat de mesura	String
Precisió	Decimal

Taula 2: Sensors ambientals – Sensor de temperatura

Sensors de pressió: Els sensors de pressió ambiental permeten conèixer la pressió ambiental a la posició d'un peer. Els sensors de pressió tenen associat un nivell de precisió.

Paràmetre	Tipus
Pressió atmosfèrica	Decimal
Unitat de mesura	String
Precisió	Decimal

Taula 3: Sensors ambientals – Sensor de pressió

Sensors de llum: Permet conèixer la lluminositat que existeix al lloc on es troba el peer i la precisió de les dades.

Paràmetre	Tipus
Lluminositat	Decimal
Unitat de mesura	String
Precisió	Decimal

Taula 4: Sensors ambientals – Sensor de llum

Sensor acústic: Permet conèixer el nivell de soroll al lloc on es troba el peer i la precisió de les dades.

Paràmetre	Tipus
Soroll	Decimal
Unitat de mesura	String
Precisió	Decimal

Taula 5: Sensors ambientals – Sensor de soroll

## SENSORS DE PRESENCIA

Els sensors de presència permeten detectar si hi ha algú en el lloc on es troba el peer. En aquest sensor no té sentit el concepte de precisió.

Paràmetre	Tipus
Presència	Booleà

Taula 6: Sensor de presència

El fet de definir al sistema aquests tipus de sensors no vol dir que el sistema no pugui arribar a implementar altres tipus de sensors. Es tracta, per tant, d'especificar el rang de magnituds possibles que el sistema pot mesurar a través dels seus sensors, que en definitiva correspon a la informació que es pot adquirir dels diferents peers distribuïts per la xarxa JXTA.

### 3.1.3. Estats dels sensors i les motes sensores

S'ha considerat que les motes sensores actuen com a peers dintre de la xarxa. Aquestes poden passar per diferents estats durant el seu funcionament habitual. Els estats pels que poden passar les motes sensores són les següents:

Estat	Descripció
Connected	Indica que una determinada mota sensora es troba en connexió a la xarxa P2P
Disconnected	Indica que una determinada mota sensora es troba desconnectada de la xarxa P2P

Taula 7: Estats de les motes sensores

Així mateix, els sensors poden passar pels estats que es defineixen al següent quadre:

Estat	Descripció
Open	Indica que el sensor està en servei
Closed	Indica que el sensor es troba fora de servei
Standby	Indica que el sensor es troba en servei però no està capturant dades en aquell moment

Checking	Indica que el sensor es troba en el procés de mesura, capturant dades de l'exterior
Modified	Indica que una determinada operació de control sobre el sensor ha estat finalitzada

Taula 8: Estats dels sensors



### 3.2. Definició dels missatges

El propòsit dels missatges és mantenir la informació dels diferents peers distribuïda a través de la xarxa JXTA. Aquest funcionament permet que la xarxa JXTA sigui més àgil en les cerques i la distribució de serveis entre els diferents peers.

Amb aquest propòsit els missatges necessaris pel sistema s'han dividit segons la naturalesa funcional d'aquests. Els missatges identificats pel funcionament del sistema es defineixen breument a continuació.

#### Missatge d'incorporació al Peer Group

La connexió dels diferents peers de la xarxa JXTA es realitzarà utilitzant els protocols definits per a la xarxa en aquest propòsit. Així mateix, s'afegiran al Peer Group que correspongui utilitzant els protocols definits per JXTA per aquest efecte. Els missatges utilitzats per a l'establiment de connexió són els que es detallen tot seguit:

Missatge	Descripció
joinGroup	Aquest missatge associa un peer a un peer group.

Taula 9: Incorporació al Peer Group

#### ANUNCI D'INCORPORACIÓ AL PEER GROUP

Un cop connectat al Peer Group, el peer informarà a la xarxa JXTA quins són els sensors que té instal·lats a fi de facilitar a la xarxa la informació de quines són les dades que pot subministrar. El missatge que es facilita a la xarxa o advertisement seria el següent:

Missatge	Descripció
sendAdvertisement	Aquest missatge enviarà la informació dels sensors que conté un peer i el tipus, i anirà dintre de la informació que es transfereix entre el Peer Group.

Taula 10: Missatge d'anunci d'incorporació al Peer Group

## MISSATGES DE MONITORITZACIÓ

Els missatges de monitorització permeten interrogar un determinat peer sobre dades referents als seus sensors. Una vegada la mota sensora s'uneix al grup, publica el seu Peer Advertisement<sup>1</sup> on especifica que ofereix servei de monitorització, a quina pipe escolta i els sensors que té disponibles. Els missatges que s'han definit a nivell de monitorització són els següents:

Missatge	Descripció
getConnectionStatus	Permet saber si una determinada mota sensora es troba connectada o no a la xarxa JXTA.
isStatic	Permet saber si la mota sensora és estàtica o dinàmica.
getSensorStatus	Permet saber si un determinat sensor a un peer es troba obert o tancat.
getDataMotaSensora	Permet obtenir l'estat complet de la mota sensora.
getDataSensorStatus	Permet rebre totes les dades de l'estat en que es troba un sensor.
getDataSensorStandBy	Permet conèixer si un sensor es troba en Stand By.
getSensorType	Permet conèixer quin és el tipus del sensor.
getSensorPosition	Permet conèixer quina és la posició del sensor.
getAllDataMotaSensora	Permet rebre totes les dades d'una mota sensora amb la llista dels sensors que conté

Taula 11: Missatges de monitorització de sensors

---

<sup>1</sup> Advertisement – Document XML intercanviat pels diferents Peers d'una xarxa JXTA i que descriu els diferents recursos que es poden trobar a la xarxa (nodes, grups, canals, serveis,...).

## MISSATGES DE CONTROL DE SENSORS

A nivell de control dels sensors s'han definit els missatges necessaris a fi de prendre el control d'un sensor ubicat a un determinat peer. La següent taula mostra quins són els missatges definits:

Missatge	Descripció
setConnectionStatus	Permet desconnectar o connectar la mota sensora remota.
setSensorStatus	Permet obrir o tancar un determinat sensor a un peer.
setSensorStandBy	Permet posar un determinat sensor en Stand By.
moveSensorPosition	Permet modificar la posició del sensor.
setDataSensorValueAtPosition	Permet modificar el valor d'un determinat sensor a una posició.
checkDataSensor	Permet indicar a un determinat sensor que faci la lectura en aquell instant al sensor.
getAllDataSensora	Permet obtenir totes les dades d'un sensor

Taula 12: Missatges de control de sensors

### **3.3. Protocol d'intercanvi de dades**

El funcionament del sistema de monitorització i control de peers es fonamenta en una xarxa de tipus P2P. El correcte funcionament d'aquest tipus de sistema té com a punt clau l'intercanvi de dades entre els diferents peers i la superació dels possibles reptes de connexió entre els diferents components de la xarxa.

Un dispositiu mòbil de baixa capacitat ha de poder rebre la informació necessària de l'estat d'un determinat sensor o mota sensora que li permeti la monitorització, poder prendre el control si ho estima necessari i modificar el seu estat.

A l'apartat d'"Arquitectura del sistema" d'aquest projecte es defineix de quina forma es comuniquen i intercanvien les dades els diferents components de la xarxa P2P a fi de poder oferir el servei de monitorització i control dels diferents sensors o motes sensores de la xarxa a través d'un dispositiu mòbil.

### 3.4. Anàlisi funcional del sistema

Els principals processos que defineixen el sistema a alt nivell són els que es presenten seguidament per mitjà dels diagrames de casos d'ús següents. A nivell de gestió de dades de sensors podem trobar el següent:

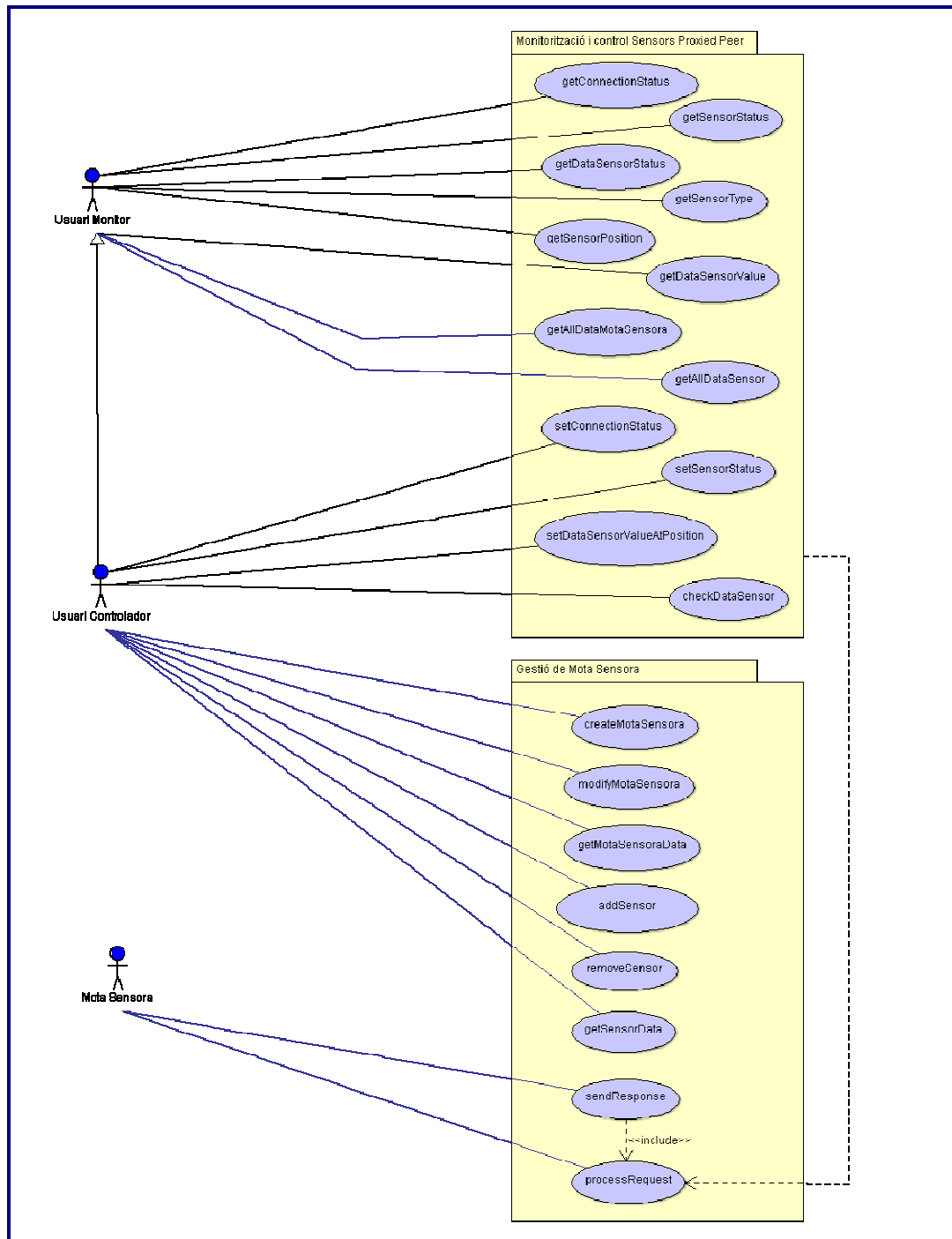


Figura 2: Descripció d'alt nivell dels processos del sistema: definició i gestió de dades de sensors

Al diagrama de casos d'ús a nivell de gestió de dades es poden observar els diferents actors que interactuen amb el sistema. D'una banda tindrem l'usuari monitor que permetrà únicament veure les dades, d'una altra banda tindriem un usuari controlador que podria fer tasques de modificació de dades. Existeix l'usuari mota sensora que realitzaria tota la part de gestió del servei de la mota sensora.

El diagrama ens mostra com es fa la gestió de la mota sensora a nivell de dades. La consulta i modificació es realitza des del Proxied Peer (que en el nostre cas serà una aplicació d'Android). La definició de la mota sensora i dels seus sensors es fa a la mota sensora a través de la seva interfície gràfica.

Pel que fa als serveis de la mota sensora el diagrama de casos d'ús queda com el següent:

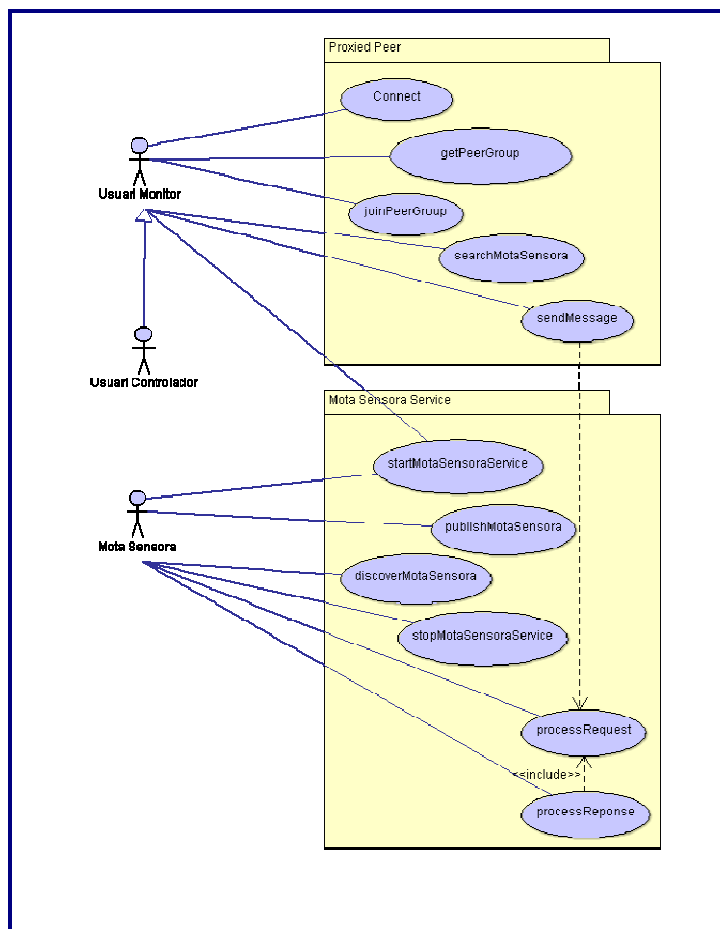


Figura 3: Descripció d'alt nivell dels processos del sistema: gestió de serveis

La gestió de serveis del sistema es divideix en dos grans blocs: per un costat els serveis referents a la mota sensora i d'un altre costat els serveis que arrenca el Proxied Peer per al seu funcionament dintre de la xarxa JXTA a través del seu Relay Peer.

En aquest diagrama de casos d'ús podem observar com els diferents actors interaccionen amb el sistema. Els actors usuari monitor i controlador únicament poden interaccionar directament amb els serveis de la mota sensora per arrencar-los o aturar-los. Un cop una mota sensora està arrencada aquesta es gestiona a través del Proxied Peer.

A nivell de serveis, el Proxied Peer únicament gestiona la seva connexió a la xarxa JXTA a través del seu Relay Peer. Per tant aquest té els mètodes que li són necessaris per a la connexió, descobriment de recursos de la xarxa i enviament i rebuda de missatges.

## 4. Disseny

### 4.1. Arquitectura del sistema

#### 4.1.1. Definició del sistema

El sistema de control i monitorització de sensors s'ha implementat basant el seu funcionament en una xarxa P2P. La xarxa P2P, com el seu nom indica, es basa en parells d'iguals que es comuniquen en xarxa, el que permet una millor distribució dels serveis i una capacitat superior de tolerància a errors entre d'altres.

La plataforma escollida per al desenvolupament de la xarxa P2P ha estat JXTA. JXTA està basat en un conjunt de protocols que permeten la publicació de recursos en el medi així com la seva cerca i consum, independentment de la capa de transport i del llenguatge d'implementació utilitzat. Permet la comunicació d'un ampli ventall de dispositius: ordinadors, servidors, mòbils, smartphones, PDA's,... D'aquesta forma el sistema manté una comunicació descentralitzada.

La següent figura vol mostrar un exemple d'equivalència entre la xarxa JXTA i el seu equivalent físic.

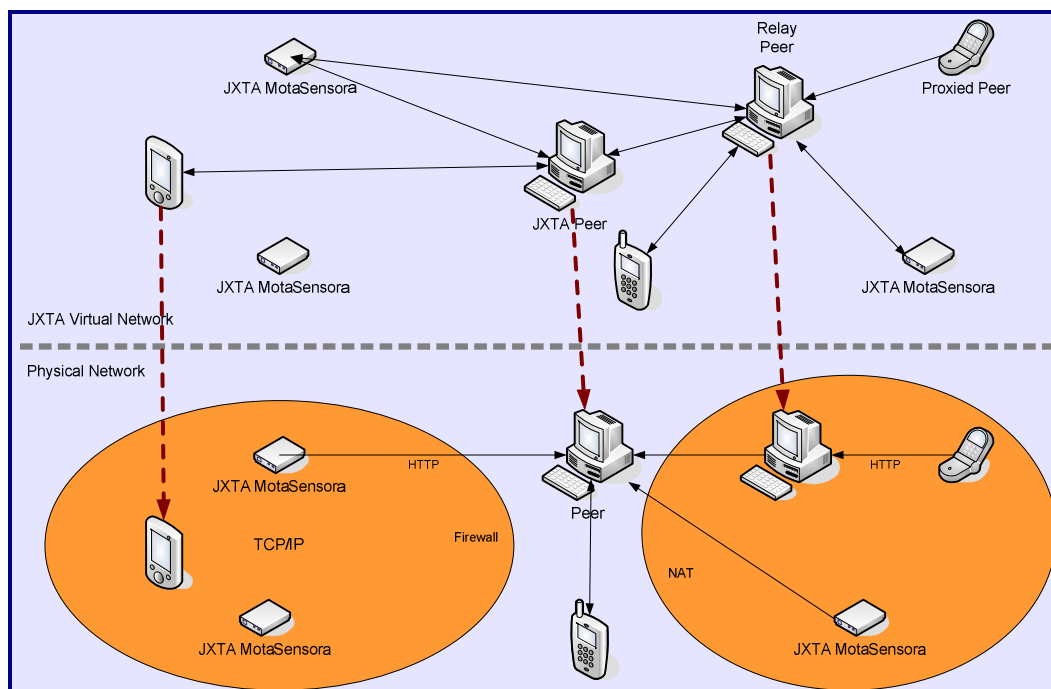


Figura 4: Equivalència xarxa JXTA amb xarxa física



A la figura es pot observar com la comunicació entre els diferents peers de la xarxa JXTA es realitza independentment de la infraestructura de xarxa existent i és transparent a dispositius físics com firewalls o possibles traduccions d'adreçaments de xarxes utilitzant NAT.

Un dels objectius del projecte és simular el funcionament de sensors que implementarien protocols JXTA i que en el conjunt d'una mota sensora serien per si mateixos peers de la xarxa P2P. Per tant, compartrien recursos amb altres peers que podrien ser motes sensores o, com anteriorment s'ha especificat, altre tipus de dispositius.

La monitorització i control de les motes sensores es realitza per mitjà d'una aplicació que accedeix a la xarxa JXTA amb el rol de Proxied Peer. El fet de que hagi de funcionar amb el rol de Proxied Peer respon a que el dispositiu mòbil pot no implementar els protocols JXTA per falta de recursos o bé perquè no els vol implementar. Aquest fet no ha de ser un inconvenient per utilitzar l'aplicació de monitorització i control.

El Proxied Peer com no té capacitat de connectar per si mateix a la xarxa JXTA i compartir recursos amb la resta de peers, ho haurà de fer a través d'un Relay Peer que és qui farà aquesta tasca. La comunicació entre Proxied Peer i Relay Peer es fa a través del protocol HTTP que és comú als dos dispositius. La següent figura mostra com es fa la comunicació entre aquests dos elements.

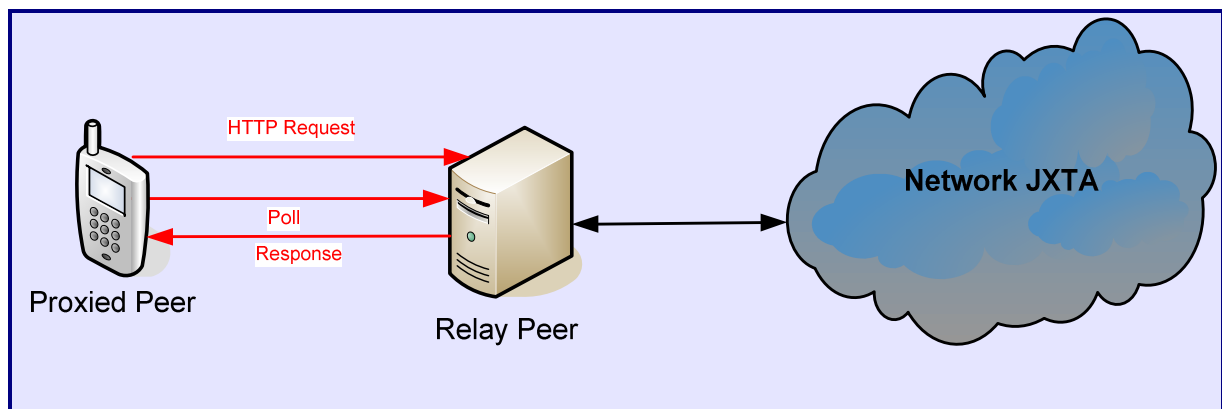


Figura 5: Comunicació del Proxied Peer amb el Relay Peer

Els dispositius de tipus Proxied Peer utilitzen una versió de JXTA reduïda anomenada JXME<sup>2</sup>. Dintre de JXME podem trobar la versió més simple (JXME-Proxied) que implementa la comunicació entre el Proxied Peer i el Relay Peer. Aquesta comunicació, com hem vist a la figura, es basa en HTTP. Les responsabilitats del Relay Peer són:

- Atendre les peticions del Proxied Peer.
- Traduir missatges que venen del Proxied Peer al format XML utilitzat a la xarxa JXTA.
- Guardar els missatges de la xarxa JXTA dirigits al Proxied Peer fins que aquest els requereixi per consumir-los.
- Convertir els missatges de la xarxa JXTA en un format simple que pugui entendre el Proxied Peer.

Per tant, l'arquitectura del sistema que estem definint queda com la que es descriu a la següent figura:

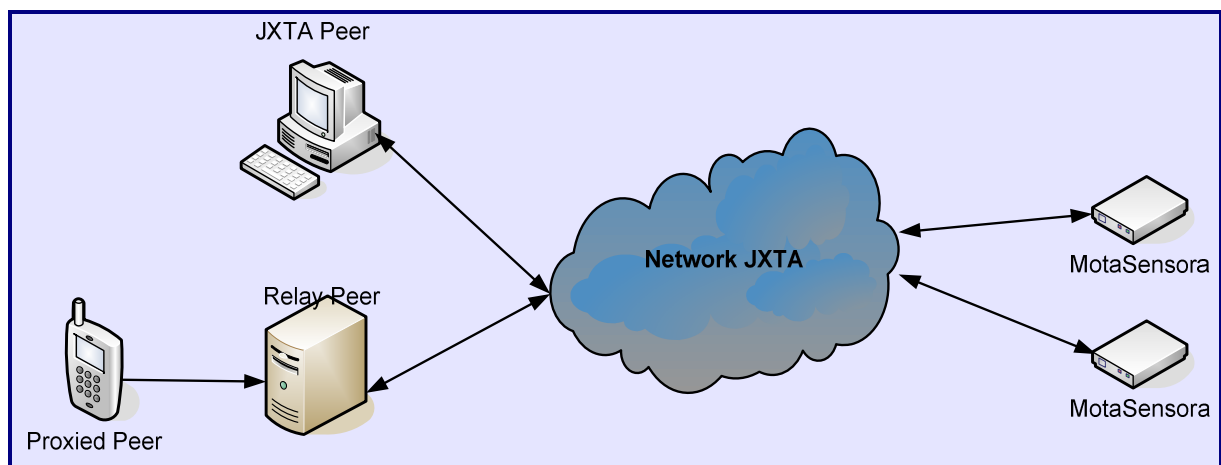


Figura 6: Arquitectura del sistema de monitorització i control de sensors

---

<sup>2</sup> JXME – Sigles que corresponen a JXTA Micro Edition

Les motes de sensors es comporten com a peers, els quals poden unir-se al Peer Group que correspongui, anunciar els seus serveis i proveir les dades de les magnituds mesurades per les motes de sensors. Aquestes dades es poden facilitar a aquells peers del grup que requereixin les dades.

De forma anàloga el funcionament del control dels diferents sensors serà el mateix, els sensors són controlats de forma remota per altres peers.

## **4.2. Arquitectura de software**

### **4.2.1. Breu descripció de l'arquitectura de software JXTA**

L'arquitectura de software de la plataforma JXTA es divideix en capes amb diferents propòsits. D'una banda la capa d'aplicacions conté aquelles aplicacions que funcionen sobre els peers de la xarxa JXTA i que amplien la funcionalitat dels serveis a aplicacions existents a la xarxa JXTA.

En segon lloc, al nivell de serveis hi haurà aquells serveis que un peer pot oferir per si mateix sense necessitat d'aplicacions addicionals. Aquests serveis donen suport a les aplicacions residents al nivell superior d'aplicacions i faciliten l'intercanvi entre els diferents peers.

El següent nivell és el nucli o *core* i dóna el suport fonamental a serveis P2P i a les aplicacions. Aquest nivell permet la formació de grups de peers, la comunicació entre peers, el seu monitoratge i la seguretat necessària pel seu funcionament.

Finalment trobaríem els dispositius, els quals tindran accés a la xarxa JXTA independentment del tipus de dispositiu.

La següent figura té com a objectiu aclarir el que s'especifica en aquest apartat. En ella es pot veure amb claredat quina és l'arquitectura que la plataforma JXTA proveeix.

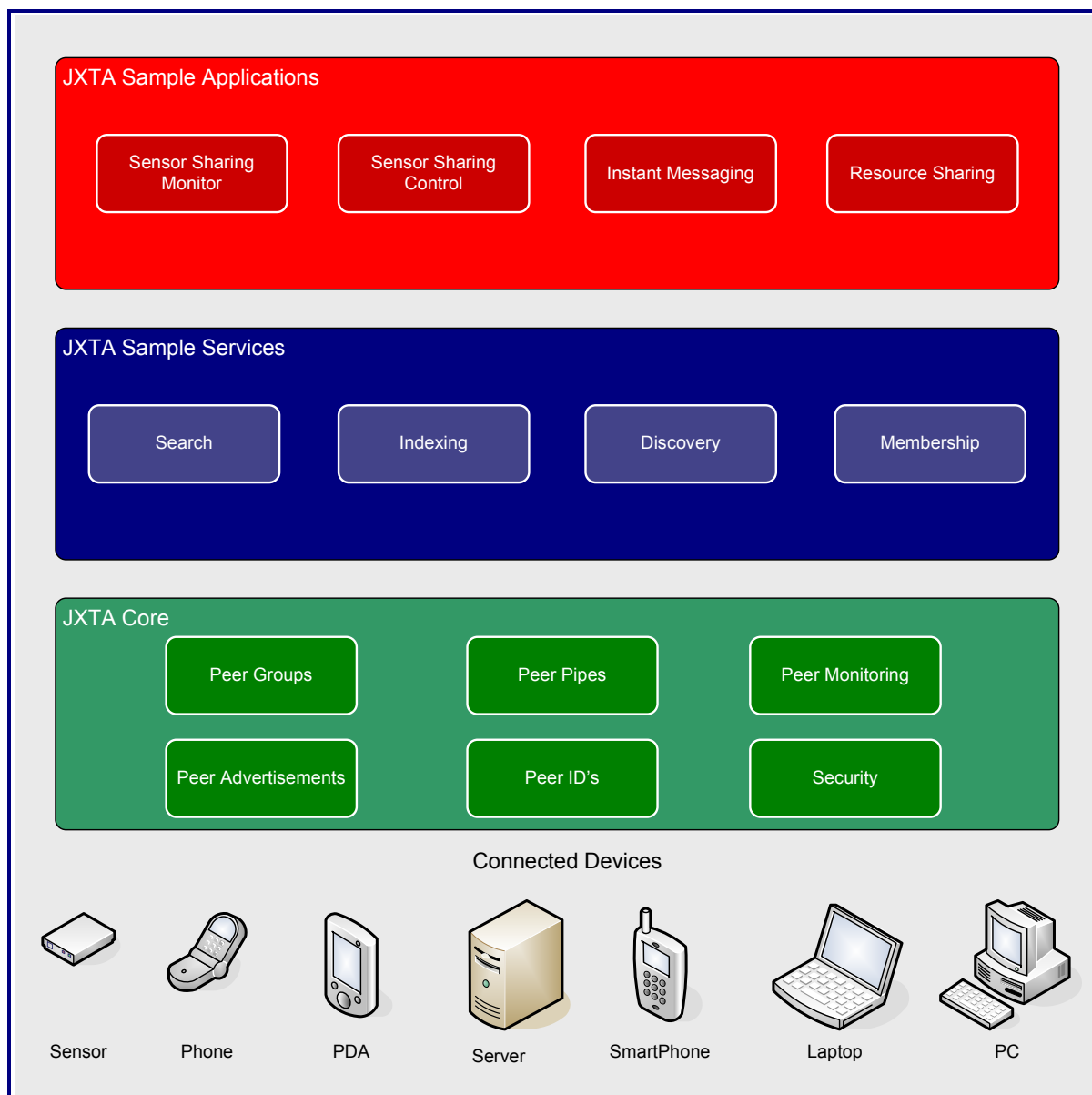


Figura 7: Arquitectura de software del sistema

#### 4.2.2. Descripció de l'arquitectura de software Android

Android és una plataforma de dispositius mòbils basada en un sistema operatiu Linux i desenvolupada per Google i Open Handset Alliance. Compta amb un conjunt de serveis de sistema que permeten la gestió de la seguretat, de la memòria, dels processos, de la xarxa i dels tipus de drivers.

La següent figura especifica de forma gràfica quina és l'arquitectura d'Android, de quina forma es distribueix el disseny de la plataforma:

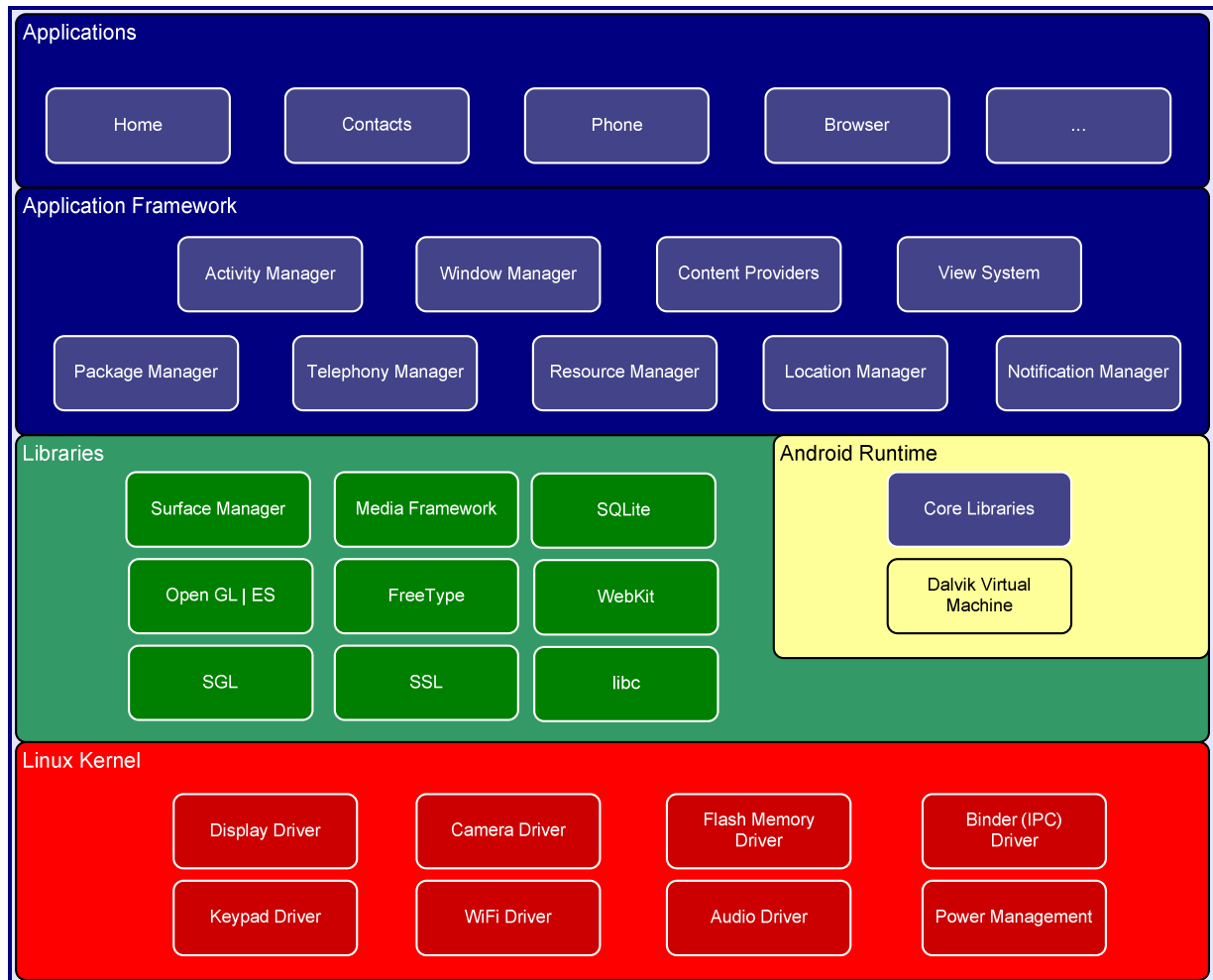


Figura 8: Arquitectura d'Android

## CAPA APLICACIONS

Android proveeix un nucli d'aplicacions que permeten diferents serveis com: e-mail, SMS, agenda, cartografia, navegador, llibreta d'adreces entre d'altres. Tots aquests estan escrits en Java.

## CAPA FRAMEWORK D'APLICACIONS

Aquesta capa ofereix als desenvolupadors la possibilitat de crear noves aplicacions. És possible l'accés al framework a través del conjunt d'API's utilitzades pel nucli d'aplicacions.

## LIBRARIES

Android proveeix un conjunt de llibreries en C/C++ utilitzades per diversos components del sistema. Existeixen llibreries per a diferents propòsits de sistema, multimèdia, superfície, web, gràfics 2D i 3D i motor de base de dades.

## ANDROID RUNTIME

Són un conjunt de llibreries de nucli que permeten l'ús de les llibreries de Java.

## LINUX KERNEL

Android confia en Linux el sistema de seguretat, gestió de memòria, gestió de processos, xarxa i drivers. És la capa que separa el programari del maquinari.

### **4.2.3. Descripció de l'arquitectura de software del sistema desenvolupat**

Com s'ha descrit anteriorment el sistema de monitoratge i control de sensors es basa en una xarxa P2P i utilitza JXTA com a plataforma. L'aplicació que interactua amb els sensors es troba en el mateix sensor i és una aplicació ubicada a nivell d'aplicacions de l'arquitectura de software JXTA.

Les aplicacions de monitorització i control de l'estat del servei que ofereixen els sensors es realitza sobre Android. Aquesta aplicació que es trobarà al nivell d'aplicacions d'Android permetrà, per mitjà de la interfície gràfica que ens ofereix l'entorn, monitoritzar els diferents sensors ubicats a les motes sensores, fer cerques referents a l'estat de les motes sensores o dels seus sensors i controlar, modificant l'estat si cal, les diferents motes sensores o els seus sensors.

Les aplicacions en Android que es realitzen tenen la particularitat que no implementaran JXTA i, per tant, utilitzaran la versió reduïda de JXTA (JXME). D'aquesta forma es permet un abast de dispositius mòbils més ampli ja que les

aplicacions poden ser utilitzades també per aquells dispositius que no tenen capacitat per implementar JXTA com per exemple dispositius CDC<sup>3</sup> i CDLC<sup>4</sup>.

---

<sup>3</sup> CDC – Sigles que corresponen a Connected Device Configuration

<sup>4</sup> CLDC – Sigles que corresponen a Connected Limited Device Configuration



### **4.3. Disseny del sistema**

El sistema s'ha definit en dos grans parts: l'aplicació de Gestió de la Mota Sensora i l'aplicació de Monitorització i Control de la Mota Sensora.

L'aplicació de Gestió de la Mota Sensora té com a objectiu gestionar les dades i serveis de la mota sensora. Les dades són les corresponents a la mota sensora i els diferents sensors que hi ha a aquesta, i els serveis són els referents al funcionament de la mota sensora dintre de la plataforma JXTA com si d'un Peer més es tractés.

L'aplicació de Monitorització i Control de la Mota Sensora és una aplicació en Android que actua com a Proxied Peer. Aquesta aplicació permet a més gestionar la connexió de l'aplicació a la plataforma JXTA a través del Relay Peer que li proveeix l'accés necessari.

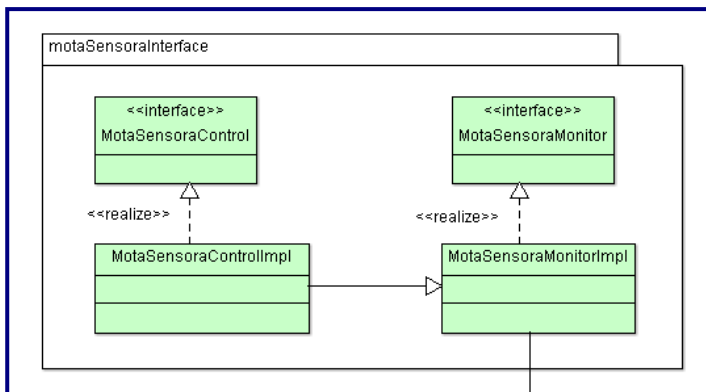
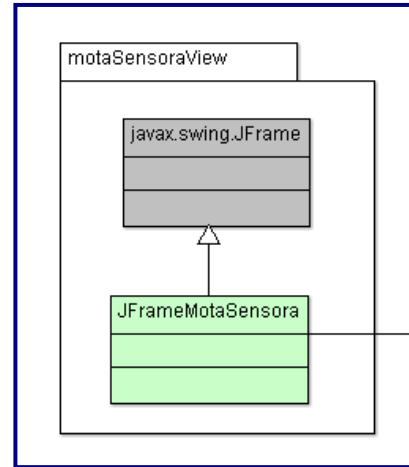
#### **4.3.1. Diagrama de disseny estàtic**

D'acord a la descripció del disseny del sistema, el diagrama de disseny estàtic s'ha definit en dos parts que corresponen a les dues aplicacions que s'han realitzat per a la gestió del sistema.

#### **DIAGRAMA DE DISSENY ESTÀTIC DE L'APLICACIÓ DE GESTIÓ DE LA MOTA SENSORA**

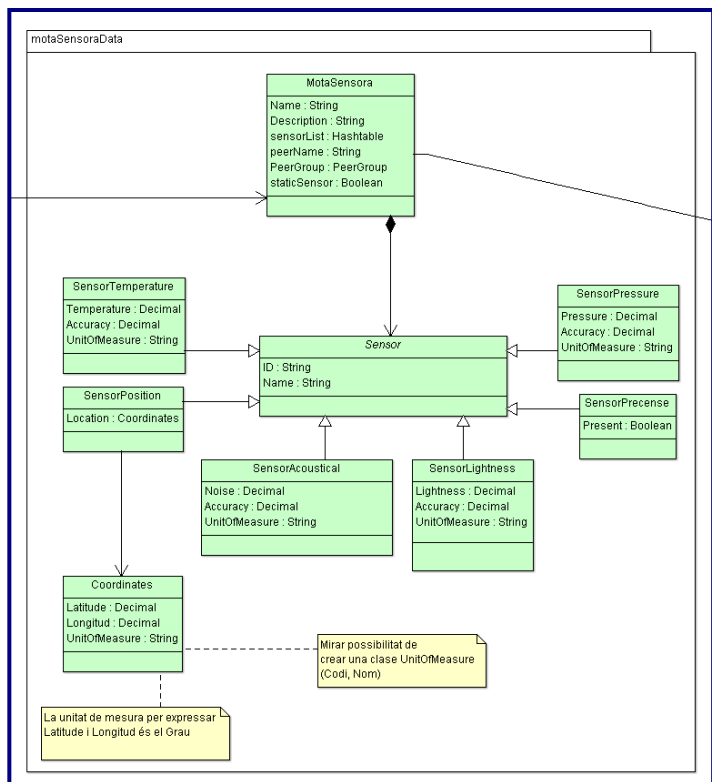
El diagrama de disseny estàtic que es planteja per a l'aplicació separa en diferents paquets les classes necessàries per a l'aplicació tenint en compte el propòsit de les classes que contenen. Tot i que no afecten a la funcionalitat de l'aplicació, es pretén donar una visió més nítida del disseny que es proposa per a l'aplicació. El diagrama de classes dividit en paquets queda de la següent manera:

- MotaSensoraView: És la interfície gràfica que ofereix una mota sensora. Permet arrencar-la, definir i configurar els diferents sensors que la formen, i seguir la seva execució.

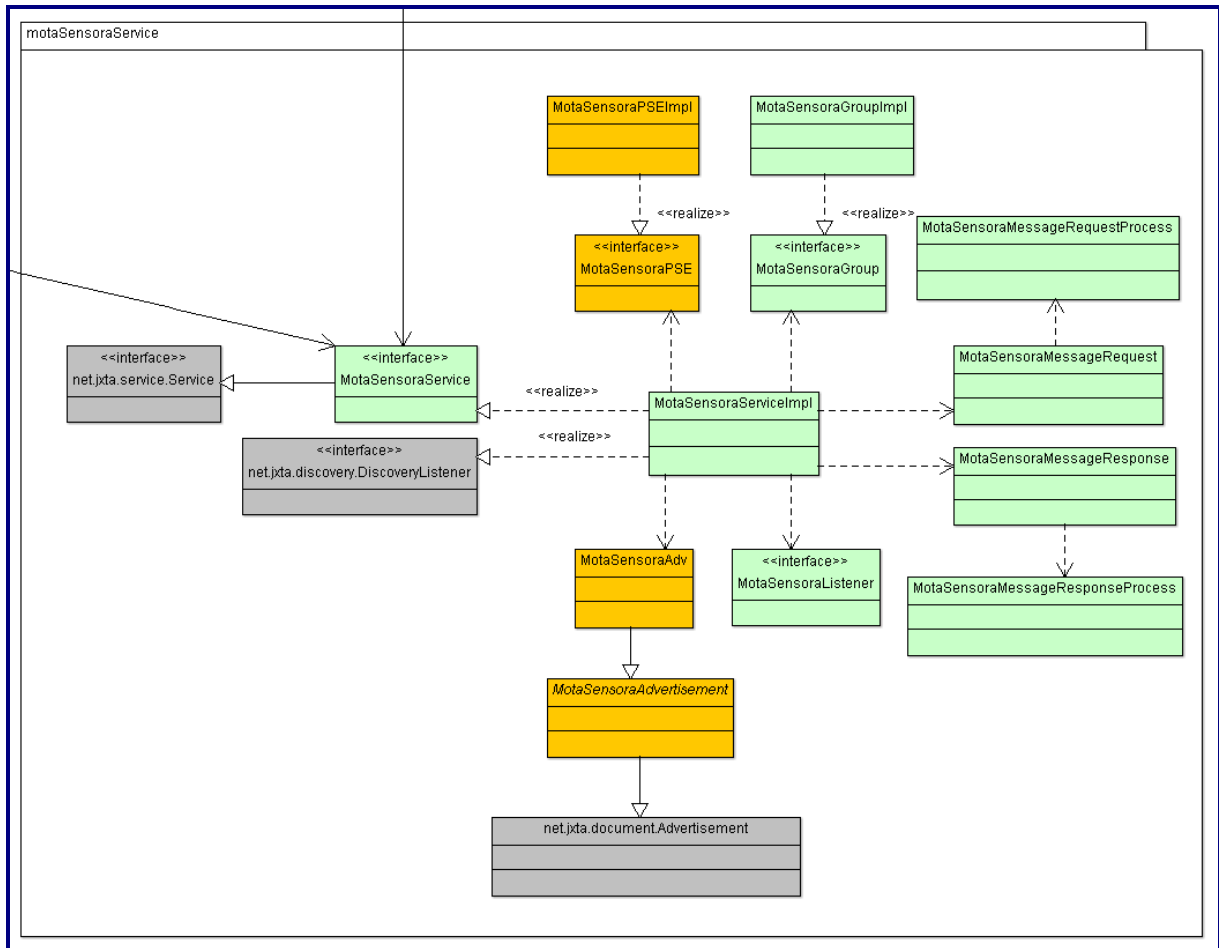


- MotaSensoraInterface: Les classes que la formen ofereixen les operacions per a la consulta i la modificació de les dades de la mota sensora i dels sensors.

- MotaSensoraData: Són les classes que defineixen les característiques d'una mota sensora i dels diferents sensors que la poden compondre.



- MotaSensoraService: Aquest paquet conté totes les classes necessàries per a que la mota sensora pugui donar servei a la xarxa JXTA: la connexió a la xarxa, la incorporació al peer grup de Motes Sensores, la implementació dels "advertiments" de MotaSensora i la definició de missatges de petició i resposta.



El conjunt del disseny estàtic de l'aplicació queda d'acord a la següent figura:

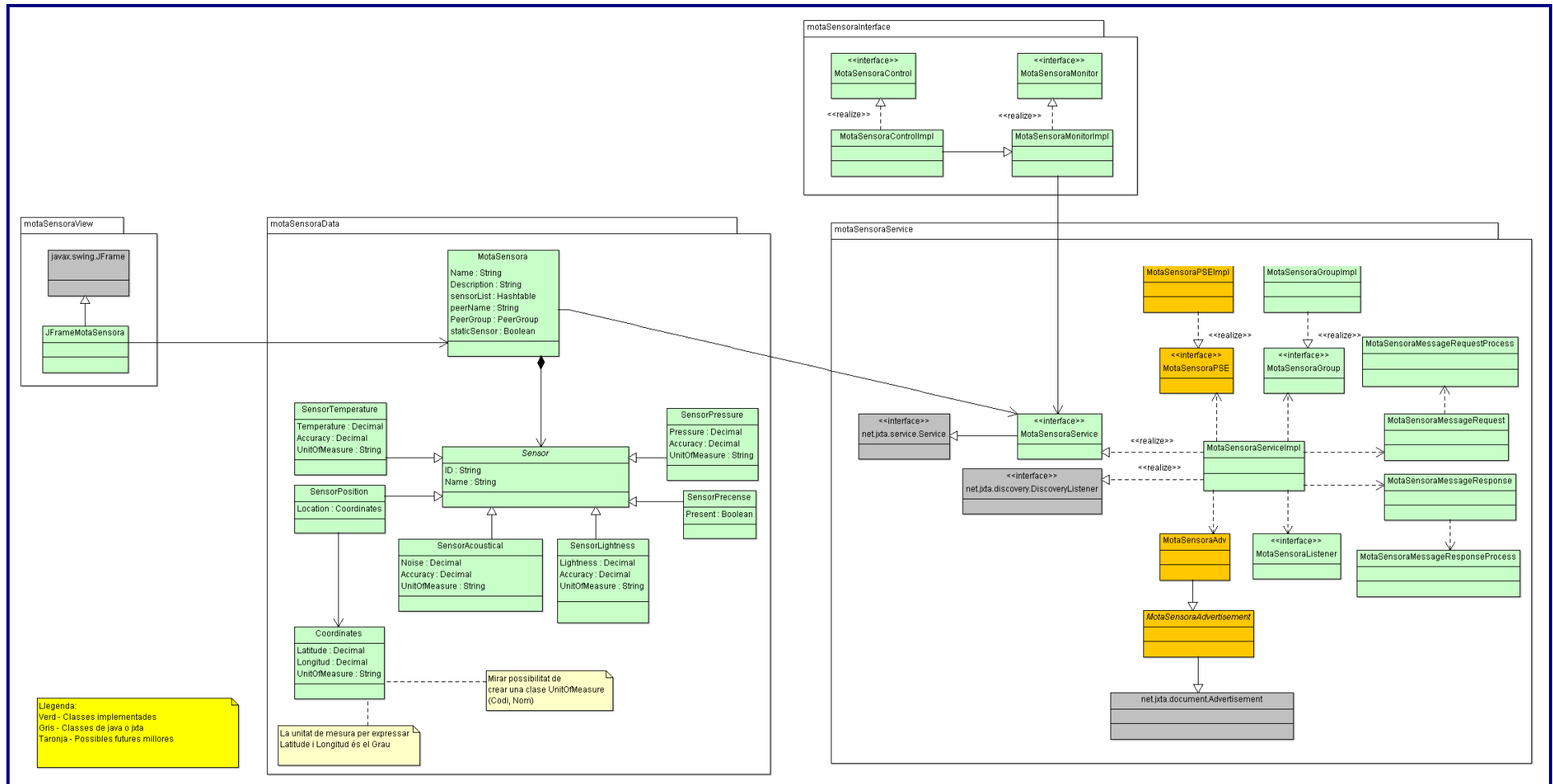


Figura 9: Diagrama disseny estàtic aplicació Gestió Mota Sensora

## DIAGRAMA DE DISSENY ESTÀTIC DE L'APLICACIÓ DE MONITORITZACIÓ I CONTROL DE LA MOTA SENSORA

El diagrama de disseny estàtic que es planteja per a l'aplicació de Monitorització i Control de Mota Sensora està centrada en la classe JXMEProxiedActivity que és la que gestionarà els processos necessaris per a la funcionalitat de Proxied Peer de l'aplicació. Aquesta classe "extend" la classe Activity i permetrà accedir a la resta de interfícies gràfiques necessàries per a la monitorització i control de la mota sensora.

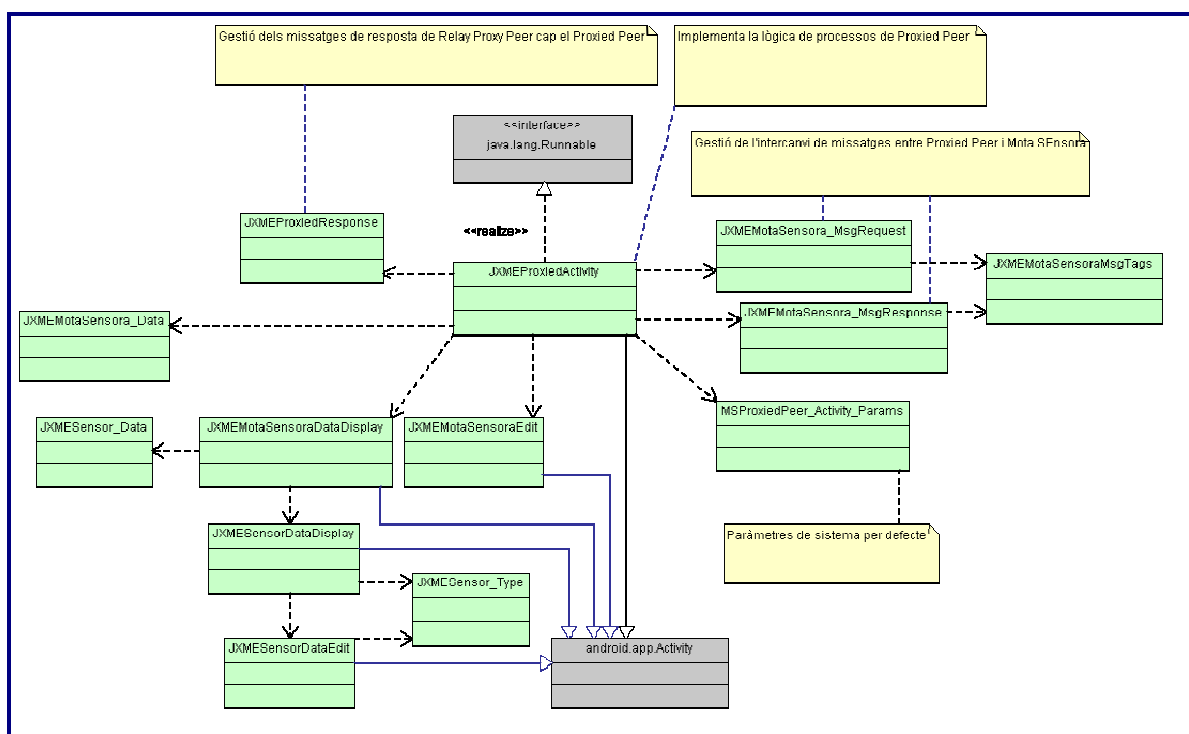


Figura 10: Diagrama disseny estàtic aplicació Monitorització i Control de Mota Sensora

Pel funcionament de l'aplicació cal destacar les següents classes davant de les altres ja que permeten que l'aplicació d'Android es comporti com un Proxied Peer.

**JXMEProxiedActivity:** Gestiona el procés de connexió de l'aplicació, com a Proxied Peer, amb el Relay Peer. A més ofereix l'accés a la resta d'activitats de l'aplicació que contenen les interfícies gràfiques

JXMEProxiedResponse: Permet processar totes aquelles respostes produïdes durant el procés de connexió del Proxied amb el Relay Peer.

JXMEMotaSensora\_MsgRequest: Aquesta classe genera el missatge de petició que realitza el Proxied Peer a qualsevol mota sensora de la xarxa JXTA. Aquestes peticions són missatges que contenen consultes (query) o modificacions (update).

JXMEMotaSensora\_MsgResponse: Classe que processa el missatge rebut d'una mota sensora i que permet obtenir les dades que s'han demanat a la petició o el resultat d'una petició realitzada.

## DIAGRAMA DE SEQÜÈNCIA

Durant l'inici les aplicacions del sistema estableixen el procés de connexió que les ha d'unir a la xarxa JXTA. Aquest procés és elemental pel funcionament de les aplicacions que formen el sistema de monitorització i control de les motes sensores. Els següents diagrames de seqüència defineixen el funcionament dels processos de connexió de les aplicacions a la xarxa JXTA.

L'aplicació de Gestió de la Mota Sensora inicia una sèrie de serveis que li permeten mantenir la connexió amb la xarxa JXTA. Durant el procés de connexió la mota sensora s'afegeix al grup de Motes Sensores que és el Peer Group al que pertanyen totes les motes sensores. Prèviament, es comprova l'existència del grup per crear-lo si és necessari. Si existeix la mota sensora s'afegeix al grup. Seguidament comença a obtenir alguns serveis necessaris pel funcionament de la mota sensora a la xarxa JXTA a través de la funció `getServices`: el `discovery service` i el `pipe service`. Una vegada iniciats els serveis, procedeix a publicar els anuncis a la xarxa JXTA a fi d'indicar l'existència de la mota sensora al Peer Group de motes sensores. Així mateix es crea un canal d'entrada (Input Pipe) que ha de servir a la mota sensora per rebre aquelles peticions que se li demanin.

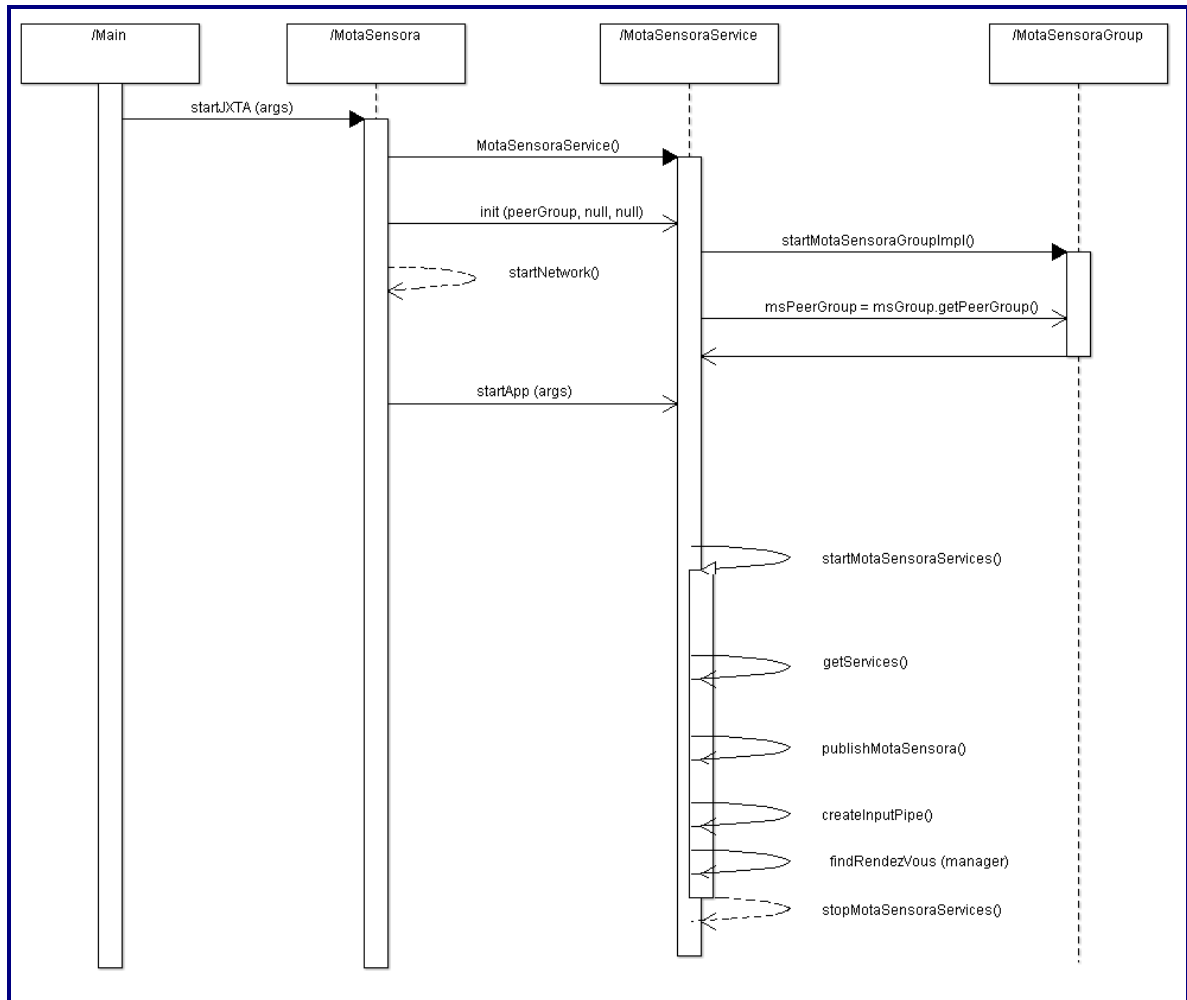


Figura 11: Diagrama de seqüència Procés de Connexió aplicació de Gestió de Mota Sensora

Per la seva banda el procés de connexió de l'aplicació de Monitorització i Control de les Motes Sensores és diferent, ja que es tracta d'un Proxied Peer que es connectarà a la xarxa JXTA a través d'un Relay Peer. El Relay Peer serà qui "representarà" al Proxied Peer a la xarxa JXTA i li farà arribar tots els missatges dirigits a ell. De la mateixa forma el Relay Peer posarà a la xarxa JXTA totes aquelles peticions que el Proxied Peer pugui fer. El procés de connexió del Proxied Peer al Relay Peer es defineix al següent diagrama de seqüència.

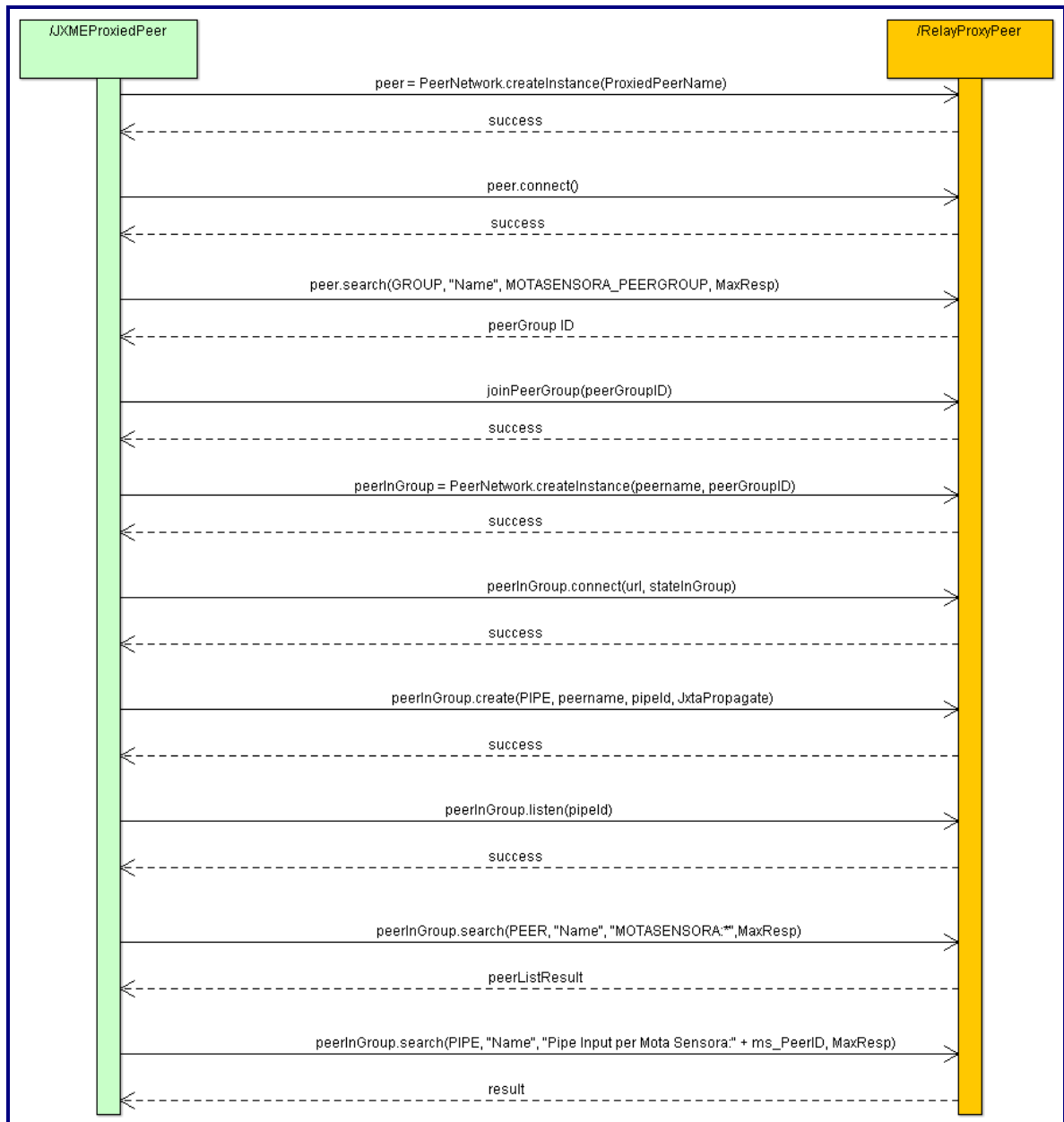


Figura 12: Diagrama de seqüència Procés de Connexió Proxied Peer a Relay Peer

Bàsicament, el procés el que ha de fer és connectar al Relay Peer generant una nova instància de la xarxa. S'uneix al Peer Group per defecte "NetPeerGroup" i comprovarà l'existència del grup de motes sensores. Si existeix s'uneix a ell i instància la xarxa per aquest nou grup. D'aquesta forma els missatges que el Proxied Peer enviï el Relay Peer els enviarà al Peer Grup de motes sensores. Un cop el Proxied Peer és al grup es busquen les motes sensores existents. Quan es



selecciona la mota sensora a la que es connectarà es cercarà el Pipe que li correspon a aquesta mota sensora a fi de poder enviar les peticions necessàries.

## 4.4. Disseny de l'entorn gràfic

El sistema ha de disposar d'un entorn gràfic que permeti realitzar totes les operacions necessàries pel seu funcionament. Com s'ha comentat amb anterioritat, el sistema conté dos aplicacions, una per a la gestió de la mota sensora i una altra per a la seva monitorització i control. Els següents apartats descriuen el disseny gràfic de les aplicacions.

### 4.4.1. Disseny entorn gràfic de la gestió de sensors

Es tracta bàsicament d'uns formularis que ha de permetre definir tant la mota sensora com els seus sensors, així com aspectes referents a la connexió de la mota sensora en la xarxa JXTA i a la seva execució en cas necessari. L'aspecte del disseny és el següent:

- La pestanya bàsics ha de permetre especificar aspectes de la mota sensora i de la seva connexió a la xarxa JXTA

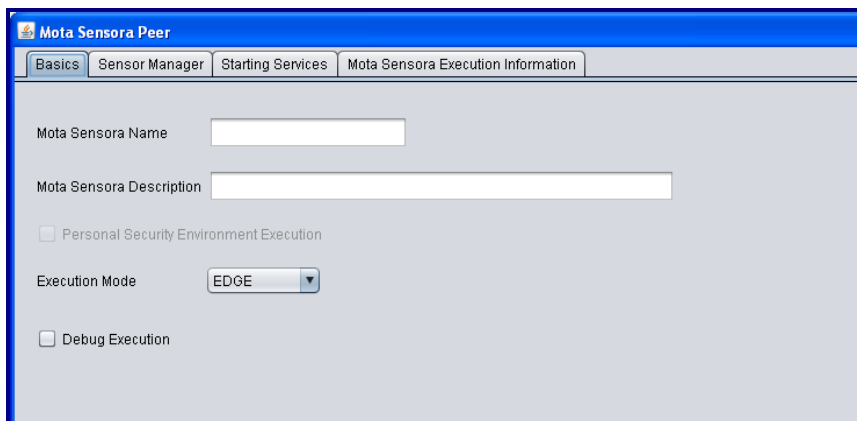


Figura 13: Aplicació gestió Motes Sensores – Pestanya Bàsics

- La pestanya "Sensor Manager" ha de servir per especificar i definir els sensors que conté la mota sensora.

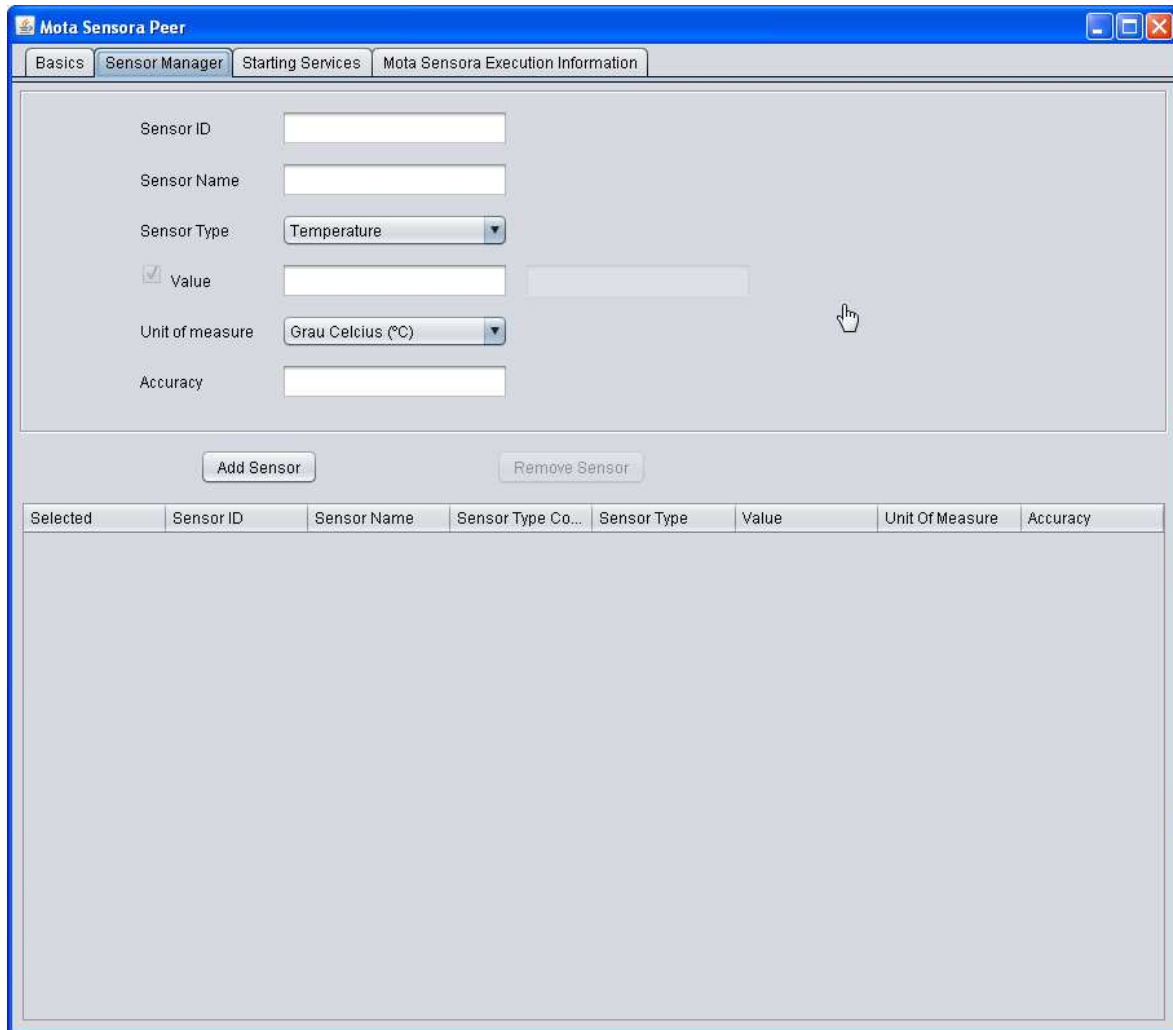


Figura 14: Aplicació gestió Motes Sensores – Pestanya Bàsics

- La pestanya Starting Services permetrà posar en marxa la mota sensora un cop estigui definida

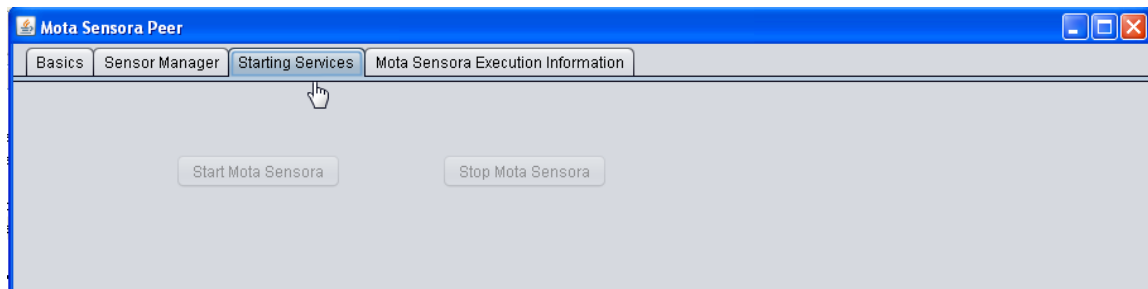


Figura 15: Aplicació de gestió de Motes Sensores – Pestanya Starting Services

- Finalment la pestanya "Mota Sensora Execution Information" permet el seguiment de l'execució de la mota sensora

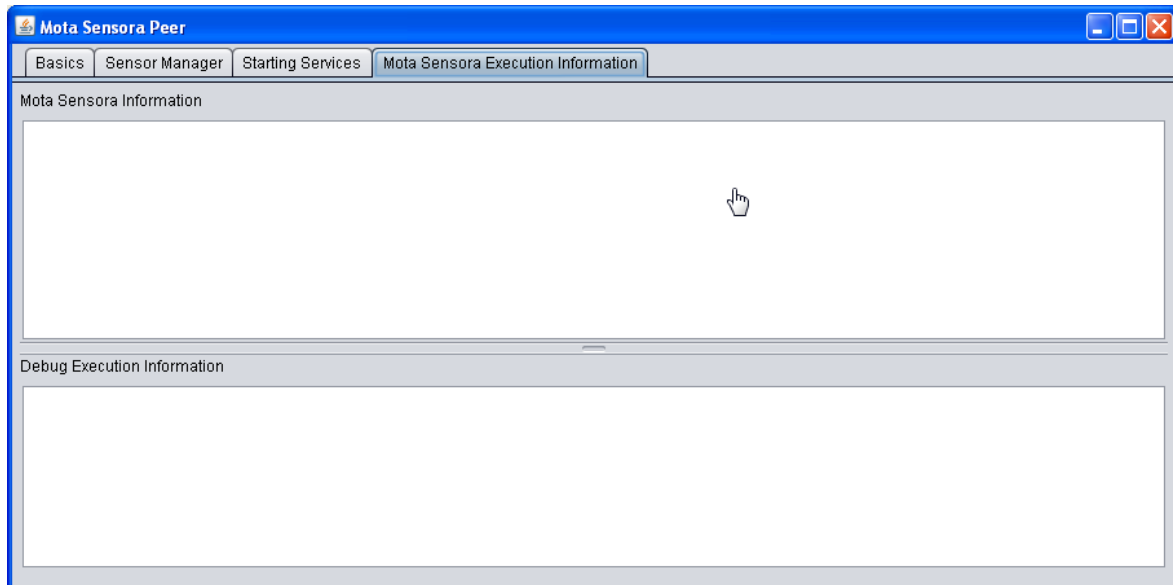


Figura 16: Aplicació de gestió de Motes Sensores – Pestanya Mota Sensora Execution Information

#### 4.4.2. Disseny entorn gràfic Monitorització i Control de Motes Sensores

L'entorn gràfic que s'ha definit per al sistema es divideix en tres: la part de connexió, les interfícies gràfiques de mostra de dades (display) i les interfícies gràfiques d'edició de dades (edit).

##### ENTORN GRÀFIC DE CONNEXIÓ

Es tracta d'una pantalla de configuració de la connexió on s'ha de poder definir la parametrització de la connexió del Proxied Peer al Relay Peer que s'especifiqui.

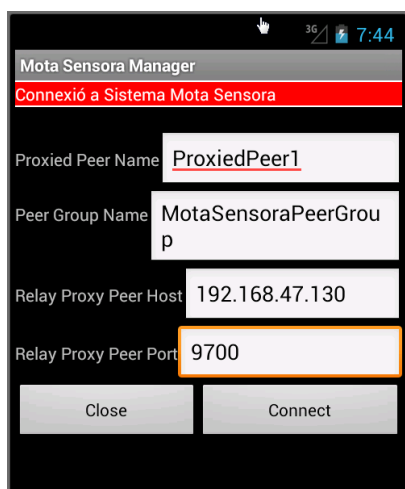


Figura 17: Monitorització i control de Motes Sensores – Interfície de connexió

### 4.4.3. Entorn gràfic de monitorització

L'entorn gràfic de monitorització ha de possibilitar veure els valors dels diferents sensors de la mota sensora, veure el seu estat i accedir si cal a l'entorn gràfic de control de l'aplicació.

La següent figura mostra les interfícies de monitorització d'una mota sensora i d'un sensor:

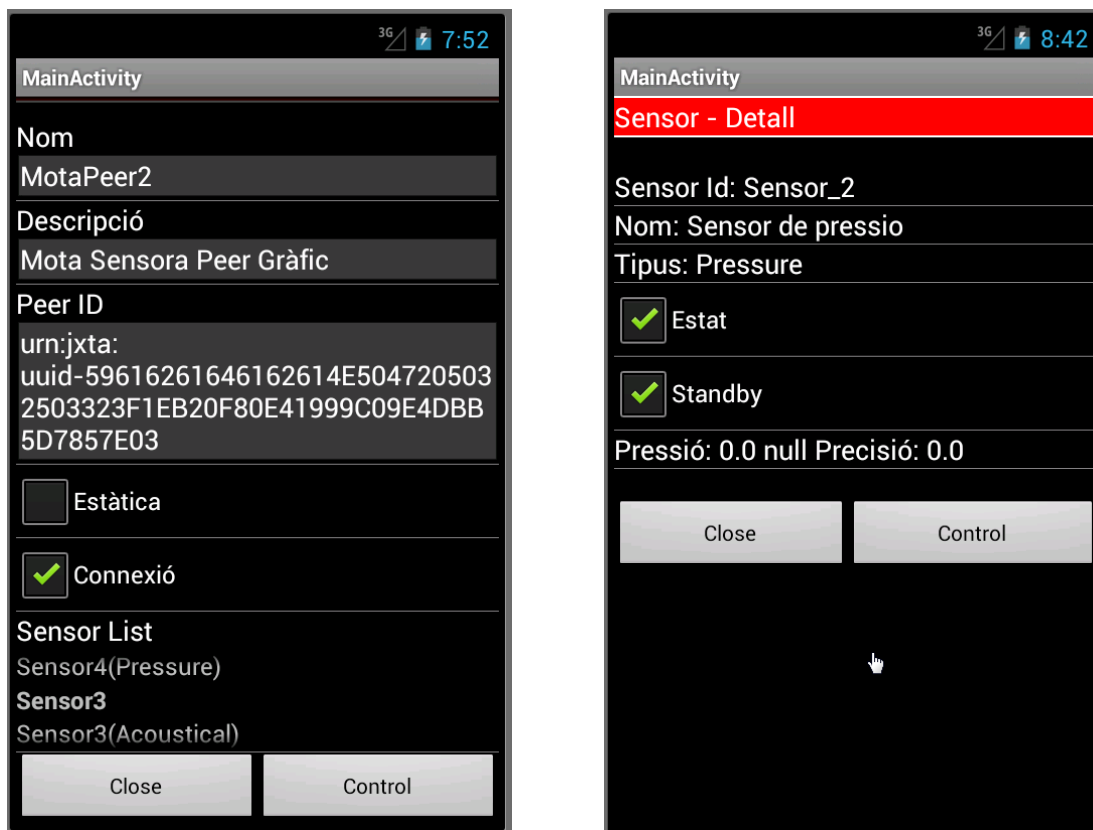


Figura 18: Vistes de l'entorn gràfic de monitorització

### 4.4.4. Entorn gràfic de control

L'entorn gràfic de control ha de permetre a l'usuari accedir a les opcions de modificació de paràmetres referents a la mota sensora o als seus sensors.

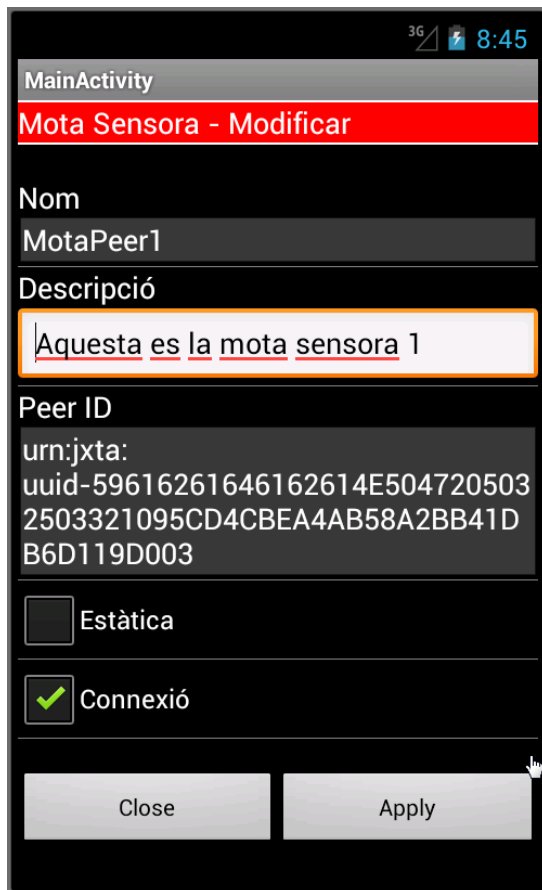


Figura 19: Vistes de l'entorn gràfic de control

#### 4.4.5. Intercanvi de missatges entre Mota Sensora i Proxied Peer

El procés de monitorització i control de les motes sensores per part del Proxied Peer es realitza mitjançant un intercanvi de missatges entre aquests dos components del sistema. L'intercanvi de missatges ha de garantir la funcionalitat de l'aplicació, per tant s'han tingut en compte aspectes de confirmació de les operacions realitzades i control d'errors.

Els missatges que s'han definit es descriuen seguidament:

Missatge de Petició: Aquest missatge permet al Proxied Peer realitzar una petició de l'operació a realitzar a la mota sensora. El contingut del missatge es pot observar en la següent taula:

Atribut	Etiqueta	Descripció	Tipus	Valors
Identificador de missatge	MSG_ID	Permet identificar el missatge de forma única	String	
Tipus de Missatge	MSG_TYPE	Permet identificar de quin tipus és la petició que es farà	String	QUERY / UPDATE
Àmbit	SCOPE	Àmbit al que aplica la petició. Pot aplicar a algun atribut de la mota sensora o a algun sensor d'aquesta	String	MOTASENSORA / SENSOR
Identificador Sensor	SENSOR_ID	Si la petició és per a un sensor determinat indica l'identificador del sensor	String	
Identificador	ID	Identificador de la Pipe que realitza la petició. Té com a propòsit facilitar el procés de resposta	String	Jxta:urn:.....
Atribut	ATTRIBUTE	Atribut sobre el que es realitza la petició. Si el	String	

		tipus de missatge és QUERY es reserva l'atribut "ALL" per expressar que es volen tots els atributs de l'àmbit (MotaSensora o Sensor)		
Valor	VALUE	Valor que se li vol assignar a l'atribut.	String	

Taula 13: Contingut del missatge de petició

El missatge de petició format té com a etiqueta MSG\_REQUEST i queda segons els següents exemples:

```
<MSG_REQUEST>
  <MESSAGE_ID>id1</MESSAGE_ID>
  <MSG_TYPE>QUERY</MSG_TYPE>
  <SCOPE>MOTASENSORA</SCOPE>
  <ID>jxta:urn.....</ID>
  <ATTRIBUTE>Connected</ATTRIBUTE>
</MSG_REQUEST>
```

Taula 14: Exemple de missatge de petició de consulta mota sensora

```
<MSG_REQUEST>
  <MESSAGE_ID>id1</MESSAGE_ID>
  <MSG_TYPE>QUERY</MSG_TYPE>
  <SCOPE>SENSOR</SCOPE>
  <SENSOR_ID>Sensor_1</SENSOR_ID>
  <ID>jxta:urn.....</ID>
  <ATTRIBUTE>Temperature</ATTRIBUTE>
</MSG_REQUEST>
```

Taula 15: Exemple de missatge de petició de consulta de sensor



```
<MSG_REQUEST>
  <MESSAGE_ID>id1</MESSAGE_ID>
  <MSG_TYPE>UPDATE</MSG_TYPE>
  <SCOPE>MOTASENSORA</SCOPE>
  <ID>jxta:urn.....</ID>
  <ATTRIBUTE>Connected</ATTRIBUTE>
  <VALUE>False</VALUE>
</MSG_REQUEST>
```

Taula 16: Exemple de missatge de petició de modificació de dades de mota sensora

```
<MSG_REQUEST>
  <MESSAGE_ID>id1</MESSAGE_ID>
  <MSG_TYPE>UPDATE</MSG_TYPE>
  <SCOPE>SENSOR</SCOPE>
  <SENSOR_ID>Sensor_1</SENSOR_ID>
  <ID>jxta:urn.....</ID>
  <ATTRIBUTE>Temperature</ATTRIBUTE>
  <VALUE>40.0</VALUE>
</MSG_REQUEST>
```

Taula 17: Exemple de missatge de petició de modificació de dades de sensor

Missatge de resposta: Aquest missatge permet a la mota sensora respondre quin ha estat el resultat de la petició sol·licitada pel Proxied Peer. El contingut del missatge de resposta depèn del tipus de petició realitzada. Els tipus que poden ser retornats s'especifiquen seguidament.

### Consultes/Modificacions d'un determinat camp

Si es fa una consulta / modificació d'un determinat camp de la mota sensora o del sensor, el missatge de resposta s'especifica a la següent taula:

Atribut	Etiqueta	Descripció	Tipus	Valors
Identificador de missatge	MSG_ID	Permet identificar el missatge de forma única	String	
Valor	VALUE	Resulta que es retorna de la consulta o valor que s'ha assignat	String	
Execució	EXECUTION	Especifica si el procés d'execució ha estat correcte	String	OK / ERROR

Taula 18: Contingut del missatge de resposta

El missatge de resposta format té com a etiqueta MSG\_RESPONSE i queda segon el següent exemple:

```
<MSG_RESPONSE>
  <MESSAGE_ID>id1</MESSAGE_ID>
  <VALUE>False</VALUE>
  <EXECUTION>OK</EXECUTION>
</MSG_RESPONSE>
```

Taula 19: Exemple de missatge de resposta del tipus Mota Sensora

```
<MSG_RESPONSE>
  <MESSAGE_ID>id1</MESSAGE_ID>
  <VALUE>40.0</VALUE>
  <EXECUTION>OK</EXECUTION>
</MSG_RESPONSE>
```

Taula 20: Exemple de missatge de resposta del tipus sensor

### Petició de totes les dades d'una mota sensora

En el cas que la consulta es realitzi a l'àmbit d'una mota sensora i s'especifiqui que es volen consultar totes les dades (atribut amb valor "ALL") la resposta que

s'obtindrà serà un missatge amb totes les dades de la mota sensora i els sensors que aquesta conté. En referència al sensors únicament es mostraran les dades de l'identificador, el nom i el tipus de sensor de què es tracta perquè serveixi d'ajuda a l'usuari per saber amb més precisió quines són les magnituds que la mota sensora pot mesurar. Al següent quadre es pot observar un exemple de resposta de totes les dades d'una mota sensora:

```
<MSG_RESPONSE>
  <MESSAGE_ID>Id1</MESSAGE_ID>
  <NAME>MotaPeer1</NAME>
  <PEER_ID>urn:jxta:uuid-59616261646162614E50472.....</PEER_ID>
  <DESCRIPTION>Aquesta es la mota sensora 1</DESCRIPTION>
  <CONNECTED>>true</CONNECTED>
  <STATIC>>false</STATIC>
  <SENSOR_LIST>
    <SENSOR>
      <SENSOR_ID>
        Sensor_1
      </SENSOR_ID>
      <SENSOR_NAME>
        Sensor de presencia
      </SENSOR_NAME>
      <TYPE>
        6
      </TYPE>
    </SENSOR>
    <SENSOR>
      <SENSOR_ID>
        Sensor_2
      </SENSOR_ID>
      <SENSOR_NAME>
        Sensor de posicio
      </SENSOR_NAME>
    </SENSOR>
  </SENSOR_LIST>
</MSG_RESPONSE>
```

```
<TYPE>
    2
</TYPE>
</SENSOR>
</SENSOR_LIST>
</MSG_RESPONSE>
```

Taula 21: Exemple de missatge de resposta a la petició de totes les dades d'un sensor

### *Petició de totes les dades d'un sensor*

Si la consulta es realitza a l'àmbit d'un sensor i s'especifica que es volen consultar totes les dades (atribut amb valor "ALL") la resposta que s'obté és un missatge amb totes les dades del sensor. El següent quadre mostra un exemple de resposta de totes les dades d'un sensor:

```
<MSG_RESPONSE>
  <MESSAGE_ID>Id1</MESSAGE_ID>
  <SENSOR_ID>Sensor_1</SENSOR_ID>
  <SENSOR_NAME> Sensor de temperatura</SENSOR_NAME>
  <STATUSON>>true</STATUSON>
  <STANDBY>>true</STANDBY>
  <TYPE>1</TYPE>
  <TEMPERATURE>0.0</TEMPERATURE>
  <ACCURACY>0.0</ACCURACY>
  <UNITOFMEASURE>Grau Celcius (°C)</UNITOFMEASURE>
</MSG_RESPONSE>
```

Taula 22: Exemple de missatge de resposta a la petició de totes les dades d'un sensor

#### 4.4.6. Funcionament del procés d'intercanvi de missatges entre Proxied Peer i Mota Sensora

L'intercanvi de missatges entre Proxied Peer i Mota Sensora es realitza a través de Pipes. La mota sensora i el Proxied Peer obren un canal d'entrada de missatges quan estableixen la connexió amb la xarxa JXTA. La mota sensora informa de l'identificador del pipe d'entrada a l'anunci que va publicant a la xarxa JXTA.

Quan el Proxied Peer demana l'anunci d'una determinada mota sensora, podrà conèixer quin és el pipe d'entrada i serà el que utilitzarà en el cas que vulgui monitoritzar o controlar una determinada mota sensora.

Una vegada una mota sensora rep un missatge de petició d'un Proxied Peer, la mota sensora procedeix al seu procés. Durant la interpretació del missatge la mota sensora podrà conèixer quin és el identificador del pipe d'entrada del Proxied Peer que serà on haurà de tornar la resposta a la petició.

Es pot representar el funcionament de l'intercanvi de missatges entre Proxied Peer i mota sensora d'acord a la següent figura:

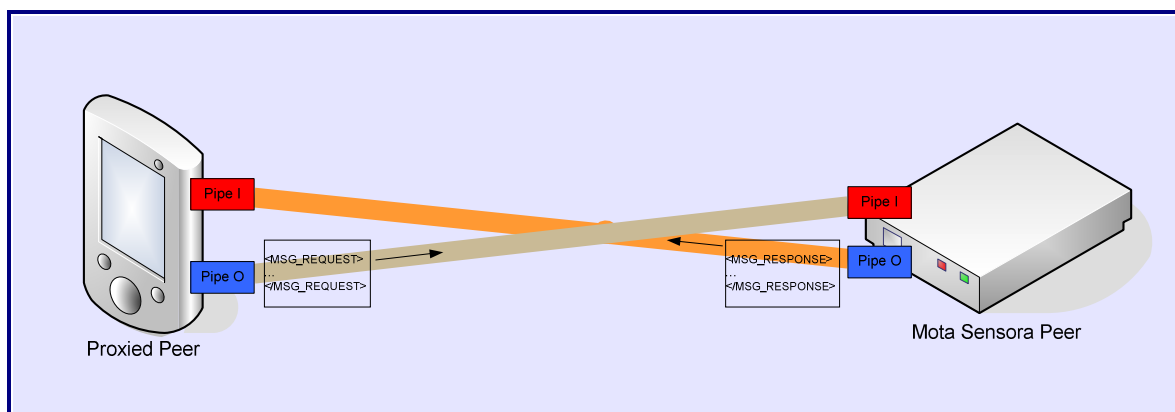


Figura 20: Intercanvi de missatges Proxied Peer – Mota Sensora

La figura mostra el funcionament simplificat de com serà la comunicació entre Mota Sensora i Proxied Peer sense tenir en compte altres elements necessaris de la xarxa JXTA. Cal advertir que la comunicació del Proxied Peer no es fa directament amb la mota sensora, ja que ho fa a través del seu Relay Proxied Peer que és qui li dona accés als serveis de la xarxa JXTA.

## 5. Implementació

Aquest apartat descriu el procés d'implementació, quines han estat les eines i tecnologies utilitzades, els fets remarcables d'aquesta fase i les decisions que s'han hagut de prendre.

### 5.1. Eines i tecnologies utilitzades

La implementació del sistema de monitorització i de control de sensors ha estat realitzada en llenguatge Java. S'ha combinat Java amb la tecnologia JXTA que ens facilita una plataforma peer-to-peer i que ens permet la creació d'una xarxa de motes sensores sobre una base de peers.

La part de mobilitat s'ha implementat utilitzant una aplicació d'Android combinada amb JXME. JXME és la versió reduïda de JXTA, bàsicament implementa la comunicació entre un Proxied Peer i el Relay Proxy Peer que li donarà accés als recursos de JXTA.

Netbeans IDE<sup>5</sup> 7.1.1 – Es tracta d'un entorn de desenvolupament lliure i multiplataforma que permet la creació d'aplicacions en Java. Aquest IDE de Java s'ha utilitzat per a la implementació de la gestió de motes sensores. Aquest software es pot trobar a:

<http://netbeans.org/downloads/start.html?platform=windows&lang=en&option=javaee>

Eclipse IDE version Juno – Igual que l'anterior, es tracta d'un entorn de desenvolupament integrat de codi obert i multiplataforma. Permet facilitar la creació d'aplicacions en Java. La implementació de la gestió del Proxied Peer s'ha realitzat utilitzant aquest IDE juntament amb el SDK d'Android. El software es pot trobar a:

---

<sup>5</sup> IDE: Entorn de Desenvolupament Integrat

[http://www.eclipse.org/downloads/download.php?file=/eclipse/downloads/drops/4/R-4.2.1-201209141800/eclipse-SDK-4.2.1-win32-x86\\_64.zip](http://www.eclipse.org/downloads/download.php?file=/eclipse/downloads/drops/4/R-4.2.1-201209141800/eclipse-SDK-4.2.1-win32-x86_64.zip)

SDK Android 20.0.3 – Es tracta d'una eina per assistir al desenvolupament de programes realitzats habitualment en Java. Aquesta eina s'integra dintre dels IDE Eclipse i Netbeans utilitzats en el desenvolupament del projecte. El SDK d'Android ha assistit en la implementació de la part del Proxied Peer representada en aquest projecte per a l'aplicació d'Android. Aquest software pot trobar-se a: <http://developer.android.com/sdk/index.html#download>

JXTA – Juxtapose és una plataforma peer-to-peer creada per Sun Microsystems. Està definida per un conjunt de protocols basats en XML que permeten als dispositius connectats a la xarxa l'intercanvi de missatges independentment de la topologia de xarxa. JXTA es pot trobar a: <https://svn.java.net/svn/jxta-jxse~svn/>

JXME – Es tracta de la versió per mòbils de JXTA. Aquesta és més abreujada i lleugera i, per tant, molt apropiada per a dispositius mòbils en especial aquells que tenen poca capacitat. JXME es pot trobar a: <http://download.java.net/jxta/jxta-jxme/>

VMWare Workstation 8.0.4 – Es tracta d'un software de virtualització<sup>6</sup> que permet la creació i gestió de màquines virtuals. La creació de màquines virtuals ha facilitat la implementació i la fase de proves del projecte. Tot i que no s'ha utilitzat una versió gratuïta de VMWare en aquest projecte, existeixen algunes que sí ho són i permeten realitzar virtualització.

---

<sup>6</sup> Es tracta de crear una versió virtual d'un recurs tecnològic, per exemple una plataforma de hardware, un sistema operatiu,... Permet tenir varis equips informàtics o pc's en una mateixa màquina

## **5.2. Consideracions durant el procés d'implementació**

Durant la fase d'implementació del projecte s'han hagut de prendre diferents consideracions que han estat importants pel funcionament i resultat final del sistema. A continuació es presenten aquelles que s'ha considerat que han tingut major transcendència.

### CREACIÓ DEL PEER GROUP MOTASENSORA

Com s'ha comentat, el funcionament de la mota sensora a la xarxa peer-to-peer té com a base un peer de JXTA. JXTA permet la formació de grups per facilitar la comunicació entre un conjunt de peers optimitzant així el funcionament de la xarxa JXTA.

Els diferents elements del sistema de control de motes sensores funcionaran sobre un mateix grup. El grup anomenat "MotaSensoraPeerGroup" permet agrupar a tots els elements del sistema de control de motes sensores: aplicació de Gestió de Motes Sensores i aplicació de Monitorització i Control. D'aquesta manera es garanteix que tota la comunicació del sistema es realitza dintre d'un mateix conjunt d'"usuaris" del sistema.

El funcionament del grup "MotaSensoraPeerGroup" és l'habitual de qualsevol grup de peers. El primer component del grup serà qui creï el grup i a partir d'aquí la resta de motes sensores o JXME Proxied's s'afegiran a aquest grup quan es connectin.

Davant la simplicitat del JXME Proxied la seva connexió al grup "MotaSensoraPeerGroup" es farà a través del Relay Proxy Peer al qual es troba connectat.

Bàsicament es tracta d'una connexió on s'instancia la xarxa dues vegades: la primera instància afegeix al JXME Proxied al NetPeerGroup (grup al que pertanyen tots els peers de la xarxa JXTA) i la segona instància afegeix al JXME Proxied al grup "MotaSensoraPeerGroup" on generarà les peticions que faci sobre



el sistema. El procés de connexió es detalla al diagrama de seqüència de la següent figura:

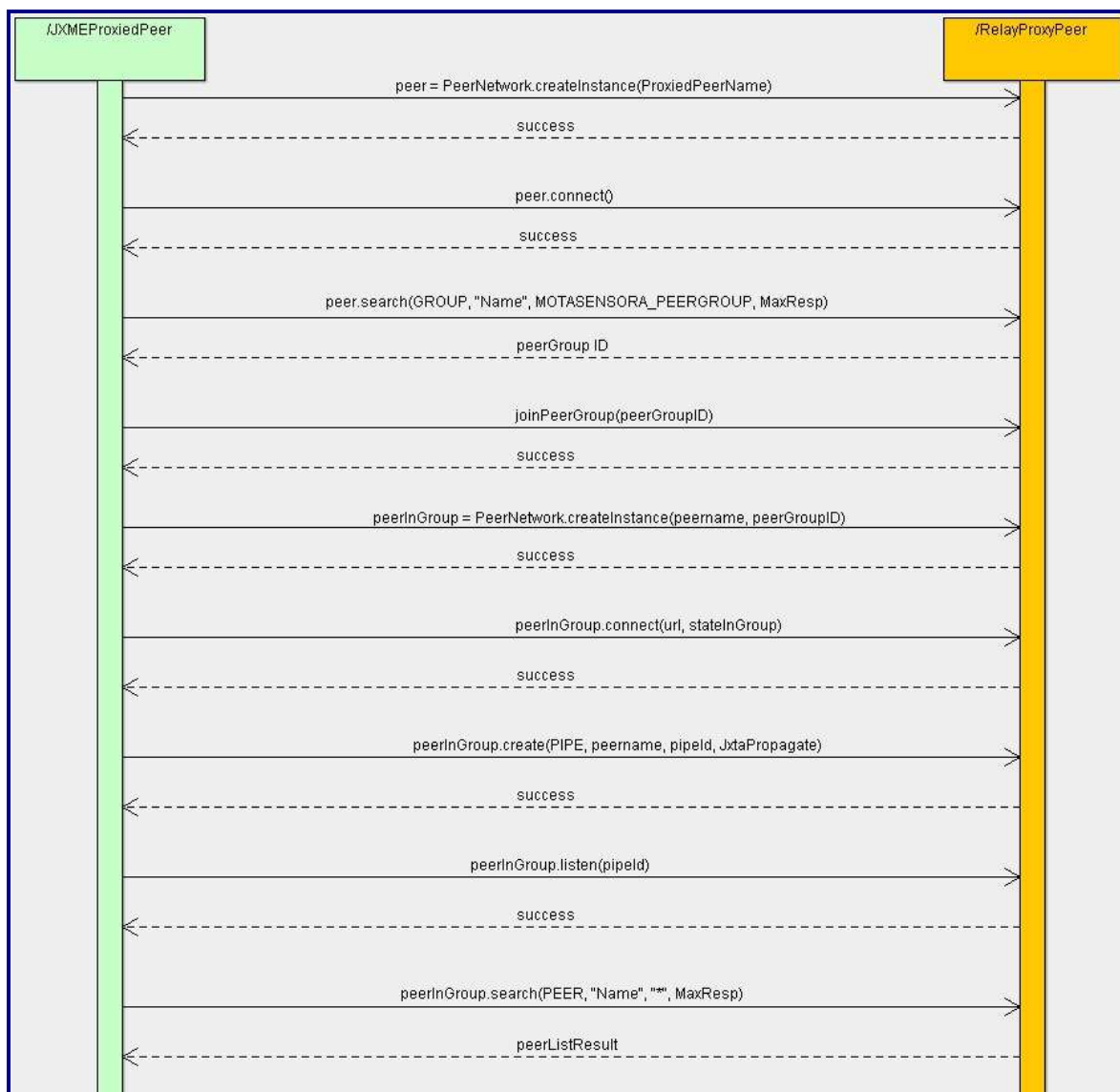


Figura 21: Procés de connexió JXME Proxied i Relay Proxied Peer

El procés de connexió, a més de connectar el JXME Proxied a la xarxa JXTA i al grup de motes sensors, obre un canal de comunicació d'entrada (Pipe) a través del qual el JXME Proxied escoltarà les respostes a les peticions que realitzi. Una vegada obert aquest canal, es realitza la cerca que permet conèixer quins són la resta de components que es troben disponibles al grup i sobre els que es podrà realitzar les peticions que siguin requerides per a la monitorització o control d'una mota sensora.

La següent figura expressa el funcionament trivial de l'arquitectura implementada:

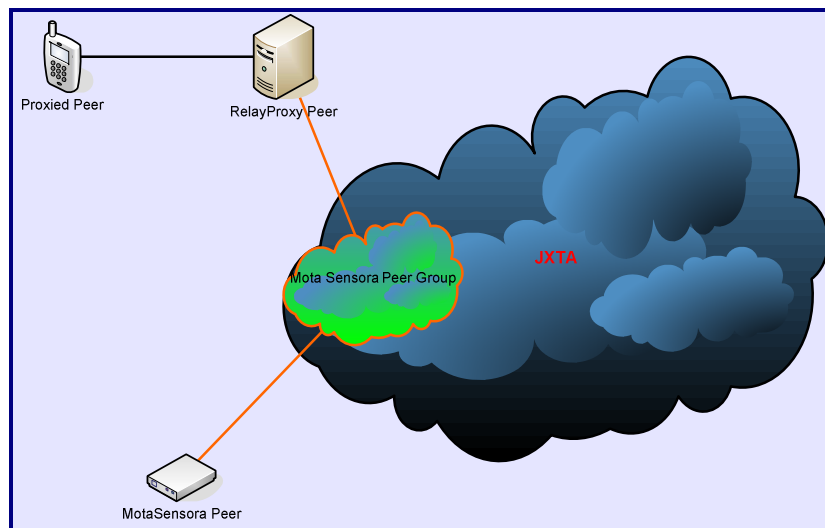


Figura 22: Arquitectura trivial del sistema de Motes Sensors

#### Nivell de seguretat del sistema:

Durant la recollida de requeriments de l'aplicació es va determinar la creació d'un nivell de seguretat a l'aplicació que permetés el funcionament correcte del sistema. Originalment es va realitzar un accés als recursos del grup de mota sensora securitzat, utilitzant Personal Security Environment (PSE). Durant la fase d'integració del JXME Proxied a la xarxa es va veure que degut a que el joc de comandes que permet JXME Proxied és molt elemental, la unió del JXME Proxied al grup de motes sensors no podia ser fàcilment securitzable. Per tant, es considera no invertir recursos en aquest punt i es proposa com a futura millora del sistema.

## GESTIÓ DELS ADVERTISEMENTS

La gestió dels advertisements que s'han de publicar en la comunicació dels diferents elements del sistema ha estat una de les parts en les que s'ha hagut de prendre més decisions durant la fase d'implementació.

### Creació d'un advertisement personalitzat:

Durant la implementació de les motes sensores es va considerar la creació d'un advertisement que hauria de servir per donar informació estàtica de la mota sensora. L'advertisement de motes sensores seria enviat amb determinada freqüència a la resta dels components del grup de motes sensores. D'aquesta forma els components del grup podrien conèixer l'existència de la mota sensora i quins eren els sensors que aquesta podia oferir. A més, l'advertisement permetia conèixer l'identificador del canal d'entrada de missatges (pipe), i que podia ser utilitzat per altres elements del grup, com els Proxied Peers, a fi de realitzar l'intercanvi de missatges necessari per al control de la mota sensora.

D'aquesta forma s'aconseguia l'optimització de la comunicació entre Mota Sensora i Proxied Peer, tenint en compte la baixa capacitat de recursos que pot tenir un dispositiu mòbil que utilitza el Proxied Peer per comunicar-se amb el sistema de sensors.

Es va comprovar que no era possible aquesta comunicació sense tocar la implementació del Relay Peer, que és el qui ha de fer la traducció dels advertisements de la xarxa JXTA a un format HTTP interpretable pel Proxied Peer. Es va valorar la modificació del Relay Peer que era fora de l'abast del projecte i es va decidir desestimar l'ús d'un advertisement personalitzat. Tot i això, es presenta aquest punt com a una futura millora del sistema amb els seus avantatges i inconvenients. El format de l'advertisement de mota sensora personalitzat quedava com es pot observar al següent quadre:

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jxta:MotaSensoraAdvertisement>
<jxta:MotaSensoraAdvertisement xml:space="default" xmlns:jxta="http://jxta.org">
  <PeerID>urn:jxta:uuid-59616261646162614E504720503250330C6A072A59894E759130CEC9F13779BF03</PeerID>
  <Name>MotaPeer1</Name>
  <Description>Aquesta es la mota sensora 1</Description>
  <StaticMota>false</StaticMota>
  <jxta:PipeAdvertisement>
    <Id>urn:jxta:uuid-59616261646162614E504720503250330E836F1D6D8845B084EEDF3AB19D072404</Id>
    <Type>JxtaUnicast</Type>
    <Name>Pipe_MotaPeer1</Name>
    <Desc>Input Pipe per Mota Sensora MotaPeer1</Desc>
  </jxta:PipeAdvertisement>
  <SensorList>
    <Sensor>
      <ID>Sensor_1</ID>
      <SensorName>Sensor de temperatura</SensorName>
      <SensorType>1</SensorType>
      <SensorTypeDesc>Temperature</SensorTypeDesc>
    </Sensor>
    <Sensor>
      <ID>Sensor_3</ID>
      <SensorName>Sensor de posicio</SensorName>
      <SensorType>2</SensorType>
      <SensorTypeDesc>Position</SensorTypeDesc>
    </Sensor>
  </SensorList>
</jxta:MotaSensoraAdvertisement>

```

Figura 23: Format de la Mota Sensora Advertisement

### Personalització del Peer Advertisement:

Com a alternativa es planteja la modificació del Peer Advertisement de JXTA. Aquest es personalitza per incloure en els seus serveis la informació del pipe que s'ha d'utilitzar per enviar missatges a aquesta mota sensora. El següent quadre mostra un exemple de contingut del Peer Advertisement després de la modificació:

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jxta:PA>
<jxta:PA xml:space="default" xmlns:jxta="http://jxta.org">
  <PID>
    urn:jxta:uuid-59616261646162614E50472050325033F9155DEC40ED45A5B9FBB0AC575D13403
  </PID>
  <GID>
    urn:jxta:uuid-D9F5FB077208427DA9F6F78C56F20AFA02
  </GID>
  <Name>
    MotaPeer1
  </Name>
  <Desc>
    Aquesta es la mota sensora 1
  </Desc>
  <Svc>
    <MCID>
      urn:jxta:uuid-DEADBEEFDEAFFEEDBABE0000001205
    </MCID>
    <Parm>
      <jxta:PipeAdvertisement xml:space="preserve" xmlns:jxta="http://jxta.org">
        <Id>urn:jxta:uuid-59616261646162614E5047205032503300FF2C4872F840C49A63C147E5BCE8C404</Id>
        <Type>JxtaUnicast</Type>
        <Name>Pipe_MotaPeer1</Name>
        <Desc>Input Pipe per Mota Sensora MotaPeer1</Desc>
      </jxta:PipeAdvertisement>
    </Parm>
  </Svc>
</jxta:PA>

```

Figura 24: Format del Peer Advertisement personalitzat

Una vegada implementat i durant les proves realitzades sobre aquesta part de la implementació, es va poder observar que els advertisements de peer no eren reenviats o traduïts completament pel Relay Proxy Peer al Proxied Peer, i que aquests eren resumits i s'enviava informació molt concreta del Peer: PID, Type, Name. Per tant la inclusió de la informació de la Pipe al Peer Advertisement no era útil i no tenia sentit seguir la implementació per aquesta línia per incloure informació estàtica dels sensors de la mota sensora a la secció de serveis del Peer Advertisement.

#### Utilització del format estàndard d'advertisement:

Com a darrera alternativa es presenta utilitzar els advertisements "estàndard" de JXTA (Peer advertisement i Pipe advertisement). Es modifica l'atribut nom de forma que ens ajudi a que el Proxied Peer tingui capacitat per identificar aquells peers que són motes sensores i associar-los amb la Pipe d'entrada que correspon a la mota sensora. Aquesta Pipe d'entrada és la que fa servir la mota sensora per

rebre els missatges de sol·licitud de consulta o modificació del Proxied Peer o altres motes sensores.

Bàsicament el que es realitza és d'una banda afegir a l'atribut nom del peer advertisement que genera la mota sensora el prefix "MOTASENSORA:". Aquest prefix és eliminat quan la informació arriba al destí, doncs no forma part del nom original que ha pogut donar l'usuari a la mota sensora. El peer advertisement que quedaria com es veu a la següent figura:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jxta:PA>
<jxta:PA xml:space="default" xmlns:jxta="http://jxta.org">
  <PID>
    urn:jxta:uuid-59616261646162614E5047205032503321095CD4CBEA4AB58A2BB41DB6D119D003
  </PID>
  <GID>
    urn:jxta:uuid-401F1C58D3CF4F38AECFF9FC2ACD23E302
  </GID>
  <Name>
    MOTASENSORA:MotaPeer1
  </Name>
  <Desc>
    Aquesta es la mota sensora 1
  </Desc>
</jxta:PA>
```

Figura 25: Format del Peer Advertisement final

D'altra banda el contingut de l'atribut nom del pipe advertisement de la mota sensora està compost pel text "Pipe Input per Mota Sensora:" concatenat amb l'identificador de Peer al que fa referència. D'aquesta forma el Proxied Peer pot fer la cerca d'aquells missatges de Pipe del Peer que vol comunicar. El pipe advertisement quedaria com es pot observar a la següent figura:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jxta:PipeAdvertisement>
<jxta:PipeAdvertisement xml:space="default" xmlns:jxta="http://jxta.org">
  <Id>
    urn:jxta:uuid-59616261646162614E5047205033E6BAA014EA5744CEBBA9FE09C5D64B5404
  </Id>
  <Type>
    JxtaUnicast
  </Type>
  <Name>
    Pipe Input per Mota Sensora:urn:jxta:uuid-59616261646162614E504720503321095CD4CBEA4AB58A2BB41DB6D119D003
  </Name>
  <Desc>
    Input Pipe per Mota Sensora MotaPeer1
  </Desc>
</jxta:PipeAdvertisement>
```

Figura 26: Format del Pipe Advertisement final

Aquesta petita adaptació dels advertisements s'ha combinat amb intercanvi de missatges entre la Mota Sensora i Proxied Peer que permeten obtenir tota la informació de la mota sensora o qualsevol dels seus sensors, tal i com hem vist anteriorment a l'apartat "Intercanvi de missatges entre Mota Sensora i Proxied Peer".

## 6. Manual d'ús del Sistema de Monitorització de Motes Sensores

### 6.1. Aplicació de Gestió de Mota Sensora

#### 6.1.1. Inici de l'aplicació de Gestió de Motes Sensores

L'aplicació de Gestió de Motes Sensores es pot iniciar de dues formes:

a) A través de la classe Main. L'execució únicament posa en funcionament la mota sensora (sense GUI<sup>7</sup>) i es pot fer el seu seguiment a través de la consola del mateix Netbeans o Eclipse:

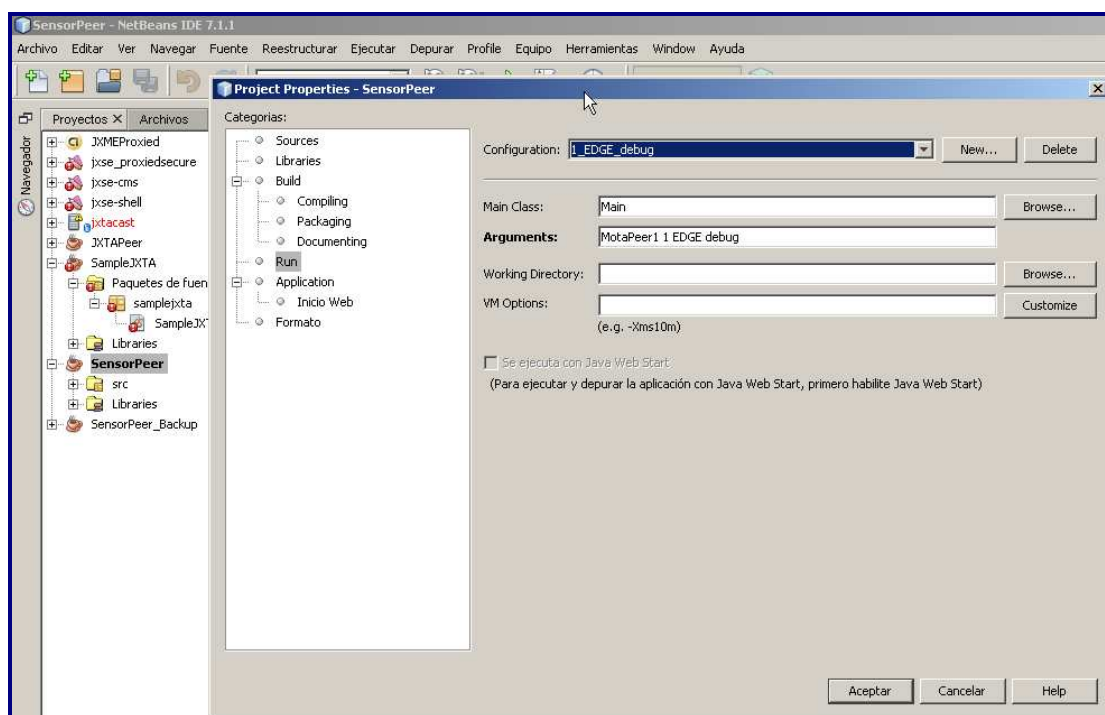


Figura 27: Execució aplicació Gestió de Motes Sensores sense GUI

Un cop posada la mota sensora en funcionament es pot seguir la seva execució a la sortida de l'IDE com es pot veure en el següent detall. Bàsicament es tracta

<sup>7</sup> GUI - Interfície Gràfica d'Usuari



d'una execució d'exemple en el que es defineix una mota sensora i els sensors que conté.



```
Salida - MotaSensoraPeer_Pck (run)
<Id>
urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
</Id>
<Type>
JxtaPropagate
</Type>
<Name>
ProxiedPeer1
</Name>
</jxta:PipeAdvertisement>

[MSINFO] 21:35:36,014 - Output Pipe created: urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
[MSINFO] 21:35:36,014 - Missatge resposta:
<?xml version="1.0"?>
<!DOCTYPE MHC_RESPONSE>
<MHC_RESPONSE>
  <MESSAGE_ID>
    Rqst_SENSOR_ID_20121128203504711
  </MESSAGE_ID>
  <SENSOR_ID>
    Sensor_4
  </SENSOR_ID>
  <SENSOR_NAME>
    Sensor de temperatura
  </SENSOR_NAME>
  <STATUSON>
    true
  </STATUSON>
  <STANDBY>
    true
  </STANDBY>
  <TYPE>
    1
  </TYPE>
  <TEMPERATURE>
    0.0
  </TEMPERATURE>
  <ACCURACY>
    0.0
  </ACCURACY>
  <UNITOFMEASURE/>
</MHC_RESPONSE>

[MSINFO] 21:36:45,952 - Publicant Mota Sensora Advertisements...
[MSTRACE]21:36:45,952 - Publicant Mota Sensora Advertisements...
```

Figura 28: Sortida execució aplicació Gestió de Motes Sensores

b) A través de la classes JFrameMotaSensora. L'execució es realitza per mitjà d'un GUI que permet tota la parametrització de la mota sensora i seguir la seva execució a través de la pestanya "Mota Sensora Execution Information" de la GUI.

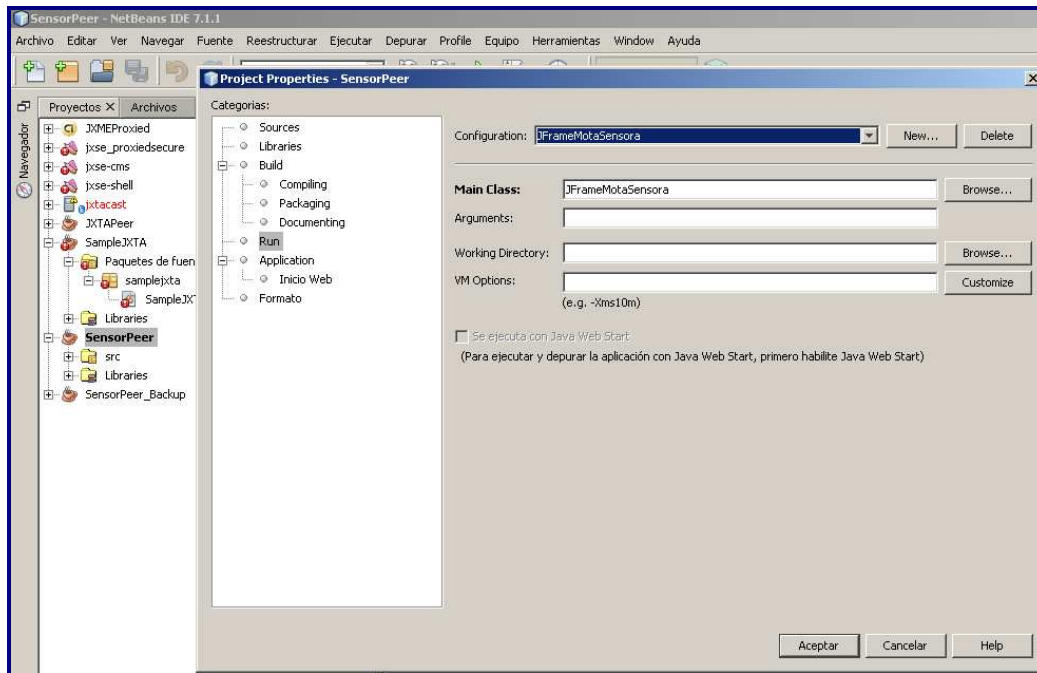


Figura 29: Execució aplicació Gestió de Motes Sensores amb GUI

## EL FUNCIONAMENT DE LA GUI

La pestanya "Basics" permet informar element elementals de la mota sensora: El ID o nom, la descripció, node d'execució (en el nostre cas EDGE) i si volem treure la sortida del debugador a la pestanya "Mota Sensora Execution Information":

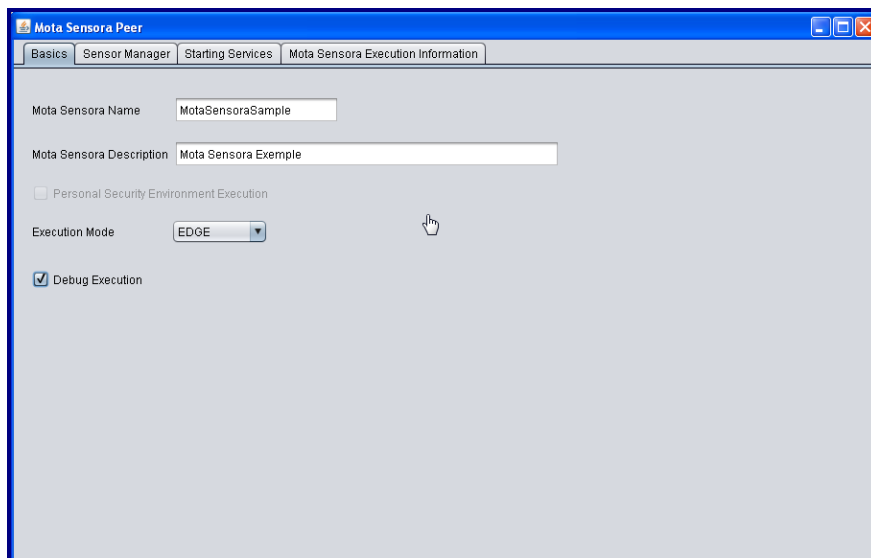


Figura 30: Funcionament GUI – Pestanya Basics

La pestanya "Sensor Manager" permet la parametrització dels diferents sensors que formen la mota sensora i, per mitjà del botó "Add Sensor", afegir-los a la llista de sensors

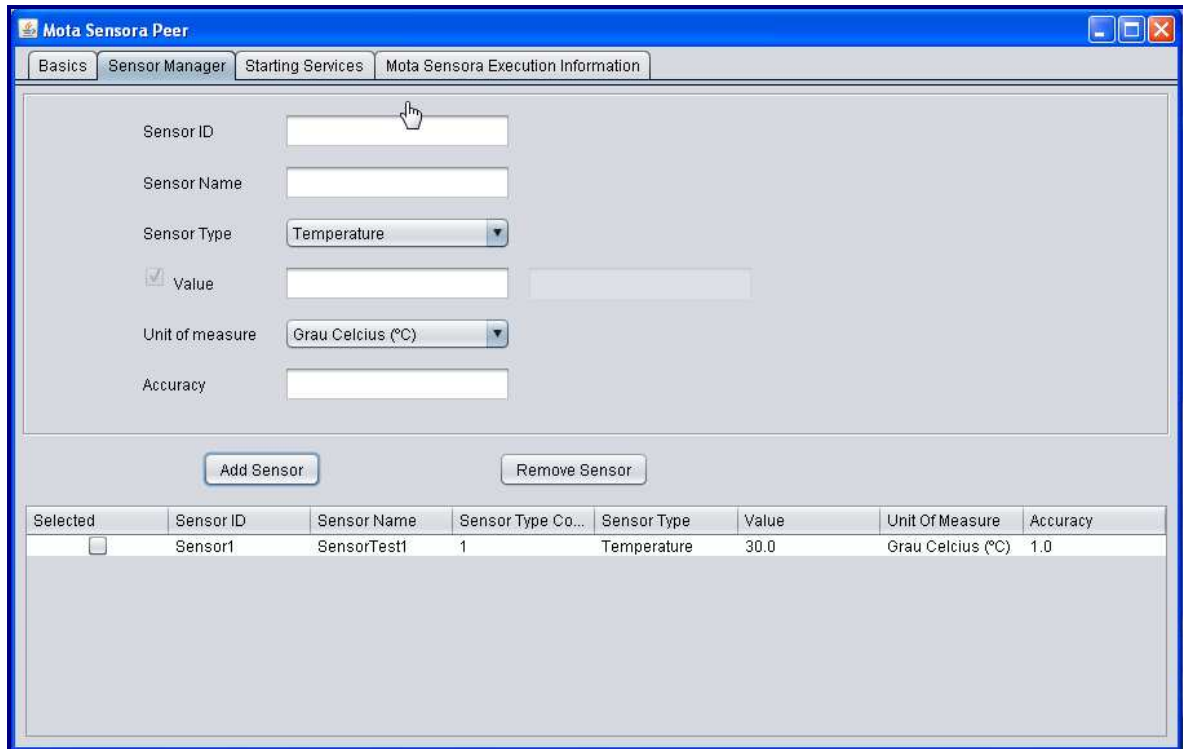


Figura 31: Funcionament GUI – Pestanya Sensor Manager

La pestanya "Starting Services" permet arrencar o aturar una mota sensora. Cal que la mota sensora hagi estat definida:



Figura 32: Funcionament GUI – Pestanya Starting Services

Finalment la pestanya "Mota Sensora Execution Information" permet fer el seguiment de l'execució de la mota sensora:

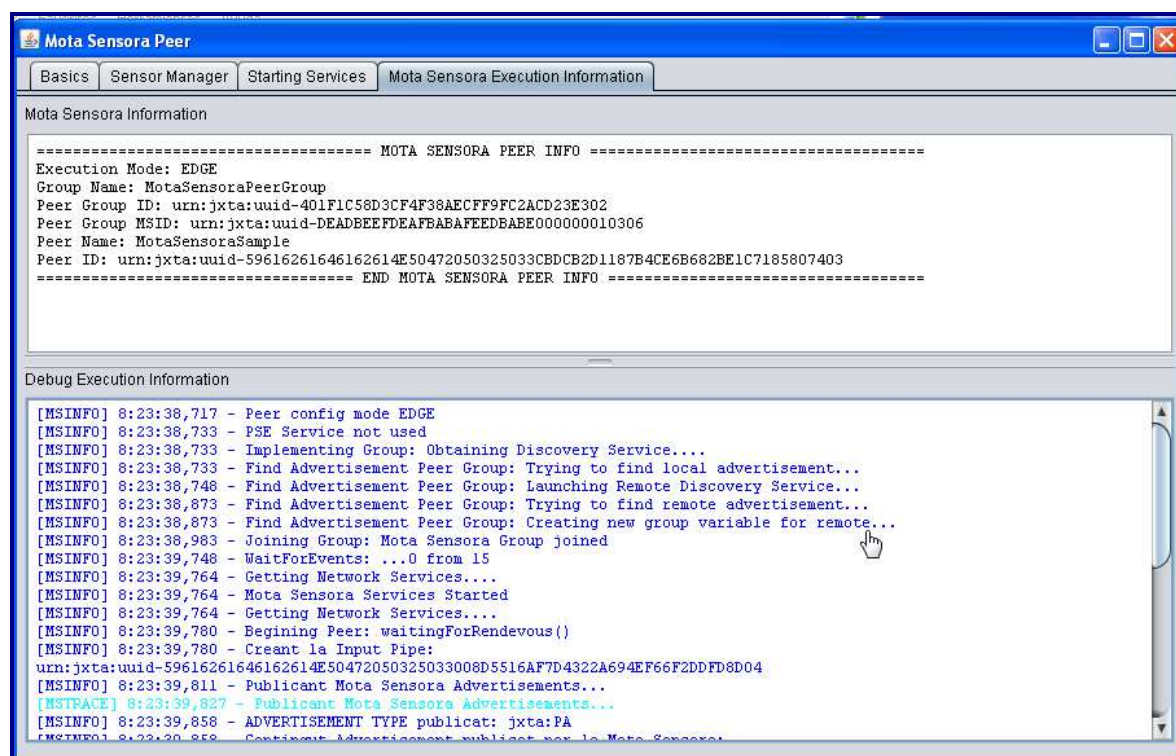


Figura 33: Funcionament GUI – Pestanya Mota Sensora Execution Information

### 6.1.2. Creació d'una mota sensora

La creació de la mota sensora es fa a través del GUI. La pestanya bàsics ens permet informar dades referents a la mota sensora, les seves característiques de funcionament i el comportament que tindrà dintre de la xarxa JXTA.

Dades de la mota sensora: Es pot informar el seu nom i donar-li una descripció. Aquestes dades s'inclouran en l'advertiment que la mota sensora generarà a la xarxa per anunciar la seva presència i alguna de les seves característiques com el nom, la descripció, el Peer ID (com a peer que és) i el Group Peer al que pertany la mota sensora.

Característiques de funcionament: La opció "Debug Execution" ens permet realitzar el seguiment de l'execució de la mota sensora a través de la pestanya "Mota Sensora Execution Information".

Comportament a la xarxa JXTA: Especifica quin ha de ser el comportament de la mota sensora com a Peer de la xarxa JXTA com és. En el nostre cas la mota sensora es comportarà com a EDGE dintre del sistema.

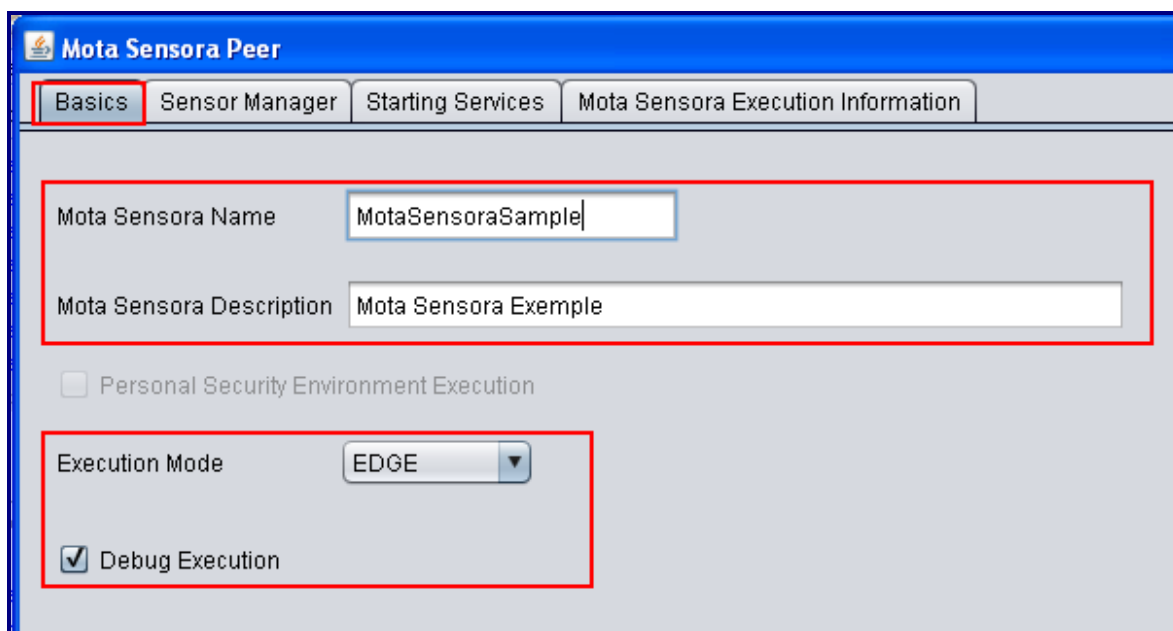


Figura 34: Creació d'una mota sensora – Pestanya Bàsics

Una vegada definides les dades de la mota sensora podem procedir a especificar els sensors de què es compona la mota sensora. La pestanya "Sensor Manager" de la interfície gràfica ens permet aquesta gestió.

El funcionament de la pestanya "Sensor Manager" es basa en una llista on es poden afegir i esborrar sensors. Aquesta part de la interfície gràfica es divideix en tres parts: definició de les dades d'un sensor, controls d'afegir i esborrar sensors i llista de sensors.

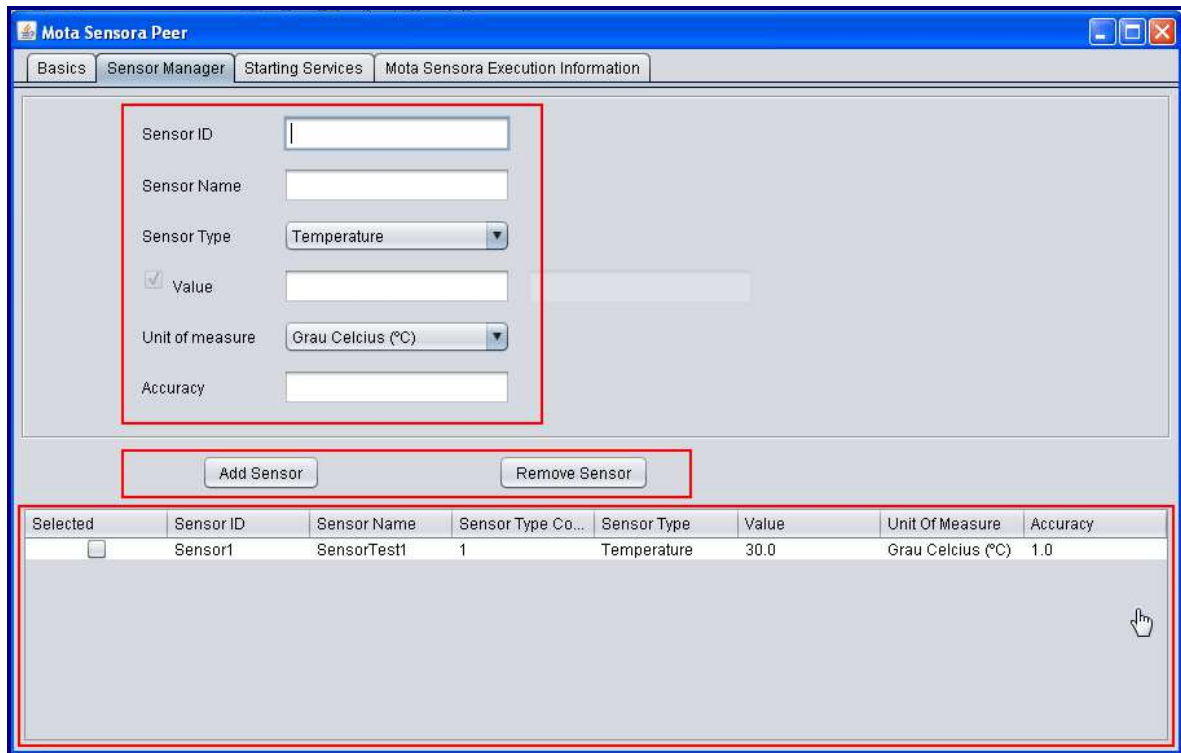


Figura 35: Creació d'una mota sensora – Pestanya Sensor Manager

Definició de les dades d'un sensor: Ens permet informar les dades del sensor que són necessàries per al funcionament de la mota sensora. Les dades que podem informar al sensor són les següents:

**Sensor ID:** Ens permet identificar de forma única un sensor.

**Sensor Name:** Ens permet especificar el nom del sensor.

**Sensor Type:** Ens permet especificar de quin tipus és el sensor. El sistema té definits els següents tipus de sensors: temperatura, posició, acústic, lluminositat, presència i pressió.

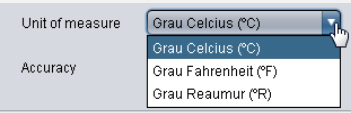
**Value:** Valor del sensor a la seva posada en marxa.

**Unit of measure:** Unitat de mesura en què el sensor està prenent la magnitud.

**Accuracy:** Permet especificar la precisió en què el sensor ha de prendre les dades. En el funcionament d'un sensor entenem que la precisió, a més de les característiques físiques del sensor, anirà relacionada amb la freqüència de

lectura dels sensors. Aquest fet condicionarà el consum de la mota sensora i, en casos en què una mota sensora funcioni per bateries, aquest fet s'ha de tenir en compte.

En aquest punt cal aclarir que depenen del tipus de sensor s'especificaran unes dades o unes altres, i que el valor de l'unitat de mesura canviarà d'acord al tipus de sensor. D'acord al tipus de sensor la definició de dades es presenten de la següent forma:

Tipus Sensor	Interfície definició de dades	
Temperatura	Sensor ID <input type="text"/> Sensor Name <input type="text"/> Sensor Type <input type="text" value="Temperature"/> <input checked="" type="checkbox"/> Value <input type="text"/> Unit of measure <input type="text" value="Grau Celcius (°C)"/> Accuracy <input type="text"/>	
Posició	Sensor ID <input type="text"/> Sensor Name <input type="text"/> Sensor Type <input type="text" value="Position"/> <input checked="" type="checkbox"/> Value <input type="text"/> <input type="text"/> Unit of measure <input type="text"/> Accuracy <input type="text"/>	
Acústic	Sensor ID <input type="text"/> Sensor Name <input type="text"/> Sensor Type <input type="text" value="Acoustical"/> <input checked="" type="checkbox"/> Value <input type="text"/> Unit of measure <input type="text" value="Decibels(dB)"/> Accuracy <input type="text"/>	Només conté la opció Decibels

Lluminositat	Sensor ID <input type="text"/> Sensor Name <input type="text"/> Sensor Type <input type="text" value="Lightness"/> <input checked="" type="checkbox"/> Value <input type="text"/> Unit of measure <input type="text" value="Energia lluminica (lm-s)"/> Accuracy <input type="text"/>	Unit of measure <input type="text" value="Energia lluminica (lm-s)"/> Accuracy <input type="text"/> Add Sens <ul style="list-style-type: none"> <li>Energia lluminica (lm-s)</li> <li>Energia lluminica (lm-s)</li> <li>Fluxe lluminós (lm)</li> <li>Intensitat lluminosa (cd)</li> <li>Lluminància (cd/m2)</li> <li>lluminància (lx)</li> </ul>
Pressió	Sensor ID <input type="text"/> Sensor Name <input type="text"/> Sensor Type <input type="text" value="Pressure"/> <input checked="" type="checkbox"/> Value <input type="text"/> Unit of measure <input type="text" value="Pascal (Pa)"/> Accuracy <input type="text"/>	Unit of measure <input type="text" value="Pascal (Pa)"/> Accuracy <input type="text"/> <ul style="list-style-type: none"> <li>Pascal (Pa)</li> <li>Pascal (Pa)</li> <li>Bar (bar)</li> <li>Atmosfera (atm)</li> </ul>
Presència	Sensor ID <input type="text"/> Sensor Name <input type="text"/> Sensor Type <input type="text" value="Presence"/> <input checked="" type="checkbox"/> Value <input type="text"/> Unit of measure <input type="text"/> Accuracy <input type="text"/>	El valor serà present o no present

Taula 23: Interfícies de definició de les dades d'un sensor

Controls d'afegir i esborrar sensors: Aquestes accions es realitzen a través dels botons. Es tracta d'afegir un sensor quan estan especificades les seves dades o esborrar-lo en el cas necessari. Si s'ha d'esborrar un sensor, primer cal marcar el camp "Selected" del sensor que es vol esborrar.

Llista de sensors: És la llista amb els sensors que s'han creat per a la mota sensora.



### 6.1.3. Posada en marxa de la mota sensora

La posada en marxa de la mota sensora es realitza a través de la pestanya "Starting Services". El botó "Start Mota Sensora" s'habilitarà un cop la mota sensora estigui definida amb els seus sensors.

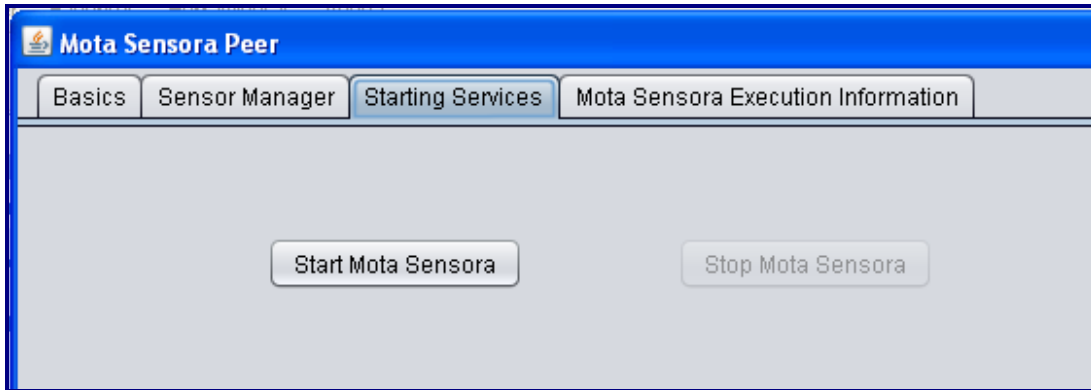


Figura 36: Posada en marxa d'una mota sensora – Pestanya Starting Services

### 6.1.4. Seguiment de l'execució d'una mota sensora

Si s'ha habilitat la opció "Debug Execution" de la pestanya Basics és possible fer el seguiment de l'execució a través de la pestanya "Mota Sensora Execution Information". Aquesta pestanya es divideix en dues parts:

Mota Sensora Information: Mostra la informació de la mota sensora com a Peer: el seu nom, identificador dins la xarxa JXTA, identificador de Peer Group, el nom del Peer Group, l'identificador del Module Specification i el mode d'execució.

Debug Execution Information: Mostra la informació del què s'està realitzant durant l'execució de la mota sensora. Els missatges que es mostren s'han definit de diferents colors a fi de poder fer un seguiment visual més fàcil de l'execució.

Es mostraran els següents tipus de missatges:

**MSINFO**: Són missatges informatius d'una acció realitzada per la mota sensora. El color del missatge és blau.

**MSERROR**: Missatge que informa d'un error durant l'execució de la mota sensora. El color del missatge és el vermell.

**MSWARNING**: Es produeix un avís durant l'execució de la mota sensora. El color del missatge és el magenta.

MSTRACE: Missatge de seguiment específic. S'utilitza quan es vol seguir alguna part de l'execució en especial. El color assignat ha estat el cyan.

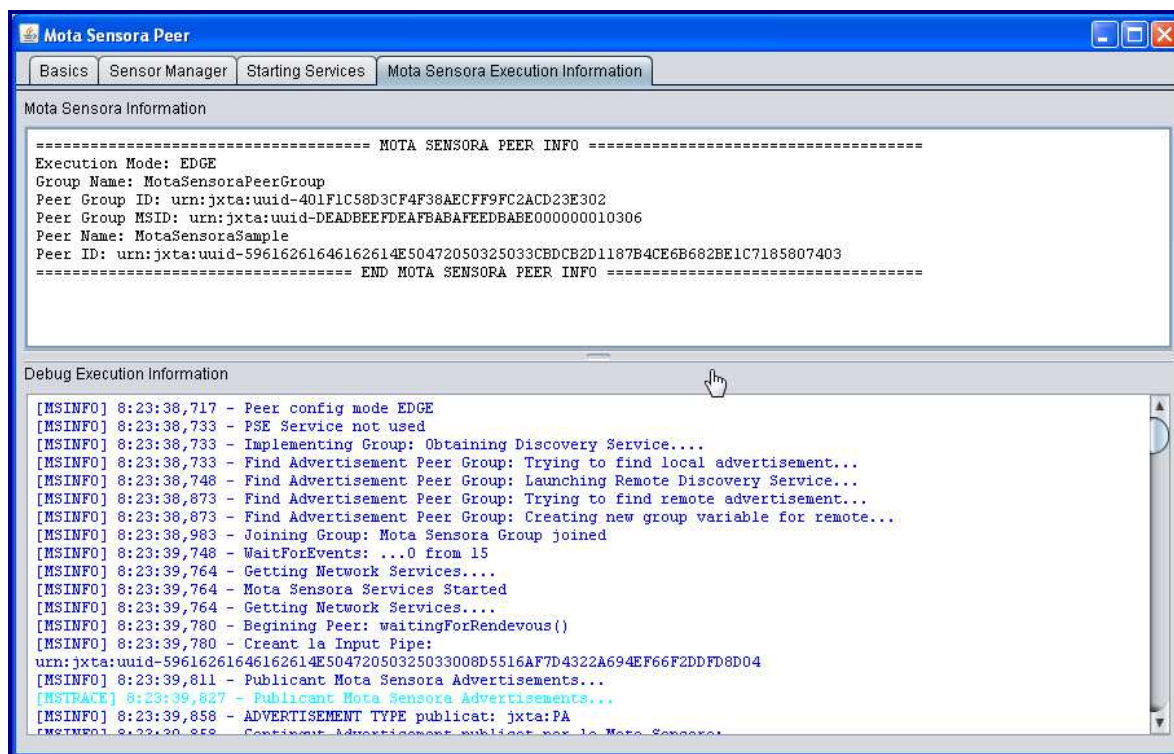


Figura 37: Seguiment execució d'una mota sensora – Pestanya Mota Sensora Execution Information

En el cas que l'execució es realitzi sense utilitzar el GUI de l'aplicació és possible el mateix seguiment a través de la sortida de l'IDE tal i com es pot observar a la següent figura.

```

Salida - MotaSensoraPeer_Pck (run)
run:
[MSINFO] 8:23:38,717 - Peer config mode EDGE
[MSINFO] 8:23:38,733 - PSE Service not used
[MSINFO] 8:23:38,733 - Implementing Group: Obtaining Discovery Service....
[MSINFO] 8:23:38,733 - Find Advertisement Peer Group: Trying to find local advertisement...
[MSINFO] 8:23:38,748 - Find Advertisement Peer Group: Launching Remote Discovery Service...
[MSINFO] 8:23:38,873 - Find Advertisement Peer Group: Trying to find remote advertisement...
[MSINFO] 8:23:38,873 - Find Advertisement Peer Group: Creating new group variable for remote...
[MSINFO] 8:23:38,983 - Joining Group: Mota Sensora Group joined
[MSINFO] 8:23:39,748 - WaitForEvents: ...0 from 15
[MSINFO] 8:23:39,764 - Getting Network Services....

[MSINFO] 8:23:39,764 - Mota Sensora Services Started
[MSINFO] 8:23:39,764 - Getting Network Services....

***** JXTA PEER INFO *****
Execution Mode: EDGE
Group Name: MotaSensoraPeerGroup
Peer Group ID: urn:jxta:uuid-401f1c58d3cf4f38a8cfff9fc2acd23e302
Peer Group MSID: urn:jxta:uuid-DEADBEFFDEAFBABAFFREDBABE00000010306
Peer Name: MotaSensoraSample
Peer ID: urn:jxta:uuid-59616261646162614E50472050325033CBDCB2D1187B4CE6B682BEE1C7185807403
***** END JXTA PEER INFO *****
[MSINFO] 8:23:39,780 - Begining Peer: waitingForRendevous()
[MSINFO] 8:23:39,780 - Creant la Input Pipe: urn:jxta:uuid-59616261646162614E50472050325033008D5516AF7D4322A694EF66F2DDFD8D04
[MSINFO] 8:23:39,811 - Publicant Mota Sensora Advertisements...
[MSTRACE]8:23:39,827 - Publicant Mota Sensora Advertisements...

```

Figura 38: Seguiment execució d'una mota sensora – Pestanya Mota Sensora Execution Information

### 6.1.5. Aturada de la mota sensora

L'aturada d'una mota sensora es realitza per mitjà del botó "Stop Mota Sensora" de la pestanya "Starting Services". Aquesta acció atura el funcionament de la mota sensora i la seva connexió a la xarxa JXTA.



Figura 39: Inici del servei d'una mota sensora – Pestanya Starting Services

## 6.2. Aplicació de Monitorització i control de Mota Sensora

### 6.2.1. Inici de l'aplicació

L'aplicació es pot iniciar des de l'IDE Netbeans o a través de l'emulador d'Android que conté el SDK Android. L'aplicació "Mota Sensora Manager" es pot trobar junt a la resta d'aplicacions de l'emulador d'Android.

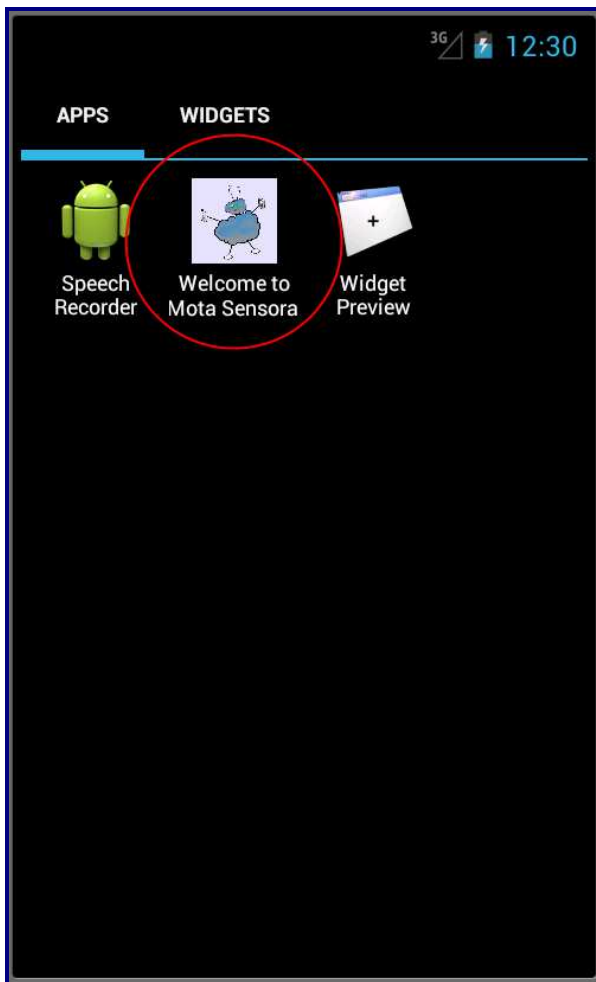


Figura 40: Localització de l'aplicació de Monitorització i control al menú principal Android

### 6.2.2. Pantalla principal de l'aplicació

Una vegada iniciada l'aplicació es presenta una pantalla de benvinguda.

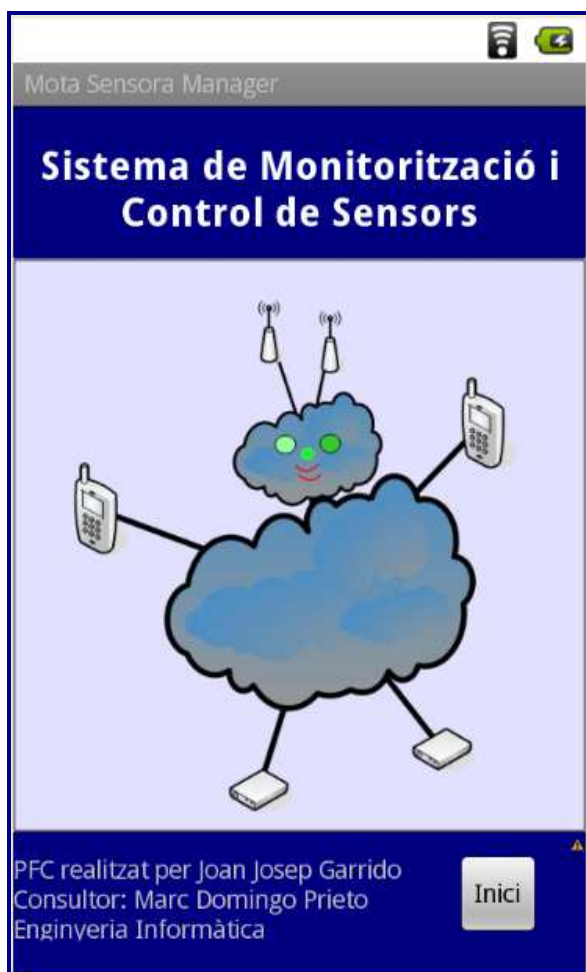


Figura 41: Aplicació Monitorització – Pantalla de benvinguda

A través del botó "Inici" podem entrar a la pantalla de connexió de l'aplicació.

### 6.2.3. Connexió a la xarxa de l'aplicació

La interfície de connexió de l'aplicació de Monitorització ens permet la parametrització de l'aplicació com a Proxied Peer per realitzar la connexió amb el Relay Peer que correspongui. Les dades s'informen automàticament amb les valors per defecte de l'aplicació, però es poden modificar en cas de que sigui necessari.

Les dades que calen són les següents:

Proxied Peer Name: Nom que se li vol donar al Proxied Peer a la xarxa JXTA.

Peer Group Name: Nom del Peer Group al que s'afegirà l'aplicació.

Relay Proxy Peer Host: Adreça IP del Relay Peer amb a través del qual volem realitzar la connexió a la xarxa JXTA.

Relay Proxy Peer Port: Port IP que el Relay Peer escolta i podem fer la connexió.

Una vegada s'ha parametritzat l'aplicació es pot continuar la connexió a través del botó "Connect".

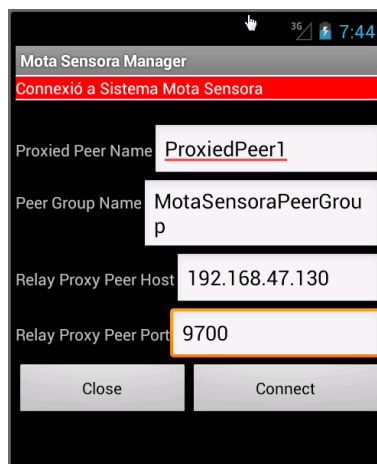


Figura 42: Aplicació Monitorització – Pantalla de configuració de la connexió

Un cop iniciada la connexió es pot seguir a través de la pantalla de log que es presenta a la següent figura:

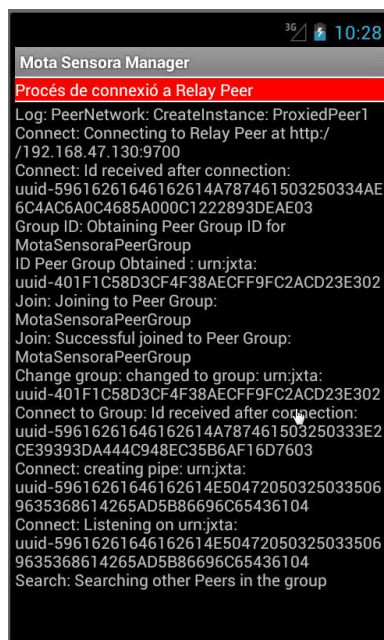


Figura 43: Aplicació Monitorització – Seguiment Connexió

#### 6.2.4. Cerca de les motes sensores disponibles

Durant l'establiment de la connexió es llença una cerca de les motes sensores disponibles al Peer Group de Motes Sensores de la xarxa JXTA. Aquesta cerca es retorna en una llista de motes sensores trobades on podrem seleccionar aquella mota sensora a la qual vulguem monitoritzar o controlar.



Es pot repetir la cerca prenent el botó "Search Peers", el qual envia una petició de cerca al Relay Peer a fi que aquest retorni aquelles motes sensores disponibles.

Figura 44: Aplicació Monitorització – Seguiment Connexió

#### 6.2.5. Monitorització i control d'una mota sensora

Quan s'ha seleccionat una mota sensora de la llista de motes sensores, es presenta la interfície de Monitorització de la Mota Sensora. Aquesta interfície únicament permet la visualització de dades d'una mota sensora i la llista de sensors que aquesta conté. Els sensors de la llista es poden identificar pel seu "Sensor Id" i és possible conèixer el tipus de sensor de què es tracta.

A través del botó "Control" es pot accedir a la interfície de control de la mota sensora.



Figura 45: Aplicació Monitorització – Monitorització de les dades de la mota sensora

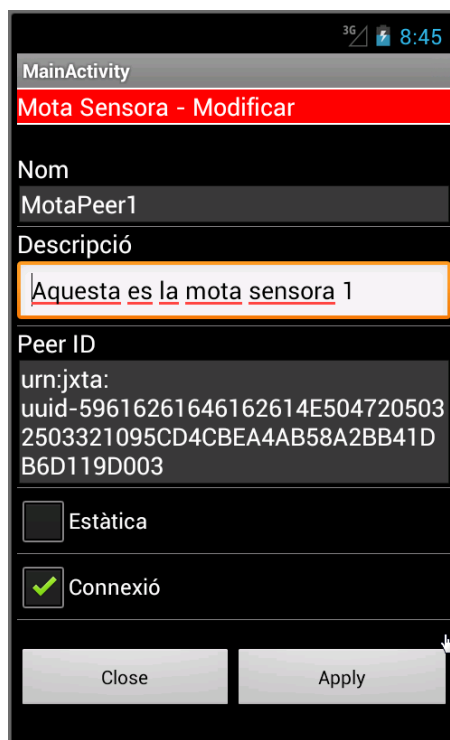


Figura 46: Aplicació Monitorització – Control de les dades de la mota sensora

La interfície de Control de la Mota Sensora ens permet la modificació de les dades d'aquesta. Una vegada s'han modificat, s'ha de prémer el botó "Apply" que generarà els missatges necessaris per a l'actualització de les dades a la mota sensora.

A la interfície de monitorització de la mota sensora es presenta la llista dels sensors que conté. Seleccionant un d'aquests sensors, es generà el missatge de petició a fi d'obtenir les dades d'un sensor i visualitzar-lo a la seva interfície.



### 6.2.6. Monitorització i control d'un sensor

Quan s'ha seleccionat un sensor a la llista de sensors de la mota sensora aquest es mostra a la interfície gràfica per a la seva monitorització. Les dades mostrades, de la mateixa manera que a la interfície de monitorització de la mota sensora, no són modificables i únicament es permet la seva visualització. A través del botó "Control" de la interfície es pot accedir a la interfície de control del sensor.



Figura 47: Aplicació Monitorització – Monitorització de les dades del sensor



Figura 48: Aplicació Monitorització – Control de les dades del sensor

La interfície de control del sensor ens permet modificar dades referents al sensor. Una vegada modificades, a través del botó "Apply", es poden generar les peticions de modificació.

### 6.3. Funcionament trivial de l'aplicació

El funcionament trivial de l'aplicació són aquells elements mínims que calen pel funcionament bàsic de l'aplicació. La següent figura mostra un esquema de l'arquitectura mínima de funcionament:

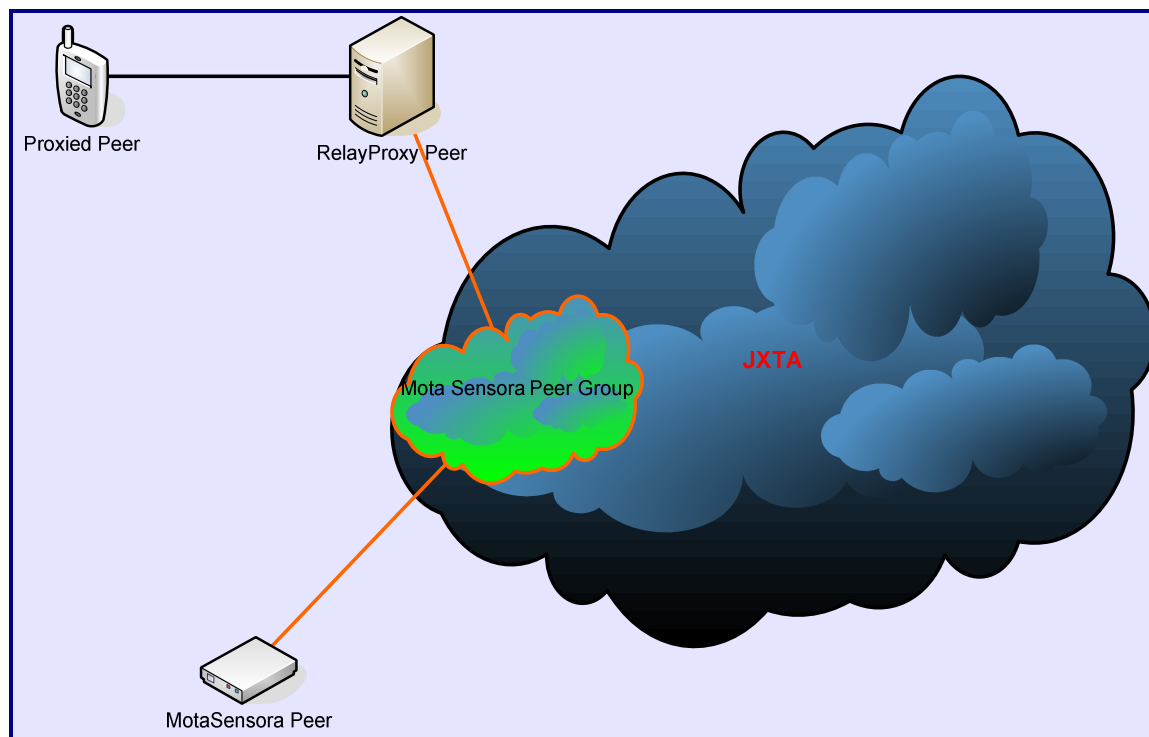


Figura 49: Arquitectura trivial de funcionament de l'aplicació

Proxied Peer: Aplicació en Android. Aquesta es connecta a la xarxa a través d'un Relay Peer.

RelayProxy Peer: Es tracta d'un peer de JXTA configurat a través del JXTA configurador a fi de fer les funcions de Relay i Proxy requerides pel Proxied Peer per tal de facilitar la connexió amb la Mota Sensora Peer.

Per tant, aquest RelayProxy Peer es pot arrencar des del mateix jxta-shell i afegir-lo per la línia de comandes al grup en què es realitzarà la comunicació a l'arquitectura. La configuració del jxta configurador s'especifica posteriorment.

MotaSensora Peer: Aquesta és la part implementada per a la gestió dels sensors de la mota sensora. Es tracta d'un peer que conté un conjunt de sensors. Aquests peers permeten donar informació (monitoratge) i modificar el seu estat (control). La MotaSensora Peer funciona en mode d'execució EDGE.

### 6.3.1. Configuració i execució del Relay Peer

La configuració i execució del Relay Peer es realitza configurant un peer amb el JXTA configurator de la següent forma:

- Executem el jxta-shell utilitzant per exemple l'Ant. Un cop s'obre la interfície gràfica de JXTA Configurator omplim les dades bàsiques: Peer Name i una contrasenya.

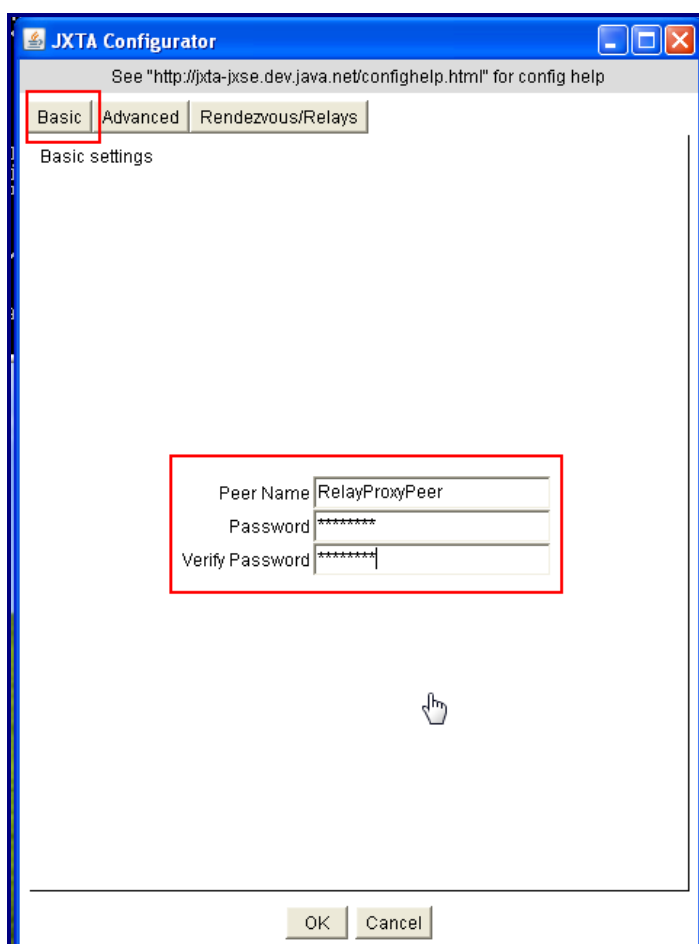


Figura 50: Configuració Relay Peer – Basic settings

- Prement el botó Advanced, podem configurar algunes opcions avançades del peer que estem creant. Aquesta pantalla s'ha de configurar com s'adjunta a la següent figura:

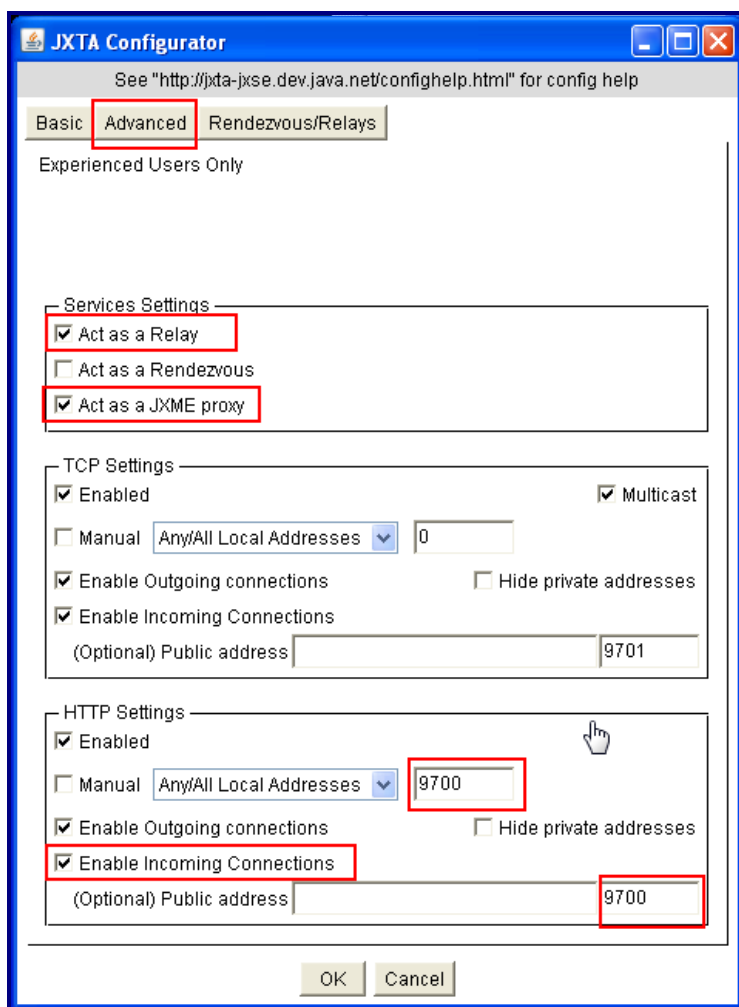


Figura 51: Configuració Relay Peer – Advanced settings

Es important indicar que el Peer s'ha de comportar també com a Proxy, ja que ha de ser qui permeti la connexió al Proxied Peer.

- Finalment a través del botó "Rendezvous/Relays" especifiquem les opcions tal i com indica la següent figura:

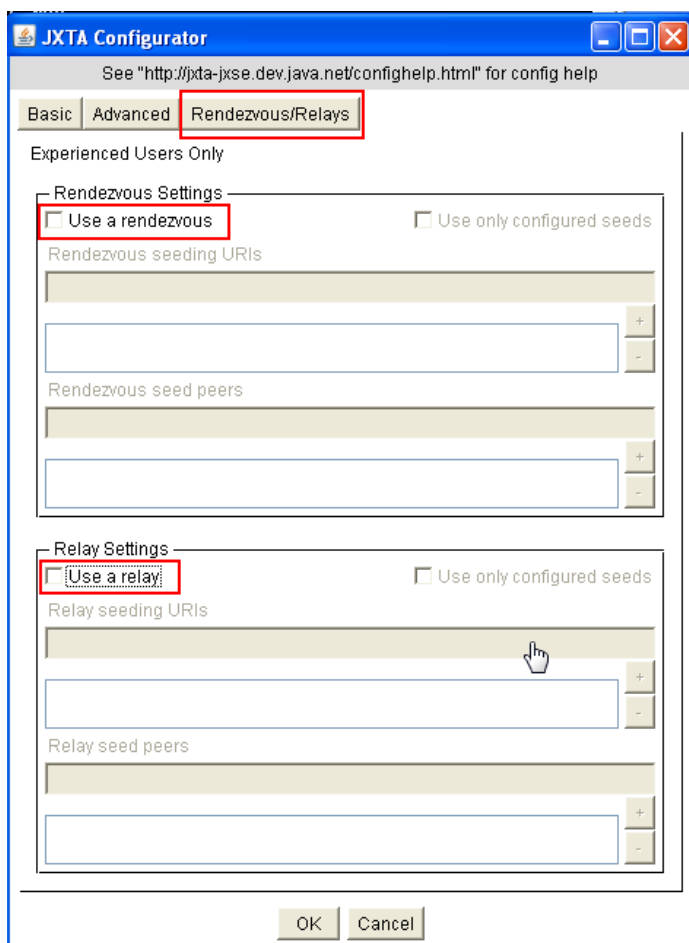


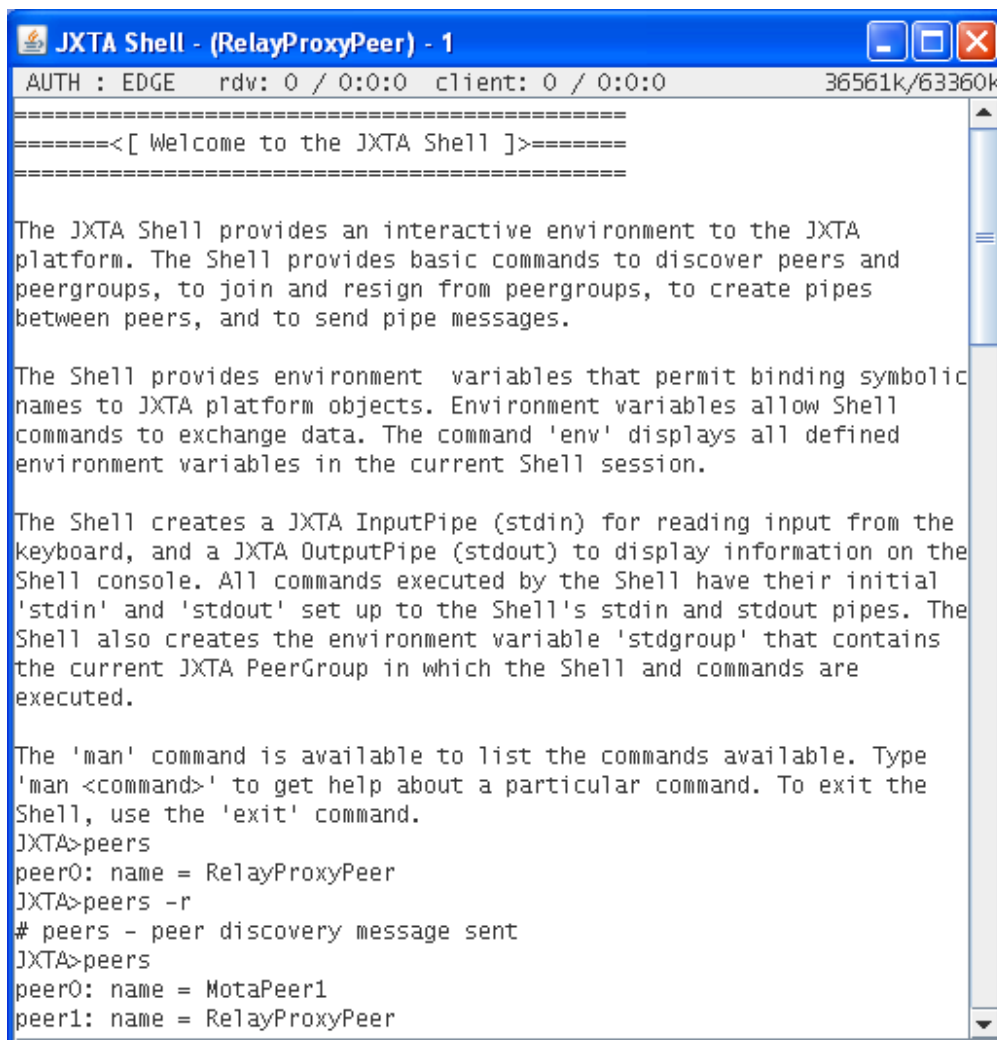
Figura 52: Configuració Relay Peer - Rendezvous/Relays settings

- Un cop feta la configuració ens demana introduir la contrasenya, que hem especificat anteriorment i premem el botó OK:



Figura 53: Iniciar Relay Peer

- Una vegada validat ens permet accedir al shell de JXTA en el peer que fa de Relay Peer.



```
JXTA Shell - (RelayProxyPeer) - 1
AUTH : EDGE   rdv: 0 / 0:0:0   client: 0 / 0:0:0   36561k/63360k
=====
=====<[ Welcome to the JXTA Shell ]>=====
=====
The JXTA Shell provides an interactive environment to the JXTA
platform. The Shell provides basic commands to discover peers and
peergroups, to join and resign from peergroups, to create pipes
between peers, and to send pipe messages.

The Shell provides environment variables that permit binding symbolic
names to JXTA platform objects. Environment variables allow Shell
commands to exchange data. The command 'env' displays all defined
environment variables in the current Shell session.

The Shell creates a JXTA InputPipe (stdin) for reading input from the
keyboard, and a JXTA OutputPipe (stdout) to display information on the
Shell console. All commands executed by the Shell have their initial
'stdin' and 'stdout' set up to the Shell's stdin and stdout pipes. The
Shell also creates the environment variable 'stdgroup' that contains
the current JXTA PeerGroup in which the Shell and commands are
executed.

The 'man' command is available to list the commands available. Type
'man <command>' to get help about a particular command. To exit the
Shell, use the 'exit' command.
JXTA>peers
peer0: name = RelayProxyPeer
JXTA>peers -r
# peers - peer discovery message sent
JXTA>peers
peer0: name = MotaPeer1
peer1: name = RelayProxyPeer
```

Figura 54: Relay Peer - Shell de JXTA

## 7. Canvis i millores de futur del sistema

### Gestió acotada geogràficament:

Un possible treball futur sobre el sistema es basaria en l'ampliació dels protocols existents a fi que permetin conèixer dades referents als sensors que es troben en una àrea. Per exemple:

- getDataSensorValueAtPosition: Aquest missatge permet conèixer una determinada dada del sensor més proper a la posició indicada
- getMaxSensorTemperatureAtArea: Aquest missatge permetria conèixer la temperatura màxima a un àrea delimitada.

### Implementació d'un nivell de seguretat:

El joc de comandes que implementa un Proxied Peer per connectar a un Relay Proxy Peer és molt simple. Com a conseqüència d'això quan un Proxied Peer vol unir-se a un grup de peers, aquesta connexió no permet passar dades referents a la seguretat.

Una millora del projecte es basaria en l'extensió del joc de comandes de connexió entre Proxied Peer i Relay Proxy Peer a fi que permetés passar paràmetres relacionats amb la seguretat com per exemple una contrasenya o un certificat. Així mateix es podria fer que el Relay Proxy Peer generés un certificat que es passés al Proxied Peer quan es connectés per primera vegada, i que el fes servir automàticament per les comandes de join. Qualsevol de les opcions porta associada la modificació del codi del Relay Proxy Peer, el qual té una complexitat important a l'estar entrellaçat amb tot el codi de JXTA.

### Implementació de l'advertiment de motes sensores:

Durant el disseny de l'aplicació es va decidir la creació d'un advertiment personalitzat que seria intercanviat per les diferents motes sensores del sistema. L'advertiment dissenyat tenia com a objectiu informar de les característiques d'una determinada mota sensora: nom, descripció, si es tracta d'una mota sensora estàtica, els diferents sensors que la componen i característiques dels diferents sensors a més de l'advertiment de Pipe inclòs en el seu contingut.

D'aquesta forma es pretenia que el tràfic de missatges entre una mota sensora i els Proxied Peers (l'aplicació d'Android en el nostre projecte) fos el mínim possible ja que, com sabem, els Proxied Peers per les seves característiques són dispositius de capacitat limitada. Amb la optimització del tràfic entre Mota Sensora i Proxied Peer s'aconseguia una millora del conjunt.

El Proxied Peer necessita de l'existència d'un Relay Proxy Peer per a la seva connexió amb la xarxa JXTA. El Relay Proxy Peer com a "super-peer", entre d'altres, realitza la traducció dels advertiments que van dirigits al Proxied Peer de format XML a format HTTP més lleuger i que el Proxied Peer té la capacitat de processar. El resultat ha estat que el Relay Proxy Peer en realitat no ha estat capaç de traduir l'advertiment personalitzat creat per les motes sensores en el format necessari perquè el Proxied Peer el pogués processar.

Davant aquest escenari es presenten diferents millores. D'una banda la modificació del Relay Proxy Peer a fi que pugui realitzar la traducció de l'advertiment personalitzat. Aquesta modificació portaria com a conseqüència una menor portabilitat de l'aplicació, ja que no serviria qualsevol Relay Proxy Peer de la xarxa JXTA per mantenir el sistema de gestió i control de motes sensores. Al mateix temps seria una solució més òptima pel que es refereix a intercanvi de dades entre els diferents components del sistema proposat.

D'altra banda s'obre l'opció de crear una aplicació en Android sobre una versió completa de JXTA i no sobre la versió JXME-Proxied que és limitada en si mateixa. Aquesta millora requeriria de dispositius amb més capacitat.



## 8. Conclusions

En aquest projecte s'ha implementat un sistema de monitorització i control de Motes Sensores. Tot i que el projecte s'emmarca en l'àrea de Xarxes de Computadors, s'han hagut de treballar molts altres conceptes apresos durant els estudis d'Enginyeria Informàtica. El projecte brinda l'oportunitat de posar en joc tot allò que s'ha anat treballant durant aquests estudis. Permet aplicar els coneixements adquirits en un context pràctic on la planificació i metodologia utilitzada és de vital importància per aconseguir un resultat exitós.

Durant l'elaboració del projecte s'ha pogut observar com en ocasions no es pot abastar allò que s'ha definit en un principi. Sovint, una de les raons és que les eines que s'han de fer servir no ho permeten realitzar amb facilitat i sense comprometre l'èxit del projecte. Com s'ha pogut veure, un exemple és el fet de que JXME Proxied no implementa seguretat amb facilitat. Aquest fet ha condicionat que en el producte final s'hagi hagut de prescindir d'aquest nivell de seguretat per no complicar en excés el projecte i les conseqüències que aquest fet tindrien sobre la planificació realitzada.

En ocasions el fet de no voler restringir la portabilitat de la solució creant estructures específiques comporta haver de canviar alguns trets que s'havien definit durant el disseny. En el cas d'aquest projecte, s'ha pogut comprovar com es comporta el Relay Peer a l'hora de fer la traducció dels advertisements al Proxied Peer i de quina forma fa el resum d'aquests missatges. Adequar el funcionament del Relay Peer al funcionament desitjat, que seria que el Relay Peer realitzés la traducció del nostre advertisement personalitzat, tindria conseqüències pel nostre sistema ja que ens seria necessari tenir una xarxa de Relay Peers personalitzats distribuïts arreu.

D'altra banda, s'ha pogut treballar el P2P i s'ha pogut veure quines avantatges té una solució de parells iguals per a un sistema de les característiques del que es presenta en aquest projecte. A més, l'arquitectura del sistema creat en aquest projecte ens mostra quantes possibilitats existeixen per poder realitzar

aplicacions que controlin sistemes complexos remotament sense necessitat de disposar de dispositius molt avançats o d'altres capacitats.

Des d'un punt de vista personal ha estat un repte molt encoratjador, un treball dur que m'ha donat la oportunitat d'explorar noves tecnologies que són la base del que avui en dia està succeint en el sector de desenvolupament de software i que m'ajuda a consolidar molts coneixements adquirits durant els estudis de l'Enginyeria.

## 9. Glossari

Advertisement: Un advertisement o anunci és un document XML que descriu tots els recursos de la xarxa JXTA (peers, grups, pipes, serveis, etc.). La comunicació a JXTA pot ser interpretada com un intercanvi d'advertiments.

Android: És un sistema operatiu per dispositius mòbils el qual es va estendre posteriorment a dispositius tipus tablet, netbook, reproductors MP3,... Les aplicacions per aquest sistema operatiu estan escrites en Java i s'executen sobre un framework Java que inclou les seves llibreries.

CDC: Sigles que corresponen a Connected Device Configuration

CLDC: Sigles que corresponen a Connected Limited Device Configuration

Comunicació peer-to-peer (P2P): La comunicació P2P es basa en una xarxa d'elements iguals. A diferència del paradigma client-servidor, on els diferents clients requereixen del servidor que els proveeix del servei que necessiten, la comunicació P2P es basa en elements iguals on tots es comporten com a client i com a servidor. Aquesta arquitectura té com a principal avantatge la distribució dels serveis en els diferents nodes de la xarxa, minimitzant la criticitat que es produeix en les arquitectures client-servidor quan cau l'element que ofereix el servei.

HTTP (Hypertext Transfer Protocol): És el protocol utilitzat en les transaccions de la World Wide Web. En aquest projecte és utilitzat per a la comunicació entre el Relay Peer i el Proxied Peer. El Relay Peer rep els missatges en format XML i els tradueix a HTTP que és interpretable pel Proxied Peer.

IDE (Integrated Development Environment): Es tracta d'un entorn compost per un conjunt d'eines de programació que es pot dedicar a un o varis llenguatges de programació.

JXTA (Juxtapose): Es tracta d'un conjunt de protocols que permeten la creació i desenvolupament de xarxes P2P. Amb l'ús dels protocols JXTA els nodes poden publicar recursos i trobar i consumir d'altres remots independentment de la capa de transport i del llenguatge d'implementació. Aquest protocol està concebut perquè una gran quantitat de dispositius des d'ordinadors a mòbils o PDA's, puguin mantenir una comunicació de forma descentralitzada.

JXTA Micro Edition (JXME-Proxied): Es tracta d'una plataforma que proporciona funcionalitats compatibles amb JXTA a dispositius de poca capacitat com a mòbils o PDA's. Amb aquesta plataforma, dispositius que no tindrien possibilitat de comunicar-se a través d'una xarxa JXTA (Proxied Peer) poden fer-ho a través d'un node repetidor (Relay Peer).

Mota sensora: Es tracta d'un dispositiu que està compost de diferents tipus de sensors. Aquest dispositiu permet rebre informació de diferents magnituds, té capacitat de procés i ofereix els serveis necessaris per accedir a la plataforma JXTA que dona base al sistema de monitorització i control.

Peer: Es tracta de la unitat de programari d'una xarxa P2P. Aquesta es comunica amb altres peers dintre de la xarxa que formen. Els peers interactuen entre ells mateixos per oferir serveis al conjunt del sistema.

Pipe: Es tracta del canal de comunicació utilitzat pels peers de JXTA per a l'intercanvi de missatges. JXTA defineix tres tipus de pipes: Unicast, Unicast Secure i Propagate.

Proxied Peer: Es tracta del tipus més elemental de peer. Ha estat desenvolupat per ser utilitzat per a dispositius mòbils amb poca capacitat de procés i recursos molt limitats. La comunicació dels Proxied Peers amb la xarxa JXTA es fa a través del Relay Proxy Peer que és qui representa al Proxied Peer a la xarxa.

Relay Peer: També anomenats super-peers, és tracta d'un peer normal amb característiques avançades que permeten que actiu com a Proxy per donar accés

a la plataforma JXTA a altres peers que per les seves característiques no poden establir connexió directa als seus recursos oferts.

Sensor: És un dispositiu dissenyat per rebre informació d'una magnitud de l'exterior i transformar-la en una altra magnitud, normalment elèctrica, que podem quantificar y manipular.

XML (Extensible Markup Language): És un llenguatge de marques que ha esdevingut un estàndard per a l'intercanvi d'informació estructurat entre diferents plataformes. JXTA utilitza XML com a llenguatge d'intercanvi de missatges i advertisements.

## 10. Bibliografia

### 10.1. Referències a JXTA

DINANZO, MARIANO. (2006). "Redes Peer to Peer y Tecnología JXTA" [en línia], <http://www.dsi.fceia.unr.edu.ar/downloads/distribuidos/material/monografias/P2P-JXTA.pdf>, [data de consulta: 29 de Setembre, 2011].

SEE, Dr. SIMON. "Project JXTA Technology Overview" [en línia], <http://www.apan.net/meetings/busan03/materials/cs/p2p/jxta-overview.pdf>, [data de consulta: 22 d'Octubre, 2011]

BROOKSHIER, DANIEL; GOVONI, DARREN; KRISHNAN, NAVANEETH; SOTO, JUAN CARLOS. (2002) "JXTA: Java P2P Programming". Indianapolis: Sams Publishing.

WILSON, BRENDON. (2002). "JXTA". Indianapolis: New Riders Publishing.

GRADECKI, JOSEPH D. (2002). "Mastering JXTA". Indianapolis: Wiley Publishing, Inc.

SUN MICROSYSTEMS. (2007). " JXTA Java Standard Edition v2.5: Programmers Guide" [en línia], [http://jxse.kenai.com/Tutorials/JXSE\\_ProgGuide\\_v2.5.pdf](http://jxse.kenai.com/Tutorials/JXSE_ProgGuide_v2.5.pdf)

Wikipedia. (2012). "JXTA" [en línia] <http://es.wikipedia.org/wiki/JXTA>, [data de consulta: 02 de Desembre, 2012]

"JXTA Java 2.4" [en línia], <http://www.ppgia.pucpr.br/servicos/techdocs/jxtadocs/api/overview-summary.html>

"JXSE Wiki Home Page" [en línia], <http://kenai.com/projects/jxse/pages/Home>

## 10.2. Referències a JXME

DOMINGO-PRIETO, MARC; ARNEDO-MORENO, JOAN; HERRERA-JOANCOMARTÍ, JORDI; PRIETO-BLAZQUEZ, JOSEP. (2012). "Towards secure mobile P2P applications using JXME" [en línia], <http://openaccess.uoc.edu/webapps/o2/handle/10609/13681>, [data de consulta: 02 d'Octubre, 2012]

DOMINGO-PRIETO, MARC; ARNEDO-MORENO, JOAN; HERRERA-JOANCOMARTÍ, JORDI. (2010). "JXTA security in mobile constrained devices", [en línia], <http://isyoinfo.info/jisis/vol2/no12/jisis-2012-vol2-no12-01.pdf>

Martínez-Pabón, Francisco-Orlando; Camacho-Gómez, Ricardo-Alberto; García-Martínez, Luis-Ernesto; Caicedo-Rendón, Oscar Mauricio; Hurtado-Guaca, Javier-Alexander. (2006). "JXME: Una plataforma robusta para el desarrollo de aplicaciones P2P en dispositivos móviles" [en línia], [http://ciruelo.uninorte.edu.co/pdf/ingenieria\\_desarrollo/20/jxme\\_una\\_plataforma.pdf](http://ciruelo.uninorte.edu.co/pdf/ingenieria_desarrollo/20/jxme_una_plataforma.pdf)

ESPIRITY, TROY; YU, WILLIAM. "Universal Chat Client for J2ME Enabled Mobile Devices Using the JXME Implementation of JXTA" [en línia], <http://monevin.ristek.go.id/berita/10.1.1.71.6858.pdf>, [data de consulta: 02 d'Octubre, 2012]

Developer.com. (2012). "JXTA4J2ME Implementation Architecture" [en línia], <http://www.developer.com/java/j2me/article.php/1501461/JXTA4J2ME-Implementation-Architecture.htm>

"JXTA for J2ME" [en línia], <http://www.cs.ru.ac.za/research/g00z1423/Archives/APIs/jxme/overview-summary.html>

"JXME-Proxied" [en línia], <http://smxwiki.a.wiki-site.com/index.php/JXME-Proxied>

"RelayProxyPeer" [en línia], <http://smxwiki.a.wiki-site.com/index.php/RelayProxyPeer>

### **10.3. Referències a Android**

CATALÁN, ADRIAN. (2011). "Curso Android. Desarrollo de aplicaciones móviles". [en línia], <http://books.openlibra.com/pdf/MDW-Guia-Android-1.3.pdf>

Android Platform. (2009). "What is Android?" [en línia] <http://developer.android.com/guide/basics/what-is-android.html>.

VOGEL, LARS. (2009). "Android Development Tutorial". [en línia], <http://www.vogella.com/articles/Android/article.html>

### **10.4. Referències a sensors**

MOLINA, JOSÉ LUÍS. (2002-2011). "Que es un sensor" [en línia], [http://www.profesormolina.com.ar/tecnologia/sens\\_transduct/que\\_es.htm](http://www.profesormolina.com.ar/tecnologia/sens_transduct/que_es.htm), [data de consulta: 23 d'Octubre, 2011]

### **10.5. Referències eines utilitzades**

Wikipedia. (2012). "Eclipse (Software)" [en línia] [http://es.wikipedia.org/wiki/Eclipse\\_\(software\)](http://es.wikipedia.org/wiki/Eclipse_(software)), [data de consulta: 02 de Desembre, 2012]

Wikipedia. (2012). "Netbeans" [en línia] <http://es.wikipedia.org/wiki/NetBeans>, [data de consulta: 02 de Desembre, 2012]

Wikipedia. (2012). "VMWare" [en línia] <http://es.wikipedia.org/wiki/VMware>, [data de consulta: 02 de Desembre, 2012]



Wikipedia. (2012). "*Virtualización*" [en línia] <http://es.wikipedia.org/wiki/Virtualización>, [data de consulta: 02 de Desembre, 2012]

Wikipedia. (2012). "*JXTA*" [en línia] <http://es.wikipedia.org/wiki/JXTA>, [data de consulta: 02 de Desembre, 2012]

Wikipedia. (2012). "*Integrated Development Environment*" [en línia] [http://en.wikipedia.org/wiki/Integrated\\_development\\_environment](http://en.wikipedia.org/wiki/Integrated_development_environment), [data de consulta: 29 de Desembre, 2012]

## **10.6. Altres referències**

Wikipedia. (2012). "*Hypertext Transfer Protocol*" [en línia], <http://en.wikipedia.org/wiki/HTTP>, [data de consulta: 29 de Desembre, 2012]

Wikipedia. (2012). "*XML*" [en línia] <http://en.wikipedia.org/wiki/XML>, [data de consulta: 29 de Desembre, 2012]

## **11. Annexos**

## 11.1. Annex I – Joc de proves realitzat

### Prova 1: Creació de la mota sensora i posada en marxa

Es crea una mota sensora per mitjà de la interfície gràfica i la posem en funcionament:

The image displays two screenshots of the 'Mota Sensora Peer' application interface. The left screenshot shows the 'Sensor Manager' tab with the following fields:

- Mota Sensora Name: MotaSensoraExemple
- Mota Sensora Description: Exemple de Mota Sensora
- Personal Security Environment Execution:
- Execution Mode: EDGE
- Debug Execution:

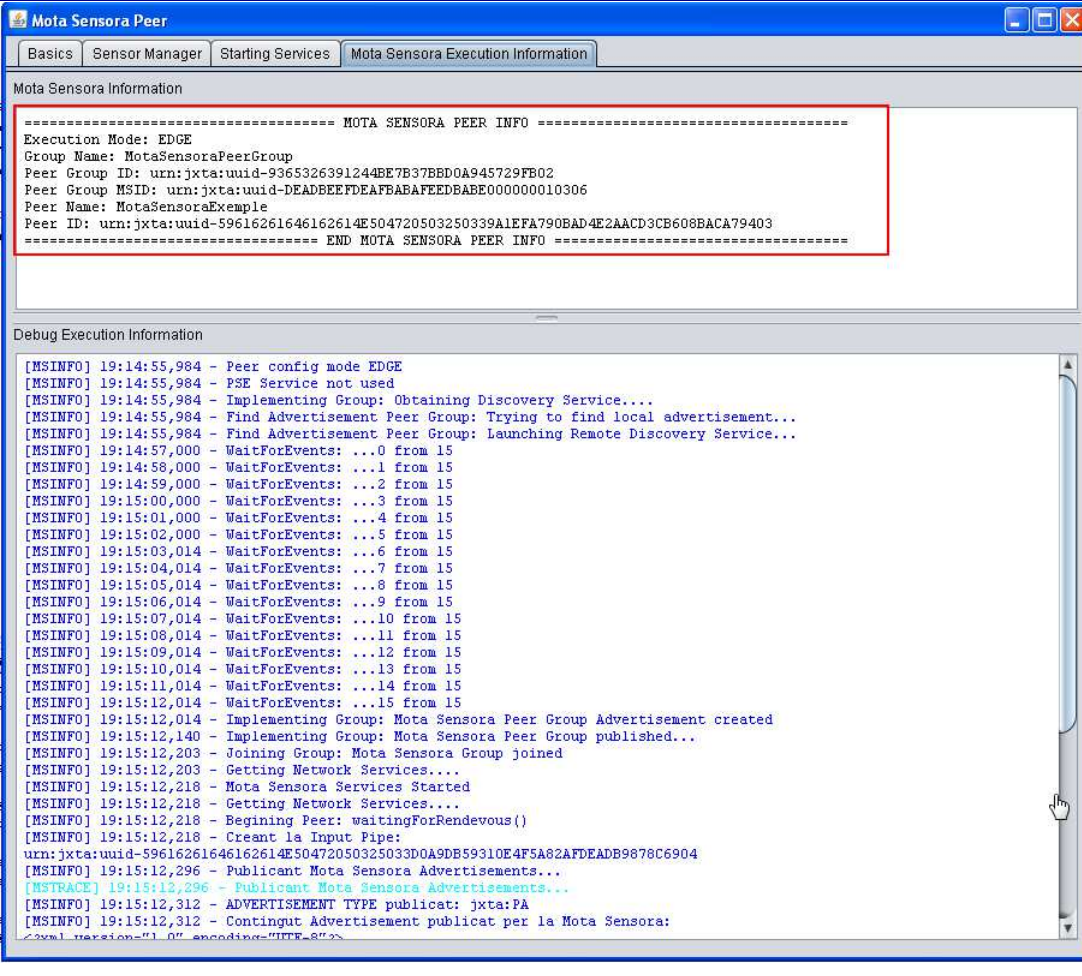
The right screenshot shows the 'Starting Services' tab with the following fields:

- Sensor ID: [Empty]
- Sensor Name: [Empty]
- Sensor Type: Temperature
- Value:  [Empty]
- Unit of measure: Grau Celcius (°C)
- Accuracy: [Empty]

Below these fields are 'Add Sensor' and 'Remove Sensor' buttons. At the bottom, there is a table of existing sensors:

Selected	Sensor ID	Sensor Name	Sensor Type Co...	Sensor Type	Value	Unit Of Measure	Accuracy
<input type="checkbox"/>	S1	Sensor 1	1	Temperature	30.0	Grau Celcius (°C)	1.0
<input type="checkbox"/>	S2	Sensor 2	2	Position	41.67,41.67		10.0
<input type="checkbox"/>	S3	Sensor 3	3	Pressure	0.5	Atmosfera (atm)	0.1
<input type="checkbox"/>	S4	Sensor 4	4	Lighness	1000	Il·luminància (lx)	50
<input type="checkbox"/>	S5	Sensor 5	6	Presence	true		

L'execució es pot seguir per la sortida que es genera a la interfície gràfica.



The screenshot shows the 'Mota Sensora Peer' application window with the 'Mota Sensora Execution Information' tab selected. The window is divided into two main sections: 'Mota Sensora Information' and 'Debug Execution Information'.

**Mota Sensora Information**

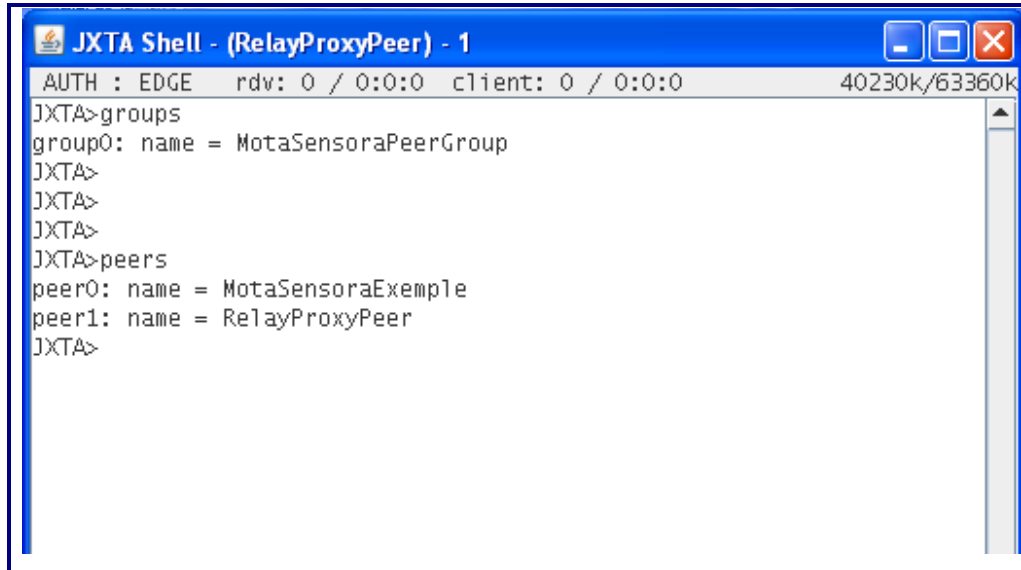
```
===== MOTA SENSORA PEER INFO =====
Execution Mode: EDGE
Group Name: MotaSensoraPeerGroup
Peer Group ID: urn:jxta:uuid-9365326391244BE7B37BED0A945729FB02
Peer Group MSID: urn:jxta:uuid-DEADBEEFDEAFBABAFAFEEDBARE00000010306
Peer Name: MotaSensoraExemple
Peer ID: urn:jxta:uuid-59616261646162614E504720503250339A1EFA790BAD4E2AACD3CB608BACA79403
===== END MOTA SENSORA PEER INFO =====
```

**Debug Execution Information**

```
[MSINFO] 19:14:55,984 - Peer config mode EDGE
[MSINFO] 19:14:55,984 - PSE Service not used
[MSINFO] 19:14:55,984 - Implementing Group: Obtaining Discovery Service...
[MSINFO] 19:14:55,984 - Find Advertisement Peer Group: Trying to find local advertisement...
[MSINFO] 19:14:55,984 - Find Advertisement Peer Group: Launching Remote Discovery Service...
[MSINFO] 19:14:57,000 - WaitForEvents: ...0 from 15
[MSINFO] 19:14:58,000 - WaitForEvents: ...1 from 15
[MSINFO] 19:14:59,000 - WaitForEvents: ...2 from 15
[MSINFO] 19:15:00,000 - WaitForEvents: ...3 from 15
[MSINFO] 19:15:01,000 - WaitForEvents: ...4 from 15
[MSINFO] 19:15:02,000 - WaitForEvents: ...5 from 15
[MSINFO] 19:15:03,014 - WaitForEvents: ...6 from 15
[MSINFO] 19:15:04,014 - WaitForEvents: ...7 from 15
[MSINFO] 19:15:05,014 - WaitForEvents: ...8 from 15
[MSINFO] 19:15:06,014 - WaitForEvents: ...9 from 15
[MSINFO] 19:15:07,014 - WaitForEvents: ...10 from 15
[MSINFO] 19:15:08,014 - WaitForEvents: ...11 from 15
[MSINFO] 19:15:09,014 - WaitForEvents: ...12 from 15
[MSINFO] 19:15:10,014 - WaitForEvents: ...13 from 15
[MSINFO] 19:15:11,014 - WaitForEvents: ...14 from 15
[MSINFO] 19:15:12,014 - WaitForEvents: ...15 from 15
[MSINFO] 19:15:12,014 - Implementing Group: Mota Sensora Peer Group Advertisement created
[MSINFO] 19:15:12,140 - Implementing Group: Mota Sensora Peer Group published...
[MSINFO] 19:15:12,203 - Joining Group: Mota Sensora Group joined
[MSINFO] 19:15:12,203 - Getting Network Services...
[MSINFO] 19:15:12,218 - Mota Sensora Services Started
[MSINFO] 19:15:12,218 - Getting Network Services...
[MSINFO] 19:15:12,218 - Begining Peer: waitingForRendevous()
[MSINFO] 19:15:12,218 - Creant la Input Pipe:
urn:jxta:uuid-59616261646162614E50472050325033D0A9DB59310E4F5A82AFDEADB9878C6904
[MSINFO] 19:15:12,296 - Publicant Mota Sensora Advertisements...
[MSTRACE] 19:15:12,296 - Publicant Mota Sensora Advertisements...
[MSINFO] 19:15:12,312 - ADVERTISEMENT TYPE publicat: jxta:PA
[MSINFO] 19:15:12,312 - Contingut Advertisement publicat per la Mota Sensora:
C:\xml version="1.0" encoding="UTF-8" >
```

## Prova 2: Visualització del Group i de la Mota Sensora en el Relay Peer

Aquesta prova ens permet veure que el Relay Peer pot veure el grup MotaSensoraPeerGroup, creat prèviament per la mota sensora, i pot veure el peer que conté la mota sensora que s'ha creat a l'anterior prova "MotaSensoraExemple".



```
JXTA Shell - (RelayProxyPeer) - 1
AUTH : EDGE   rdv: 0 / 0:0:0   client: 0 / 0:0:0   40230k/63360k
JXTA>groups
group0: name = MotaSensoraPeerGroup
JXTA>
JXTA>
JXTA>peers
peer0: name = MotaSensoraExemple
peer1: name = RelayProxyPeer
JXTA>
```

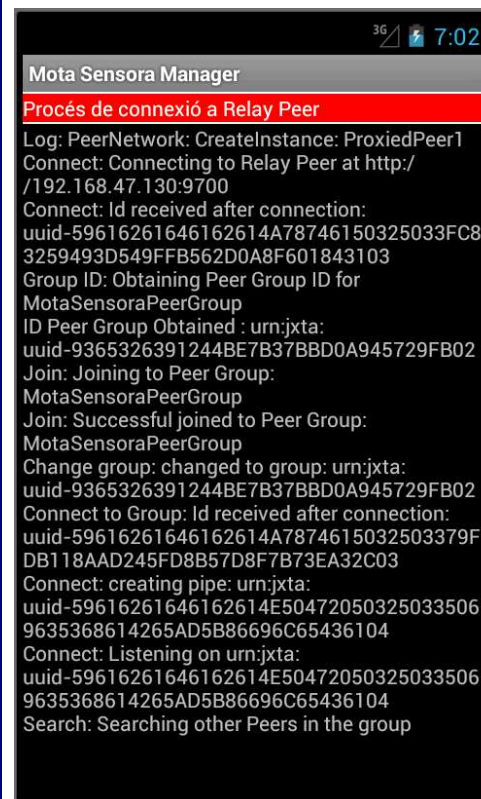
### Prova 3: Connexió de l'aplicació Android Proxied Peer a la xarxa JXTA

Es realitza la connexió d'acord a l'adreçament del Relay Peer on volem connectar:



The screenshot shows the 'Mota Sensora Manager' app interface. At the top, there's a red header 'Connexió a Sistema Mota Sensora'. Below it, there are four input fields: 'Proxied Peer Name' with the value 'ProxiedPeer1', 'Peer Group Name' with 'MotaSensoraPeerGroup', 'Relay Proxy Peer Host' with '192.168.47.130', and 'Relay Proxy Peer Port' with '9700'. The port field is highlighted with an orange border. At the bottom, there are two buttons: 'Close' and 'Connect'.

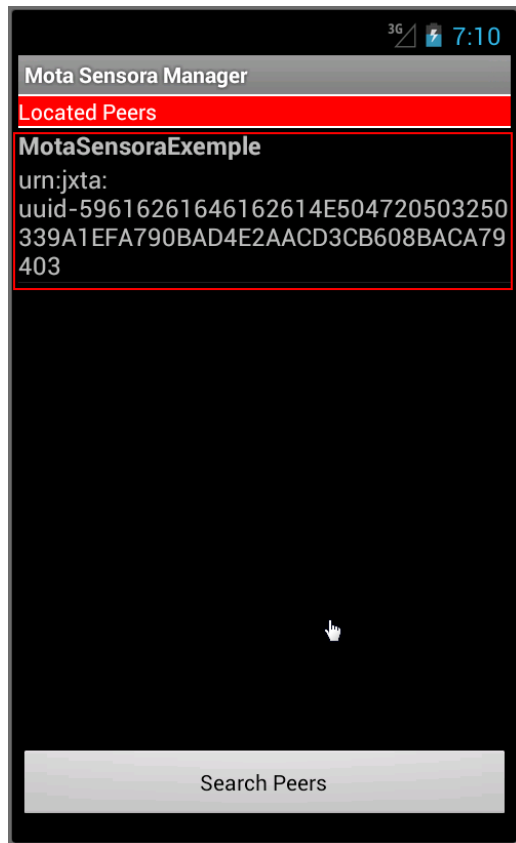
S'inicia el procés de connexió on ha de demanar la cerca del grup, la unió a aquest, crear un canal d'escolta i finalment demanar la cerca de les motes sensors:



The screenshot shows the 'Mota Sensora Manager' app interface with a red header 'Procés de connexió a Relay Peer'. Below it, there is a log of connection events:

```
Log: PeerNetwork: CreateInstance: ProxiedPeer1
Connect: Connecting to Relay Peer at http:/
/192.168.47.130:9700
Connect: Id received after connection:
uuid-59616261646162614A78746150325033FC8
3259493D549FFB562D0A8F601843103
Group ID: Obtaining Peer Group ID for
MotaSensoraPeerGroup
ID Peer Group Obtained : urn:jxta:
uuid-9365326391244BE7B37BBD0A945729FB02
Join: Joining to Peer Group:
MotaSensoraPeerGroup
Join: Successful joined to Peer Group:
MotaSensoraPeerGroup
Change group: changed to group: urn:jxta:
uuid-9365326391244BE7B37BBD0A945729FB02
Connect to Group: Id received after connection:
uuid-59616261646162614A7874615032503379F
DB118AAD245FD8B57D8F7B73EA32C03
Connect: creating pipe: urn:jxta:
uuid-59616261646162614E50472050325033506
9635368614265AD5B86696C65436104
Connect: Listening on urn:jxta:
uuid-59616261646162614E50472050325033506
9635368614265AD5B86696C65436104
Search: Searching other Peers in the group
```

Finalment presenta la llista de les motes sensores trobades; en la nostra prova la MotaSensoraExemple que havíem creat.



## Prova 4: Monitorització de les dades de la mota sensora creada

L'aplicació de Monitorització com a Proxied Peer crea un missatge de petició que s'envia al Relay Peer: En aquest cas es tracta d'una consulta (QUERY) d'una MOTASENSORA. Donem el nostre Peer Id per poder rebre la resposta a través de la Pipe que correspongui i especifiquem que volem tots els atributs (ALL).

```

MotaSensora.ProxiedPeer System.out Element 0: Name=response / Data=result
MotaSensora.ProxiedPeer System.out Element 1: Name=requestId / Data=5
MotaSensora.ProxiedPeer System.out Element 2: Name=type / Data=PIPE
MotaSensora.ProxiedPeer System.out Element 3: Name=name / Data=Pipe Input per Mota Sensora:urn:jxta:uuid-596 0
16261646162614E504720503250339A1EFA790BAD4E2AACD3CB608BACA79403
MotaSensora.ProxiedPeer System.out Element 4: Name=id / Data=urn:jxta:uuid-59616261646162614E50472050325033D0
0A9DB59310E4F5A82AFDEADB9878C6904
MotaSensora.ProxiedPeer System.out Element 5: Name=arg / Data=JxtaUnicast
MotaSensora.ProxiedPeer System.out Element 6: Name=EndpointSourceAddress / Data=jxta://uuid-5961626164616261 0
4A787461503250339581C8R0C2A7494EAA61FOFA524755F703
MotaSensora.ProxiedPeer System.out Element 7: Name=EndpointDestinationAddress / Data=http://0.0.0.0:/Endpoi 0
ntService:uuid-9365326391244BE7E37BED0A945729FB02
MotaSensora.ProxiedPeer System.out Msg Processat: Response Id: 5 Type: PIPE Name: Pipe Input per Mota Sensor 0
a:urn:jxta:uuid-59616261646162614E504720503250339A1EFA790BAD4E2AACD3CB608 0
BACA79403 Id: urn:jxta:uuid-59616261646162614E50472050325033D0A9DB59310E4 0
F5A82AFDEADB9878C6904 Response: result Arg: JxtaUnicast Error:
MotaSensora.ProxiedPeer System.out Pipe ID: urn:jxta:uuid-59616261646162614E50472050325033D0A9DB59310E4F5A82 0
AFDEADB9878C6904
MotaSensora.ProxiedPeer System.out Num elements: 17
MotaSensora.ProxiedPeer System.out Element 0: ":MSG REQUEST" "text/xml" dlen=13 Data 0: <MSG_REQUEST>
MotaSensora.ProxiedPeer System.out Element 1: ":MESSAGE_ID" "text/xml" dlen=12 Data 1: <MESSAGE_ID>
MotaSensora.ProxiedPeer System.out Element 2: ":DATA" "text/xml" dlen=4 Data 2: Id 1
MotaSensora.ProxiedPeer System.out Element 3: ":MESSAGE_ID" "text/xml" dlen=13 Data 3: </MESSAGE_ID>
MotaSensora.ProxiedPeer System.out Element 4: ":MSG_TYPE" "text/xml" dlen=10 Data 4: <MSG_TYPE>
MotaSensora.ProxiedPeer System.out Element 5: ":DATA" "text/xml" dlen=5 Data 5: QUERY
MotaSensora.ProxiedPeer System.out Element 6: ":MSG_TYPE" "text/xml" dlen=11 Data 6: </MSG_TYPE>
MotaSensora.ProxiedPeer System.out Element 7: ":SCOPE" "text/xml" dlen=7 Data 7: <SCOPE>
MotaSensora.ProxiedPeer System.out Element 8: ":DATA" "text/xml" dlen=11 Data 8: MOTASENSORA
MotaSensora.ProxiedPeer System.out Element 9: ":SCOPE" "text/xml" dlen=8 Data 9: </SCOPE>
MotaSensora.ProxiedPeer System.out Element 10: ":ID" "text/xml" dlen=4 Data 10: <ID>
MotaSensora.ProxiedPeer System.out Element 11: ":DATA" "text/xml" dlen=80 Data 11: urn:jxta:uuid-59616261 0
46162614E504720503250335069635368614265AD5B86696C65436104
MotaSensora.ProxiedPeer System.out Element 12: ":ID" "text/xml" dlen=5 Data 12: </ID>
MotaSensora.ProxiedPeer System.out Element 13: ":ATTRIBUTE" "text/xml" dlen=11 Data 13: <ATTRIBUTE>
MotaSensora.ProxiedPeer System.out Element 14: ":DATA" "text/xml" dlen=3 Data 14: ALL
MotaSensora.ProxiedPeer System.out Element 15: ":ATTRIBUTE" "text/xml" dlen=12 Data 15: </ATTRIBUTE>
MotaSensora.ProxiedPeer System.out Element 16: ":MSG REQUEST" "text/xml" dlen=14 Data 16: </MSG_REQUEST>
MotaSensora.ProxiedPeer System.out Sending message to Pipe: urn:jxta:uuid-59616261646162614E50472050325033D0 0
A9DB59310E4F5A82AFDEADB9878C6904
    
```



A la sortida de la mota sensora es pot observar que es rep el missatge de petició. Aquest és processat i utilitza el Pipe Id contingut al missatge que correspon al Proxied Peer a fi de retornar la resposta. Es crea un canal de sortida "Output Pipe" on es posarà la resposta.

```
[MSINFO] 20:19:00,656 - Tipus de missatge rebut:
MSG_REQUEST
[MSINFO] 20:19:00,656 - Missatge rebut:
<?xml version="1.0"?>
<!DOCTYPE MSG_REQUEST>
<MSG_REQUEST>
  <MESSAGE_ID>
    Id 1
  </MESSAGE_ID>
  <MSG_TYPE>
    QUERY
  </MSG_TYPE>
  <SCOPE>
    MOTASENSORA
  </SCOPE>
  <ID>
    urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
  </ID>
  <ATTRIBUTE>
    ALL
  </ATTRIBUTE>
</MSG_REQUEST>

[MSINFO] 20:19:00,671 - findAdvertisement - Buscant advertisements...
[MSINFO] 20:19:00,687 - S'han trobat advertisements...
[MSINFO] 20:19:00,687 - Es Pipe Advertisements...:
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jxta:PipeAdvertisement>
<jxta:PipeAdvertisement xml:space="default" xmlns:jxta="http://jxta.org">
  <Id>
    urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
  </Id>
  <Type>
    JxtaPropagate
  </Type>
  <Name>
    ProxiedPeer1
  </Name>
</jxta:PipeAdvertisement>

[MSINFO] 20:19:00,703 - Output Pipe created: urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65
```

Tot i que no és possible observar tot el contingut del missatge de resposta, la mota sensora genera la resposta a l'anterior petició. Aquesta és enviada a l'aplicació de monitorització.

```
[MSINFO] 20:19:00,703 - Output Pipe created: urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
[MSINFO] 20:19:00,703 - Missatge resposta:
<?xml version="1.0"?>
<!DOCTYPE MSG_RESPONSE>
<MSG_RESPONSE>
  <MESSAGE_ID>
    Id 1
  </MESSAGE_ID>
  <NAME>
    MotaSensoraExemple
  </NAME>
  <PEER_ID>
    urn:jxta:uuid-59616261646162614E504720503250339A1EFA790BAD4E2AACD3CB608BACA79408
  </PEER_ID>
  <DESCRIPTION>
    Exemple de Mota Sensora
  </DESCRIPTION>
  <CONNECTED>
    true
  </CONNECTED>
  <STATIC>
    false
  </STATIC>
  <SENSOR_LIST>
    <SENSOR>
      <SENSOR_ID>
        S5
      </SENSOR_ID>
      <SENSOR_NAME>
        Sensor 5
      </SENSOR_NAME>
      <TYPE>
        6
      </TYPE>
    </SENSOR>
    <SENSOR>
      <SENSOR_ID>
        S4
      </SENSOR_ID>
      <SENSOR_NAME>
        Sensor 4
      </SENSOR_NAME>
      <TYPE>
        4
      </TYPE>
    </SENSOR>
  </SENSOR_LIST>
</MSG_RESPONSE>
```

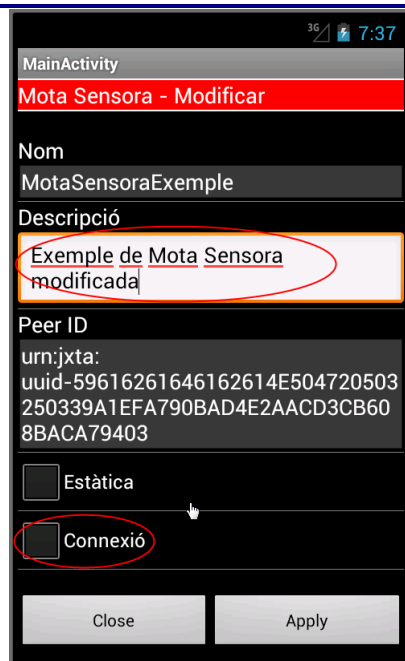


La sortida de les dades de la mota sensora es presenten de la següent forma a l'aplicació de Monitorització.



### Prova 5: Control de la mota sensora creada (modificació de les dades)

Obrim el control de la mota sensora i modifiquem algunes dades. En aquest cas es modifica la Descripció afegint "modificada" i posem el camp Connexió com no connectat. Fem "Apply" per aplicar els canvis.



The screenshot shows an Android application interface titled "MainActivity" with a red header "Mota Sensora - Modificar". The interface displays the following fields and controls:

- Nom:** MotaSensoraExemple
- Descripció:** Exemple de Mota Sensora modificada (circled in red)
- Peer ID:** urn:jxta:uuid-59616261646162614E504720503250339A1EFA790BAD4E2AACD3CB608BACA79403
- Estàtica:**
- Connexió:**  (circled in red)
- Buttons:** Close and Apply

Una vegada aplicats apareix breument un missatge a pantalla que ho especifica.



Seguidament ens torna a la pantalla prèvia de monitorització, on es pot observar que efectivament les dades estan modificades.



Però que succeït exactament?

El Proxied Peer ha generat un missatge de petició de modificació (UPDATE) d'una MOTASENSORA, on li diem que la DESCRIPTION té el nou valor "Exemple de Mota Sensora modificada".

```
System.out Num elements: 20
System.out Element 0: ":MSG_REQUEST" "text/xml" dlen=13 Data 0: <MSG_REQUEST>
System.out Element 1: ":MESSAGE_ID" "text/xml" dlen=12 Data 1: <MESSAGE_ID>
System.out Element 2: ":DATA" "text/xml" dlen=34 Data 2: Rqst_DESCRIPTION_20121130 0
193825830
System.out Element 3: ":MESSAGE_ID" "text/xml" dlen=13 Data 3: </MESSAGE_ID>
System.out Element 4: ":MSG_TYPE" "text/xml" dlen=10 Data 4: <MSG_TYPE>
System.out Element 5: ":DATA" "text/xml" dlen=6 Data 5: UPDATE
System.out Element 6: ":MSG_TYPE" "text/xml" dlen=11 Data 6: </MSG_TYPE>
System.out Element 7: ":SCOPE" "text/xml" dlen=7 Data 7: <SCOPE>
System.out Element 8: ":DATA" "text/xml" dlen=11 Data 8: MOTASENSORA
System.out Element 9: ":SCOPE" "text/xml" dlen=8 Data 9: </SCOPE>
System.out Element 10: ":ID" "text/xml" dlen=4 Data 10: <ID>
System.out Element 11: ":DATA" "text/xml" dlen=80 Data 11: urn:jxta:uuid-596162616 0
46162614E504720503250335069635368614265AD5B86696C65436104
System.out Element 12: ":ID" "text/xml" dlen=5 Data 12: </ID>
System.out Element 13: ":ATTRIBUTE" "text/xml" dlen=11 Data 13: <ATTRIBUTE>
System.out Element 14: ":DATA" "text/xml" dlen=11 Data 14: DESCRIPTION
System.out Element 15: ":ATTRIBUTE" "text/xml" dlen=12 Data 15: </ATTRIBUTE>
System.out Element 16: ":VALUE" "text/xml" dlen=7 Data 16: <VALUE>
System.out Element 17: ":DATA" "text/xml" dlen=34 Data 17: Exemple de Mota Sensora 0
modificada
System.out Element 18: ":VALUE" "text/xml" dlen=8 Data 18: </VALUE>
System.out Element 19: ":MSG_REQUEST" "text/xml" dlen=14 Data 19: </MSG_REQUEST>
System.out Sending message to Pipe: urn:jxta:uuid-59616261646162614E50472050325033D0 0
A9DB59310E4F5A82AFDEADEB9878C6904
```



A la seva banda, la mota sensora rep la petició, la processa i genera la resposta. En ella indica que l'execució ha estat correcta i l'envia al Proxied Peer a través del canal que aquest escolta.

```
[MSINFO] 20:38:51,859 - Missatge rebut:
<?xml version="1.0"?>
<!DOCTYPE MSG_REQUEST>
<MSG_REQUEST>
  <MESSAGE_ID>
    Rqst_DESCRIPTION_20121130193825830
  </MESSAGE_ID>
  <MSG_TYPE>
    UPDATE
  </MSG_TYPE>
  <SCOPE>
    MOTASENSORA
  </SCOPE>
  <ID>
    urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
  </ID>
  <ATTRIBUTE>
    DESCRIPTION
  </ATTRIBUTE>
  <VALUE>
    Exemple de Mota Sensora modificada
  </VALUE>
</MSG_REQUEST>

[MSINFO] 20:38:51,875 - findAdvertisement - Buscant advertisements...
[MSINFO] 20:38:51,875 - S'han trobat advertisements...
[MSINFO] 20:38:51,890 - És Pipe Advertisements...
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jxta:PipeAdvertisement>
<jxta:PipeAdvertisement xml:space="default" xmlns:jxta="http://jxta.org">
  <Id>
    urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
  </Id>
  <Type>
    JxtaPropagate
  </Type>
  <Name>
    ProxiedPeer1
  </Name>
</jxta:PipeAdvertisement>

[MSINFO] 20:38:51,890 - Output Pipe created: urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
[MSINFO] 20:38:51,890 - Missatge resposta:
<?xml version="1.0"?>
<!DOCTYPE MSG_RESPONSE>
<MSG_RESPONSE>
  <MESSAGE_ID>
    Rqst_DESCRIPTION_20121130193825830
  </MESSAGE_ID>
  <EXECUTION>
    OK
  </EXECUTION>
</MSG_RESPONSE>
```



La mota sensora rep la petició d'UPDATE, la processa, respon que l'ha processat correctament i envia el resultat.

```
[MSINFO] 20:38:57,500 - Missatge rebut:
<?xml version="1.0"?>
<!DOCTYPE MSG_REQUEST>
<MSG_REQUEST>
  <MESSAGE_ID>
    Rqst_CONNECTED_20121130193831950
  </MESSAGE_ID>
  <MSG_TYPE>
    UPDATE
  </MSG_TYPE>
  <SCOPE>
    MOTASENSORA
  </SCOPE>
  <ID>
    urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
  </ID>
  <ATTRIBUTE>
    CONNECTED
  </ATTRIBUTE>
  <VALUE>
    false
  </VALUE>
</MSG_REQUEST>

[MSINFO] 20:38:57,515 - findAdvertisement - Buscant advertisements...
[MSINFO] 20:38:57,531 - S'han trobat advertisements...
[MSINFO] 20:38:57,531 - És Pipe Advertisements...
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jxta:PipeAdvertisement>
<jxta:PipeAdvertisement xml:space="default" xmlns:jxta="http://jxta.org">
  <Id>
    urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
  </Id>
  <Type>
    JxtaPropagate
  </Type>
  <Name>
    ProxiedPeer1
  </Name>
</jxta:PipeAdvertisement>

[MSINFO] 20:38:57,531 - Output Pipe created: urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
[MSINFO] 20:38:57,531 - Missatge resposta:
<?xml version="1.0"?>
<!DOCTYPE MSG_RESPONSE>
<MSG_RESPONSE>
  <MESSAGE_ID>
    Rqst_CONNECTED_20121130193831950
  </MESSAGE_ID>
  <VALUE>
    false
  </VALUE>
  <EXECUTION>
    OK
  </EXECUTION>
</MSG_RESPONSE>
```

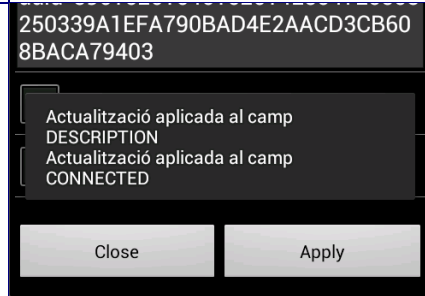
Per la seva banda el Proxied Peer rebrà la confirmació de què tot ha anat correctament.

```

MotaSensora.ProxiedPeer System.out 0 <?xml version="1.0"?>
MotaSensora.ProxiedPeer System.out <!DOCTYPE MSG_RESPONSE>
MotaSensora.ProxiedPeer System.out <MSG_RESPONSE>
MotaSensora.ProxiedPeer System.out <<MESSAGE_ID>
MotaSensora.ProxiedPeer System.out <<Rqst_CONNECTED_20121130193831950
MotaSensora.ProxiedPeer System.out <</MESSAGE_ID>
MotaSensora.ProxiedPeer System.out <<VALUE>
MotaSensora.ProxiedPeer System.out <<false
MotaSensora.ProxiedPeer System.out <</VALUE>
MotaSensora.ProxiedPeer System.out <<EXECUTION>
MotaSensora.ProxiedPeer System.out <<OK
MotaSensora.ProxiedPeer System.out <</EXECUTION>
MotaSensora.ProxiedPeer System.out </MSG_RESPONSE>
MotaSensora.ProxiedPeer System.out 1 LimitedRangeRdvMessage <?xml version="1.0" encoding="UTF-8"?>
MotaSensora.ProxiedPeer System.out <!DOCTYPE jxta:LimitedRangeRdvMessage>
MotaSensora.ProxiedPeer System.out <jxta:LimitedRangeRdvMessage xmlns:jxta="http://jxta.org">
MotaSensora.ProxiedPeer System.out <<TTL>
MotaSensora.ProxiedPeer System.out <<2
-- --

```

...es mostra el missatge, com s'ha indicat anteriorment, i es torna a la pantalla de monitorització.



## Prova 6: Monitorització d'un sensor de la mota sensora

Des de la llista de sensors de la interfície de Monitorització de la mota sensora marquem un sensor. La petició generada ens mostra les dades del sensor com es pot veure a la següent captura.



Per obtenir les dades del sensor es genera una petició de consulta (QUERY) del sensor S1 on volem obtenir totes les seves dades (ALL). Aquesta petició és enviada a la mota sensora i prèviament el Relay Peer realitzarà la traducció a un format XML, interpretable per la mota sensora.

```

MotaSensora.ProxiedPeer System.out Num elements: 20
MotaSensora.ProxiedPeer System.out Element 0: ":MSG_REQUEST" "text/xml" dlen=13 Data 0: <MSG_REQUEST>
MotaSensora.ProxiedPeer System.out Element 1: ":MESSAGE_ID" "text/xml" dlen=12 Data 1: <MESSAGE_ID>
MotaSensora.ProxiedPeer System.out Element 2: ":DATA" "text/xml" dlen=32 Data 2: Rqst_SENSOR_ID_2012113020 0
0717466
Element 3: ":MESSAGE_ID" "text/xml" dlen=13 Data 3: </MESSAGE_ID>
Element 4: ":MSG_TYPE" "text/xml" dlen=10 Data 4: <MSG_TYPE>
Element 5: ":DATA" "text/xml" dlen=5 Data 5: QUERY
Element 6: ":MSG_TYPE" "text/xml" dlen=11 Data 6: </MSG_TYPE>
Element 7: ":SCOPE" "text/xml" dlen=7 Data 7: <SCOPE>
Element 8: ":DATA" "text/xml" dlen=6 Data 8: SENSOR
Element 9: ":SCOPE" "text/xml" dlen=8 Data 9: </SCOPE>
Element 10: ":SENSOR_ID" "text/xml" dlen=11 Data 10: <SENSOR_ID>
Element 11: ":DATA" "text/xml" dlen=2 Data 11: S1
Element 12: ":SENSOR_ID" "text/xml" dlen=12 Data 12: </SENSOR_ID>
Element 13: ":ID" "text/xml" dlen=4 Data 13: <ID>
Element 14: ":DATA" "text/xml" dlen=80 Data 14: urn:jxta:uuid-596162616 0
46162614E504720503250335069635368614265AD5B86696C65436104
Element 15: ":ID" "text/xml" dlen=5 Data 15: </ID>
Element 16: ":ATTRIBUTE" "text/xml" dlen=11 Data 16: <ATTRIBUTE>
Element 17: ":DATA" "text/xml" dlen=3 Data 17: ALL
Element 18: ":ATTRIBUTE" "text/xml" dlen=12 Data 18: </ATTRIBUTE>
Element 19: ":MSG_REQUEST" "text/xml" dlen=14 Data 19: </MSG_REQUEST>
Sending message to Pipe: urn:jxta:uuid-59616261646162614E50472050325033D0 0
A9DB59310E4F5A82AFDEADB9878C6904

```

La mota sensora rep el següent missatge, el processa i genera el canal de sortida (output pipe) per enviar al Input Pipe del Proxied Peer la resposta.

```
[MSINFO] 21:07:42,640 - Tipus de missatge rebut:
MSG_REQUEST
[MSINFO] 21:07:42,640 - Missatge rebut:
<?xml version="1.0"?>
<!DOCTYPE MSG_REQUEST>
<MSG_REQUEST>
  <MESSAGE_ID>
    Rqst_SENSOR_ID_20121130200717466
  </MESSAGE_ID>
  <MSG_TYPE>
    QUERY
  </MSG_TYPE>
  <SCOPE>
    SENSOR
  </SCOPE>
  <SENSOR_ID>
    S1
  </SENSOR_ID>
  <ID>
    urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
  </ID>
  <ATTRIBUTE>
    ALL
  </ATTRIBUTE>
</MSG_REQUEST>

[MSINFO] 21:07:42,640 - findAdvertisement - Buscant advertisements...
[MSINFO] 21:07:42,640 - s'han trobat advertisements...
[MSINFO] 21:07:42,640 - És Pipe Advertisements...:
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jxta:PipeAdvertisement>
<jxta:PipeAdvertisement xml:space="default" xmlns:jxta="http://jxta.org">
  <Id>
    urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
  </Id>
  <Type>
    JxtaPropagate
  </Type>
  <Name>
    ProxiedPeer1
  </Name>
</jxta:PipeAdvertisement>

[MSINFO] 21:07:42,640 - Output Pipe created: urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
```

La resposta que genera conté totes les dades del sensor.

```
[MSINFO] 21:07:42,640 - Output Pipe created: urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
[MSINFO] 21:07:42,640 - Missatge resposta:
<?xml version="1.0"?>
<!DOCTYPE MSG_RESPONSE>
<MSG_RESPONSE>
  <MESSAGE_ID>
    Rqst_SENSOR_ID_20121130200717466
  </MESSAGE_ID>
  <SENSOR_ID>
    S1
  </SENSOR_ID>
  <SENSOR_NAME>
    Sensor 1
  </SENSOR_NAME>
  <STATUSON>
    true
  </STATUSON>
  <STANDBY>
    true
  </STANDBY>
  <TYPE>
    1
  </TYPE>
  <TEMPERATURE>
    30.0
  </TEMPERATURE>
  <ACCURACY>
    1.0
  </ACCURACY>
  <UNITOFMEASURE>
    Grau Celcius (°C)
  </UNITOFMEASURE>
</MSG_RESPONSE>
```



Un cop traduïdes pel Relay Peer, el Proxied Peer (aplicació de monitorització) rep les següents dades traduïdes.

```

MotaSensora.ProxiedPeer System.out Con Length: 1433
MotaSensora.ProxiedPeer System.out 0 <?xml version="1.0"?>
MotaSensora.ProxiedPeer System.out <!DOCTYPE MSG_RESPONSE>
MotaSensora.ProxiedPeer System.out <MSG_RESPONSE>
MotaSensora.ProxiedPeer System.out □<MESSAGE_ID>
MotaSensora.ProxiedPeer System.out □□Rqst_SENSOR_ID_20121130200717466
MotaSensora.ProxiedPeer System.out □</MESSAGE_ID>
MotaSensora.ProxiedPeer System.out □<SENSOR_ID>
MotaSensora.ProxiedPeer System.out □□S1
MotaSensora.ProxiedPeer System.out □</SENSOR_ID>
MotaSensora.ProxiedPeer System.out □<SENSOR_NAME>
MotaSensora.ProxiedPeer System.out □□Sensor 1
MotaSensora.ProxiedPeer System.out □</SENSOR_NAME>
MotaSensora.ProxiedPeer System.out □<STATUSON>
MotaSensora.ProxiedPeer System.out □□true
MotaSensora.ProxiedPeer System.out □</STATUSON>
MotaSensora.ProxiedPeer System.out □<STANDBY>
MotaSensora.ProxiedPeer System.out □□true
MotaSensora.ProxiedPeer System.out □</STANDBY>
MotaSensora.ProxiedPeer System.out □<TYPE>
MotaSensora.ProxiedPeer System.out □□1
MotaSensora.ProxiedPeer System.out □</TYPE>
MotaSensora.ProxiedPeer System.out □<TEMPERATURE>
MotaSensora.ProxiedPeer System.out □□30.0
MotaSensora.ProxiedPeer System.out □</TEMPERATURE>
MotaSensora.ProxiedPeer System.out □<ACCURACY>
MotaSensora.ProxiedPeer System.out □□1.0
MotaSensora.ProxiedPeer System.out □</ACCURACY>
MotaSensora.ProxiedPeer System.out □<UNITOFMEASURE>
MotaSensora.ProxiedPeer System.out □□Grau Celcius (*C)
MotaSensora.ProxiedPeer System.out □</UNITOFMEASURE>
MotaSensora.ProxiedPeer System.out </MSG_RESPONSE>
MotaSensora.ProxiedPeer System.out 1 LimitedRangeRdvMessage <?xml version="1.0" encoding="UTF-8"?>
MotaSensora.ProxiedPeer System.out <!DOCTYPE jxta:LimitedRangeRdvMessage>
MotaSensora.ProxiedPeer System.out <jxta:LimitedRangeRdvMessage xmlns:jxta="http://jxta.org">
MotaSensora.ProxiedPeer System.out □<TTL>
MotaSensora.ProxiedPeer System.out □□2
MotaSensora.ProxiedPeer System.out □</TTL>
MotaSensora.ProxiedPeer System.out □<Dir>
MotaSensora.ProxiedPeer System.out □□2
MotaSensora.ProxiedPeer System.out □</Dir>
MotaSensora.ProxiedPeer System.out □<SrcPeerID>
MotaSensora.ProxiedPeer System.out □□urn:jxta:uuid-59616261646162614E504720503250339A1EFA790BAD4E2AACD3CB608 □
BACA79403
MotaSensora.ProxiedPeer System.out □</SrcPeerID>

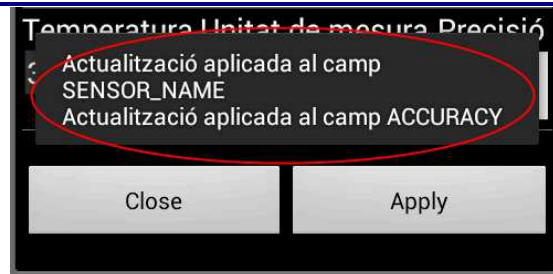
```

## Prova 7: Control d'un sensor de la mota sensora (modificació de les dades)

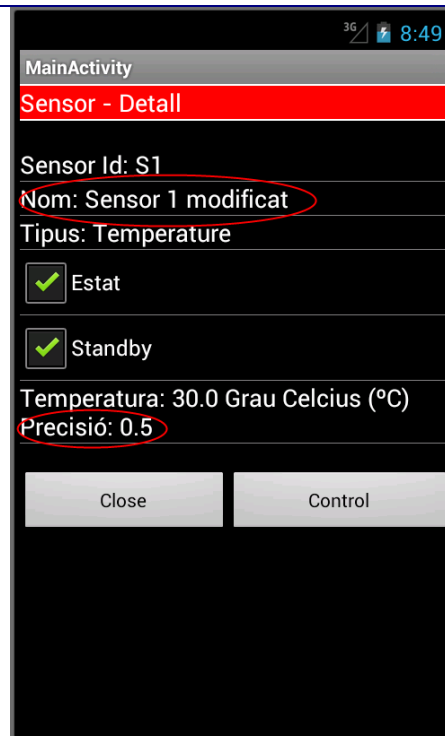
Des de la interfície de control del sensor, a la que hem accedit a través del botó "Control" de la interfície de monitorització del sensor, modifiquem el nom (afegint modificat) i canviem la precisió de la temperatura de 1.0 a 0.5. Fem "Apply" per aplicar els canvis.

MainActivity  
Sensor - Modificar  
Sensor Id  
S1  
Nom  
Sensor 1 modificat  
Tipus  
Temperature  
 Estat  
 Standby  
Temperatura Unitat de mesura Precisió  
30.0 Grau Celcius (°C) 0.5  
Close Apply

Després de fer els canvis, es mostra breument un missatge que indica que els canvis han estat realitzats.



Una vegada finalitzat el procés es torna a la pantalla de monitorització del sensor on es pot comprovar que els canvis han estat aplicats.



Per arribar fins aquí, prèviament ha succeït el següent:

S'ha fet una petició de modificació (UPDATE) del sensor amb identificador S1 on volem que el camp SENSOR\_NAME sigui "Sensor 1 modificat". L'aplicació de Monitorització com a Proxied Peer que és, envia la petició.

```
MotaSensora.ProxiedPeer System.out Num elements: 23
MotaSensora.ProxiedPeer System.out Element 0: ":MSG_REQUEST" "text/xml" dlen=13 Data 0: <MSG_REQUEST>
MotaSensora.ProxiedPeer System.out Element 1: ":MESSAGE_ID" "text/xml" dlen=12 Data 1: <MESSAGE_ID>
MotaSensora.ProxiedPeer System.out Element 2: ":DATA" "text/xml" dlen=34 Data 2: Rqst_SENSOR_NAME_20121130 0
204914796
MotaSensora.ProxiedPeer System.out Element 3: ":MESSAGE_ID" "text/xml" dlen=13 Data 3: </MESSAGE_ID>
MotaSensora.ProxiedPeer System.out Element 4: ":MSG_TYPE" "text/xml" dlen=10 Data 4: <MSG_TYPE>
MotaSensora.ProxiedPeer System.out Element 5: ":DATA" "text/xml" dlen=6 Data 5: UPDATE
MotaSensora.ProxiedPeer System.out Element 6: ":MSG_TYPE" "text/xml" dlen=11 Data 6: </MSG_TYPE>
MotaSensora.ProxiedPeer System.out Element 7: ":SCOPE" "text/xml" dlen=7 Data 7: <SCOPE>
MotaSensora.ProxiedPeer System.out Element 8: ":DATA" "text/xml" dlen=6 Data 8: <SENSOR
MotaSensora.ProxiedPeer System.out Element 9: ":SCOPE" "text/xml" dlen=8 Data 9: </SCOPE>
MotaSensora.ProxiedPeer System.out Element 10: ":SENSOR_ID" "text/xml" dlen=11 Data 10: <SENSOR_ID>
MotaSensora.ProxiedPeer System.out Element 11: ":DATA" "text/xml" dlen=2 Data 11: S1
MotaSensora.ProxiedPeer System.out Element 12: ":SENSOR_ID" "text/xml" dlen=12 Data 12: </SENSOR_ID>
MotaSensora.ProxiedPeer System.out Element 13: ":ID" "text/xml" dlen=4 Data 13: <ID>
MotaSensora.ProxiedPeer System.out Element 14: ":DATA" "text/xml" dlen=80 Data 14: urn:jxta:uuid-596162616
46162614E504720503250335069635368614265AD5B86696C65436104
MotaSensora.ProxiedPeer System.out Element 15: ":ID" "text/xml" dlen=5 Data 15: </ID>
MotaSensora.ProxiedPeer System.out Element 16: ":ATTRIBUTE" "text/xml" dlen=11 Data 16: <ATTRIBUTE>
MotaSensora.ProxiedPeer System.out Element 17: ":DATA" "text/xml" dlen=11 Data 17: <SENSOR_NAME
MotaSensora.ProxiedPeer System.out Element 18: ":ATTRIBUTE" "text/xml" dlen=12 Data 18: </ATTRIBUTE>
MotaSensora.ProxiedPeer System.out Element 19: ":VALUE" "text/xml" dlen=7 Data 19: <VALUE>
MotaSensora.ProxiedPeer System.out Element 20: ":DATA" "text/xml" dlen=18 Data 20: Sensor 1 modificat
MotaSensora.ProxiedPeer System.out Element 21: ":VALUE" "text/xml" dlen=8 Data 21: </VALUE>
MotaSensora.ProxiedPeer System.out Element 22: ":MSG_REQUEST" "text/xml" dlen=14 Data 22: </MSG_REQUEST>
MotaSensora.ProxiedPeer System.out Sending message to Pipe: urn:jxta:uuid-59616261646162614E50472050325033D0 0
A9DB59310E4F5A82AFDEADB9878C6904
```

El Relay Peer fa la traducció i l'adreça a la mota sensora que rep el missatge en format XML. Processa la petició i obre un canal de sortida (Output pipe) al canal d'entrada de l'aplicació de Monitorització.

```
[MSINFO] 21:49:39,984 - Missatge rebut:
<?xml version="1.0"?>
<!DOCTYPE MSG_REQUEST>
<MSG_REQUEST>
  <MESSAGE_ID>
    Rqst_SENSOR_NAME_20121130204914796
  </MESSAGE_ID>
  <MSG_TYPE>
    UPDATE
  </MSG_TYPE>
  <SCOPE>
    SENSOR
  </SCOPE>
  <SENSOR_ID>
    S1
  </SENSOR_ID>
  <ID>
    urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
  </ID>
  <ATTRIBUTE>
    SENSOR_NAME
  </ATTRIBUTE>
  <VALUE>
    Sensor 1 modificat
  </VALUE>
</MSG_REQUEST>

[MSINFO] 21:49:39,984 - findAdvertisement - Buscant advertisements...
[MSINFO] 21:49:40,014 - S'han trobat advertisements...
[MSINFO] 21:49:40,014 - És Pipe Advertisements...:
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jxta:PipeAdvertisement>
<jxta:PipeAdvertisement xml:space="default" xmlns:jxta="http://jxta.org">
  <Id>
    urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
  </Id>
  <Type>
    JxtaPropagate
  </Type>
  <Name>
    ProxiedPeer1
  </Name>
</jxta:PipeAdvertisement>

[MSINFO] 21:49:40,014 - Output Pipe created: urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
```

<p>Genera la resposta amb el resultat de la petició i el torna a l'aplicació de monitorització.</p>	<pre>[MSINFO] 21:49:40,014 - Output Pipe created: urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104 [MSINFO] 21:49:40,014 - Missatge resposta: &lt;?xml version="1.0"?&gt; &lt;!DOCTYPE MSG_RESPONSE&gt; &lt;MSG_RESPONSE&gt;   &lt;MESSAGE_ID&gt;     Rqst_SENSOR_NAME_20121130204914796   &lt;/MESSAGE_ID&gt;   &lt;EXECUTION&gt;     OK   &lt;/EXECUTION&gt; &lt;/MSG_RESPONSE&gt;</pre>
<p>Una vegada el missatge de resposta ha estat traduït pel Relay Peer, aquest és rebut per l'aplicació de Monitorització tal i com indica la figura.</p>	<pre>MotaSensora.ProxiedPeer System.out Con Length: 1185 MotaSensora.ProxiedPeer System.out 0 &lt;?xml version="1.0"?&gt; MotaSensora.ProxiedPeer System.out &lt;!DOCTYPE MSG_RESPONSE&gt; MotaSensora.ProxiedPeer System.out &lt;MSG_RESPONSE&gt; MotaSensora.ProxiedPeer System.out □&lt;MESSAGE_ID&gt; MotaSensora.ProxiedPeer System.out □□Rqst_SENSOR_NAME_20121130204914796 MotaSensora.ProxiedPeer System.out □&lt;/MESSAGE_ID&gt; MotaSensora.ProxiedPeer System.out □&lt;EXECUTION&gt; MotaSensora.ProxiedPeer System.out □□OK MotaSensora.ProxiedPeer System.out □&lt;/EXECUTION&gt; MotaSensora.ProxiedPeer System.out &lt;/MSG_RESPONSE&gt; MotaSensora.ProxiedPeer System.out 1 LimitedRangeRdvMessage &lt;?xml version="1.0" encoding="UTF-8"?&gt; MotaSensora.ProxiedPeer System.out &lt;!DOCTYPE jxta:LimitedRangeRdvMessage&gt; MotaSensora.ProxiedPeer System.out &lt;jxta:LimitedRangeRdvMessage xmlns:jxta="http://jxta.org"&gt; MotaSensora.ProxiedPeer System.out □&lt;TTL&gt; MotaSensora.ProxiedPeer System.out □□2 MotaSensora.ProxiedPeer System.out □&lt;/TTL&gt;</pre>

Un cop s'ha completat la primera modificació es genera la petició de la segona. Es demana una modificació (UPDATE) d'un SENSOR amb identificador S1 i volem que la seva precisió sigui 0.5.

L'enviem a la mota sensora, però prèviament el Relay Peer farà la tasca de traducció a un format interpretable per la mota sensora (XML).

```

MotaSensora.ProxiedPeer System.out Num elements: 23
MotaSensora.ProxiedPeer System.out Element 0: ":MSG_REQUEST" "text/xml" dlen=13 Data 0: <MSG_REQUEST>
MotaSensora.ProxiedPeer System.out Element 1: ":MESSAGE_ID" "text/xml" dlen=12 Data 1: <MESSAGE_ID>
MotaSensora.ProxiedPeer System.out Element 2: ":DATA" "text/xml" dlen=31 Data 2: Rqst_ACCURACY_20121130204 0
920411
MotaSensora.ProxiedPeer System.out Element 3: ":MESSAGE_ID" "text/xml" dlen=13 Data 3: </MESSAGE_ID>
MotaSensora.ProxiedPeer System.out Element 4: ":MSG_TYPE" "text/xml" dlen=10 Data 4: <MSG_TYPE>
MotaSensora.ProxiedPeer System.out Element 5: ":DATA" "text/xml" dlen=6 Data 5: UPDATE
MotaSensora.ProxiedPeer System.out Element 6: ":MSG_TYPE" "text/xml" dlen=11 Data 6: </MSG_TYPE>
MotaSensora.ProxiedPeer System.out Element 7: ":SCOPE" "text/xml" dlen=7 Data 7: <SCOPE>
MotaSensora.ProxiedPeer System.out Element 8: ":DATA" "text/xml" dlen=6 Data 8: SENSOR
MotaSensora.ProxiedPeer System.out Element 9: ":SCOPE" "text/xml" dlen=8 Data 9: </SCOPE>
MotaSensora.ProxiedPeer System.out Element 10: ":SENSOR_ID" "text/xml" dlen=11 Data 10: <SENSOR_ID>
MotaSensora.ProxiedPeer System.out Element 11: ":DATA" "text/xml" dlen=2 Data 11: S1
MotaSensora.ProxiedPeer System.out Element 12: ":SENSOR_ID" "text/xml" dlen=12 Data 12: </SENSOR_ID>
MotaSensora.ProxiedPeer System.out Element 13: ":ID" "text/xml" dlen=4 Data 13: <ID>
MotaSensora.ProxiedPeer System.out Element 14: ":DATA" "text/xml" dlen=80 Data 14: urn:jxta:uuid-596162616 0
46162614R504720503250335069635368614265AD5B86696C65436104
MotaSensora.ProxiedPeer System.out Element 15: ":ID" "text/xml" dlen=5 Data 15: </ID>
MotaSensora.ProxiedPeer System.out Element 16: ":ATTRIBUTE" "text/xml" dlen=11 Data 16: <ATTRIBUTE>
MotaSensora.ProxiedPeer System.out Element 17: ":DATA" "text/xml" dlen=8 Data 17: ACCURACY
MotaSensora.ProxiedPeer System.out Element 18: ":ATTRIBUTE" "text/xml" dlen=12 Data 18: </ATTRIBUTE>
MotaSensora.ProxiedPeer System.out Element 19: ":VALUE" "text/xml" dlen=7 Data 19: <VALUE>
MotaSensora.ProxiedPeer System.out Element 20: ":DATA" "text/xml" dlen=3 Data 20: 0.5
MotaSensora.ProxiedPeer System.out Element 21: ":VALUE" "text/xml" dlen=8 Data 21: </VALUE>
MotaSensora.ProxiedPeer System.out Element 22: ":MSG_REQUEST" "text/xml" dlen=14 Data 22: </MSG_REQUEST>
MotaSensora.ProxiedPeer System.out Sending message to Pipe: urn:jxta:uuid-59616261646162614R50472050325033D0 0
A9DB59310E4F5A82AFDEADB9878C6904

```

La mota sensora rep la petició de modificació, la processa i envia el missatge de resposta a l'aplicació de monitorització

[MSINFO] 21:49:45,671 - Missatge rebut:

```
<?xml version="1.0"?>
<!DOCTYPE MSG_REQUEST>
<MSG_REQUEST>
  <MESSAGE_ID>
    Rqst_ACCURACY_20121130204920411
  </MESSAGE_ID>
  <MSG_TYPE>
    UPDATE
  </MSG_TYPE>
  <SCOPE>
    SENSOR
  </SCOPE>
  <SENSOR_ID>
    S1
  </SENSOR_ID>
  <ID>
    urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
  </ID>
  <ATTRIBUTE>
    ACCURACY
  </ATTRIBUTE>
  <VALUE>
    0.5
  </VALUE>
</MSG_REQUEST>
```

[MSINFO] 21:49:45,671 - findAdvertisement - Buscant advertisements...

[MSINFO] 21:49:45,718 - S'han trobat advertisements...

[MSINFO] 21:49:45,718 - És Pipe Advertisements...

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jxta:PipeAdvertisement>
<jxta:PipeAdvertisement xml:space="default" xmlns:jxta="http://jxta.org">
  <Id>
    urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
  </Id>
  <Type>
    JxtaPropagate
  </Type>
  <Name>
    ProxiedPeer1
  </Name>
</jxta:PipeAdvertisement>
```

[MSINFO] 21:49:45,718 - Output Pipe created: urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104



El missatge de resposta indica que l'execució ha estat correcta.

```
[MSINFO] 21:49:45,718 - Output Pipe created: urn:jxta:uuid-59616261646162614E504720503250335069635368614265AD5B86696C65436104
[MSINFO] 21:49:45,734 - Missatge resposta:
<?xml version="1.0"?>
<!DOCTYPE MSG_RESPONSE>
<MSG_RESPONSE>
  <MESSAGE_ID>
    Rqst_ACCURACY_20121130204920411
  </MESSAGE_ID>
  <VALUE>
    0.5
  </VALUE>
  <EXECUTION>
    OK
  </EXECUTION>
</MSG_RESPONSE>
```

El missatge de resposta arriba al Proxied Peer qui coneix que la petició que ha realitzat ha estat exitosa.

```
MotaSensora.ProxiedPeer System.out Con Length: 1207
MotaSensora.ProxiedPeer System.out 0 <?xml version="1.0"?>
MotaSensora.ProxiedPeer System.out <!DOCTYPE MSG_RESPONSE>
MotaSensora.ProxiedPeer System.out <MSG_RESPONSE>
MotaSensora.ProxiedPeer System.out □<MESSAGE_ID>
MotaSensora.ProxiedPeer System.out □Rqst_ACCURACY_20121130204920411
MotaSensora.ProxiedPeer System.out □</MESSAGE_ID>
MotaSensora.ProxiedPeer System.out □<VALUE>
MotaSensora.ProxiedPeer System.out □0.5
MotaSensora.ProxiedPeer System.out □</VALUE>
MotaSensora.ProxiedPeer System.out □<EXECUTION>
MotaSensora.ProxiedPeer System.out □OK
MotaSensora.ProxiedPeer System.out □</EXECUTION>
MotaSensora.ProxiedPeer System.out </MSG_RESPONSE>
MotaSensora.ProxiedPeer System.out 1 LimitedRangeRdvMessage <?xml version="1.0" encoding="UTF-8"?>
MotaSensora.ProxiedPeer System.out <!DOCTYPE jxta:LimitedRangeRdvMessage>
MotaSensora.ProxiedPeer System.out <jxta:LimitedRangeRdvMessage xmlns:jxta="http://jxta.org">
MotaSensora.ProxiedPeer System.out □<TTL>
MotaSensora.ProxiedPeer System.out □02
MotaSensora.ProxiedPeer System.out □</TTL>
```

Un cop es rep la confirmació de que s'ha produït la actualització, es mostra el missatge de confirmació a la interfície gràfica, tal i com hem indicat a l'inici.

