SEPSen: Semantic Event Processing at the Sensor Node for Energy Efficient Wireless Sensor Networks

Mumraiz K Kasi, Annika Hinze, Catherine Legg and Steve Jones
University of Waikato, New Zealand

DEBS 2012 Berlin

Low-power consumption environmental monitoring system

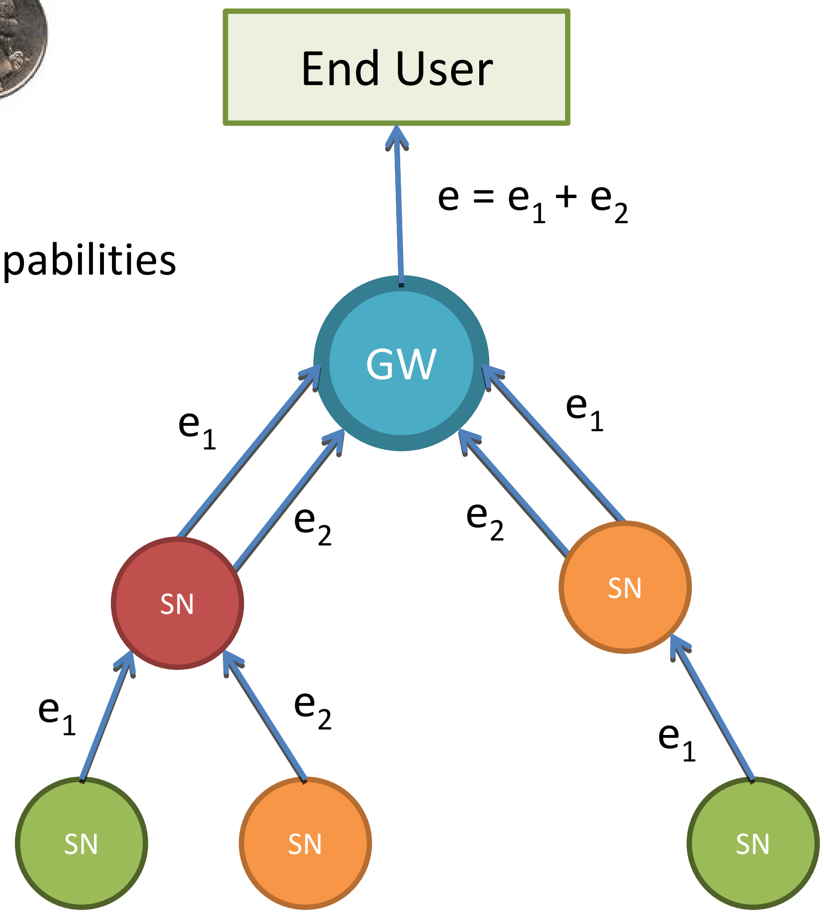
Wireless Sensor Networks (WSN's)

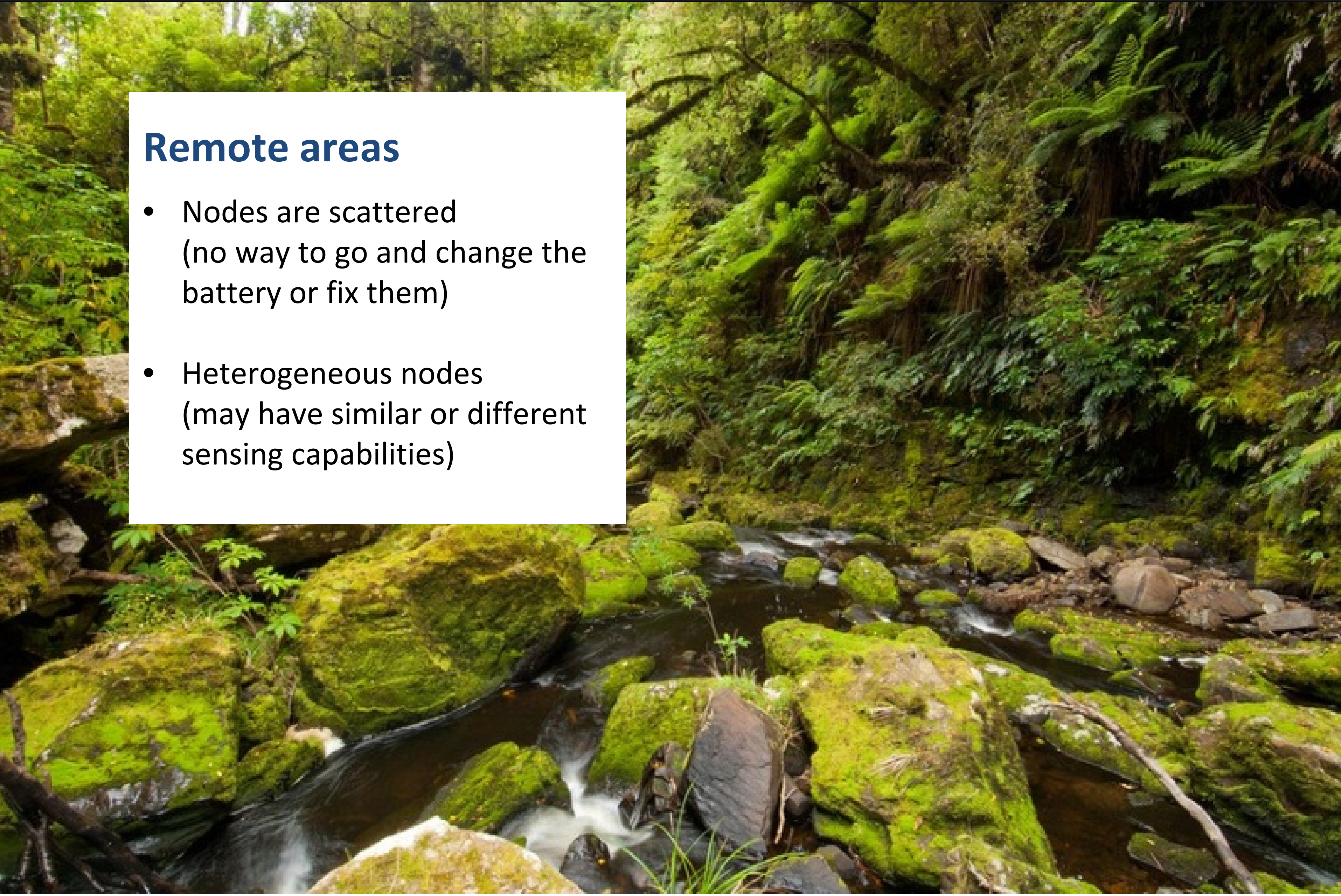
Sensor Nodes

- Large number (100-1000s)
- Sensing, processing & communication capabilities
- Collaborate with each other

Gateway Nodes

- More powerful
- Processes the data
- Relays data from network to end user





Remote areas

- Nodes are scattered
  (no way to go and change the battery or fix them)

- Heterogeneous nodes
  (may have similar or different sensing capabilities)

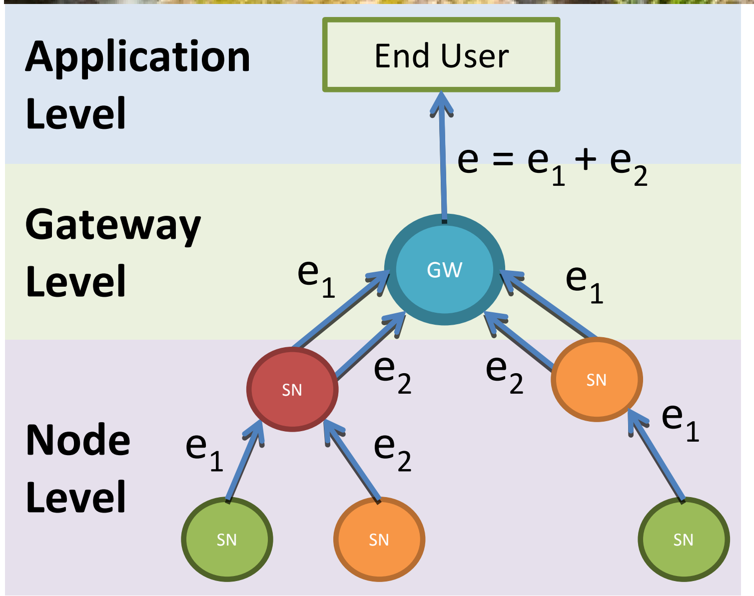
Data Integration in WSNs

User interested in higher level events

- Example: interest in the occurrence of a specific pollutant combination
- Needs data integration from heterogeneous nodes

Typical approach…

- Sensor nodes used for capturing only
- Processing at upper levels (GWs)
- Wastes energy at the sensor nodes



Challenges in WSN's

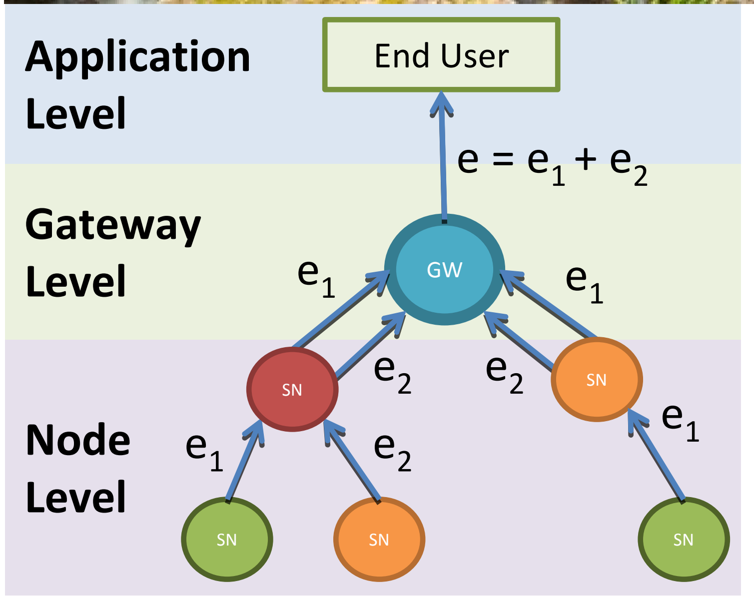
Solution	Existing work predominantly…
In-network Data Processing	**Homogeneous sensor nodes**
Ontologies	**Processing at the gateway nodes**
Context-Awareness	**Limited to infrastructure context**

Proposed Solution

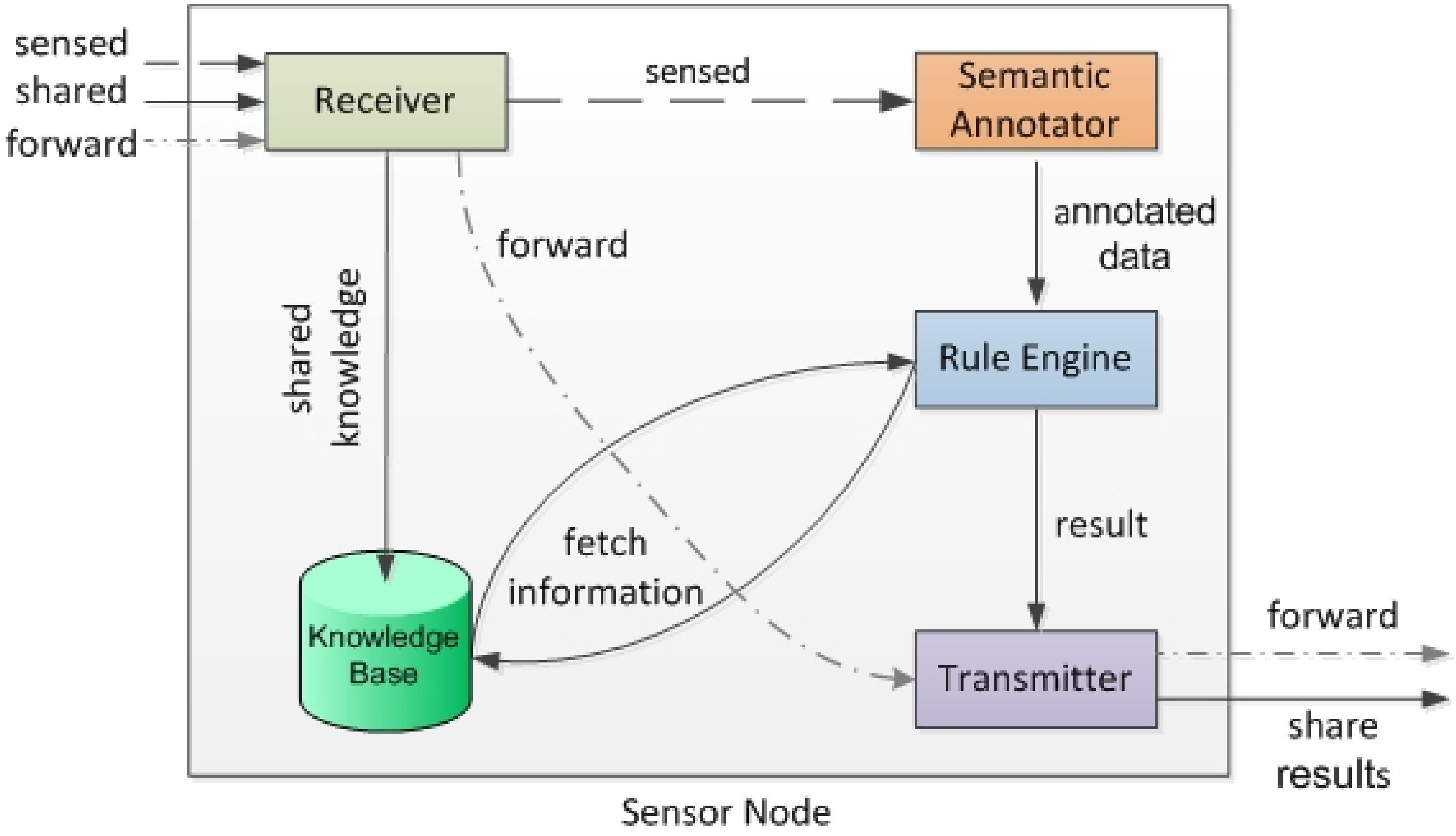
SEPSen: Processes data locally at the sensor node level

At sensor nodes level:

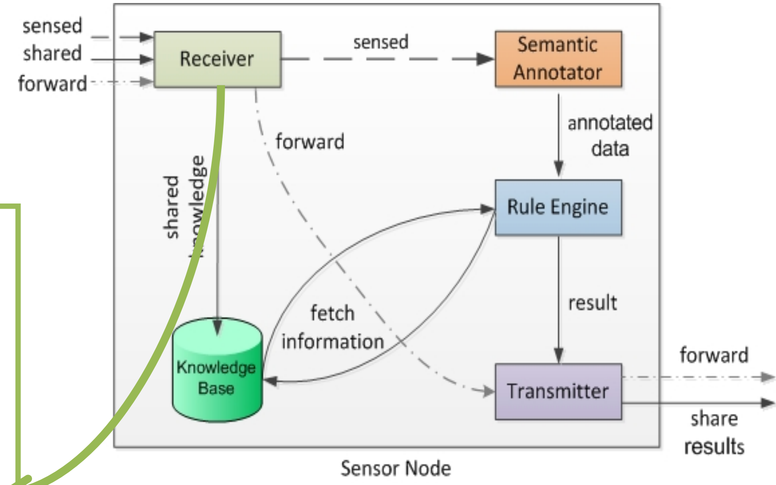
- semantically annotate sensed data
- collaborate with surrounding sensor nodes
- apply rules on the gathered knowledge.



Sensor Node Architecture



Overview of SEPSen:
Receiver



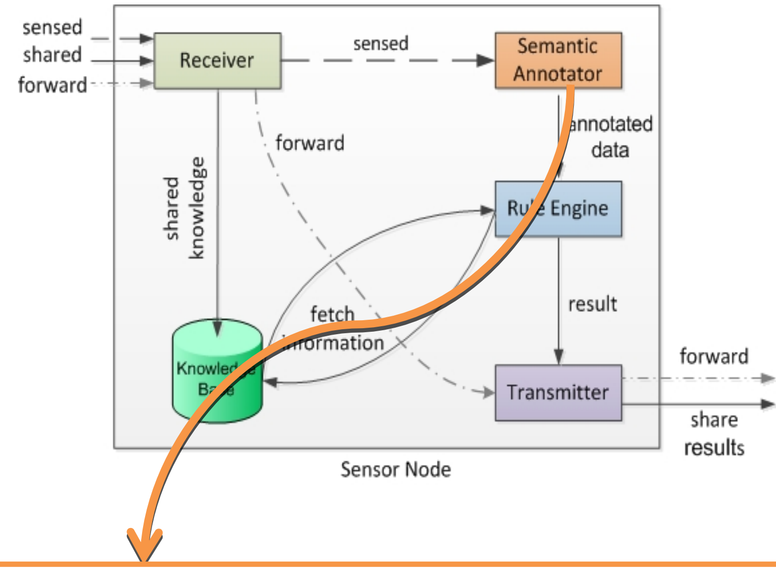
Distinguishes three types of incoming events:

- Sensed event
- Shared event (for composition)
- Forwarded event (communication)

Overview of SEPSen:
Semantic Annotator

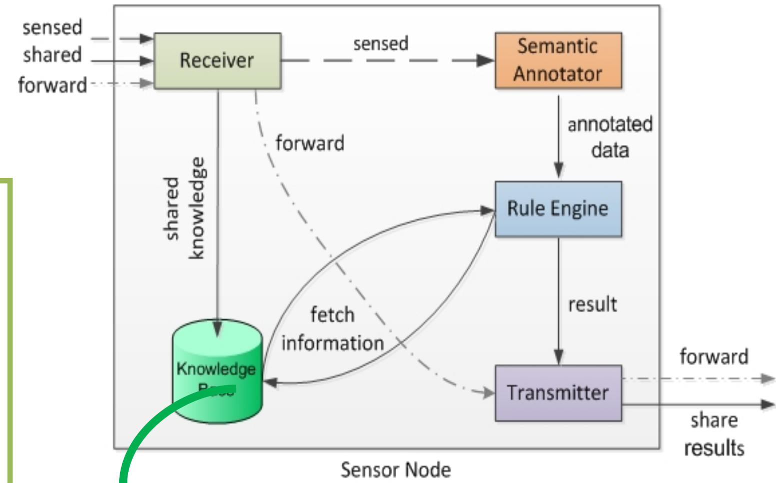
Ontology fragment available at WaterpH sensor node

```
@prefix ont:<http://www.co-ode.org/ontologies/ont.owl#>
@prefix owl:<http://www.w3.org/2002/07/owl#>
@prefix rdfs:<http://www.w3.org/2000/01/rdf-schema#>

ont:MeasurementSite rdf:type owl:Class;

ont:ObservedProperty rdf:type owl:Class;

ont:Measurement rdf:type owl:Class;

ont:WaterMeasurement rdf:type owl:Class;
        rdfs:subClassOf ont:Measurement.

ont:WaterProperty rdf:type owl:Class;
        rdfs:subClassOf ont:ObservedProperty.

ont:WaterPH rdf:type owl:Class;
        rdfs:subClassOf ont:WaterProperty.

ont:hasProperty rdf:type owl:ObjectProperty;
        rdfs:domain ont:MeasurementSite;
        rdfs:range ont:WaterProperty.

ont:hasMeasurement rdf:type owl:ObjectProperty;
        rdfs:domain ont:MeasurementSite;
        rdfs:range ont:WaterMeasurement.

ont:hasValue rdf:type owl:DatatypeProperty;
        rdfs:domain ont:WaterProperty;
        rdfs:range xsd:double.
```



- Relevant ontology fragments in each node
- Events are semantically annotated
- Main ontology at GW for overall application.
- Current status: Manual ontology infusion

Overview of SEPSen:
Knowledge Base

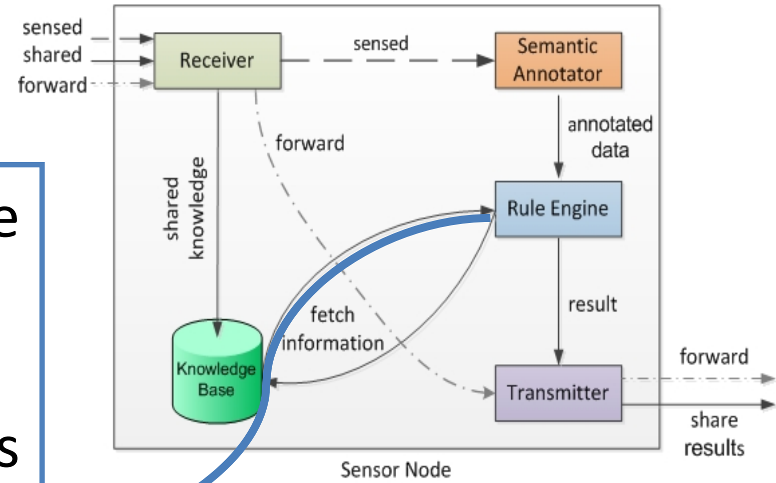


The Knowledge Base of a sensor node contains:

Facts base:
- event data recorded by sensors
- changing at runtime

Rules base:
- semantic rules (IF-THEN)
- may be updated over time

Overview of SEPSen:
Rule Engine



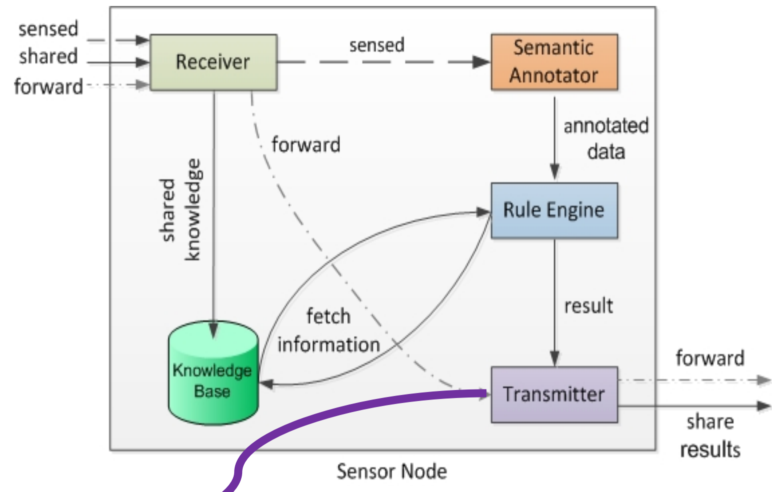
Rule Engine based on the RETE algorithm [3].

Pattern network of nodes that encodes the condition parts (IF-parts) of the rules.

Triggered actions:
- Discard (non-matching)
- Share (partial match)
- Forward (complete match)

Overview of SEPSen: Transmitter

- Transmission to gateway and other relevant sensor nodes.
- Designed to use the context information (e.g., node energy levels) of the sensor network for routing decisions.
- Current status: context information not utilised yet



Evaluation: Simulation

Setup:

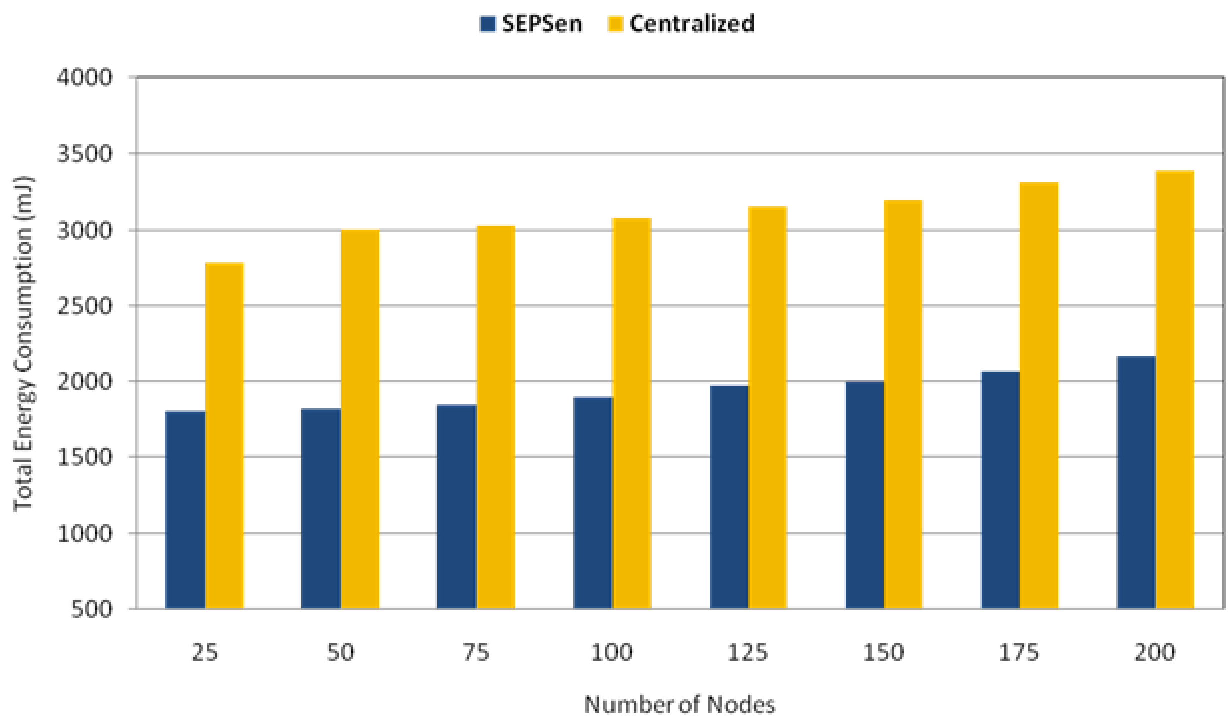
- Implementation of SEPSen and water pollutant application
- Simulations using PowerTOSSIM [16].
- Energy model based on Mica2 sensor nodes.
- Average over 60 seconds, one message per second.

Evaluation Goal:

- Performance comparison between semantic in-network processing and centralized approach (processing at gateway nodes) [cf. cost model (DEBS'11)]

Evaluation

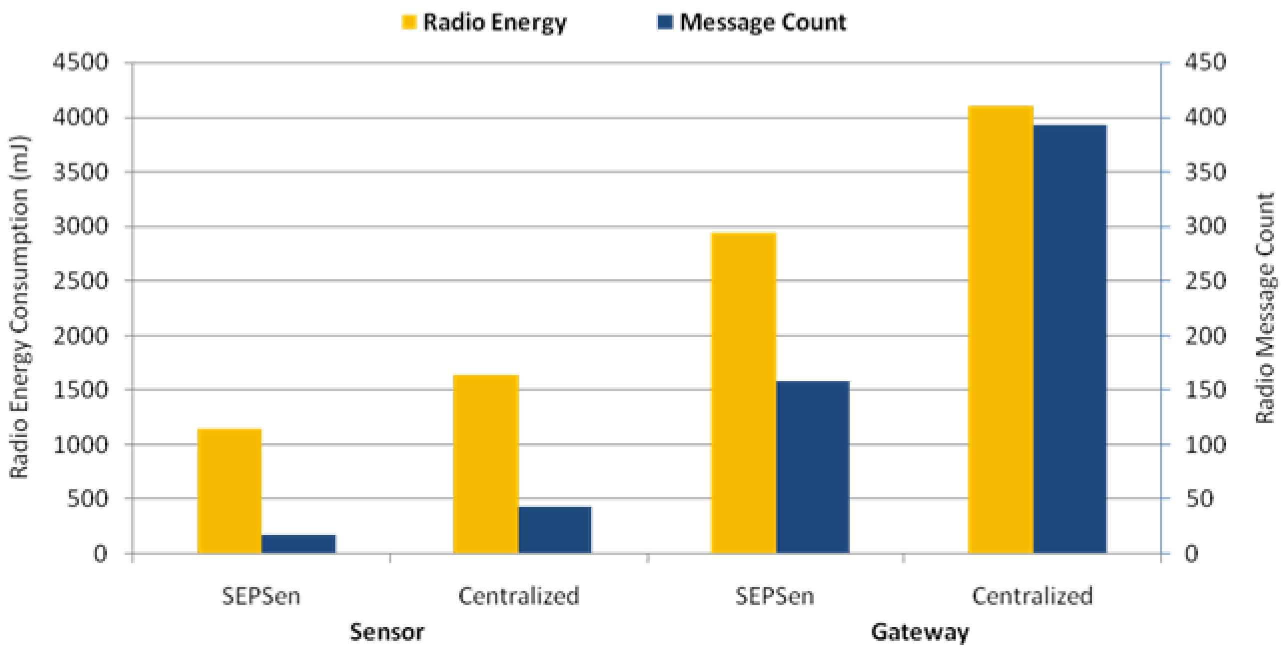
Test-1: Total Energy Consumption



Energy conservation from *13% for 25 nodes* to *16% for 200 nodes.*

Evaluation

Test-2: Radio Energy & Radio Message Counts



Nodes in the centralized architecture sent nearly *three times as many messages* as in SEPSen.

Conclusion

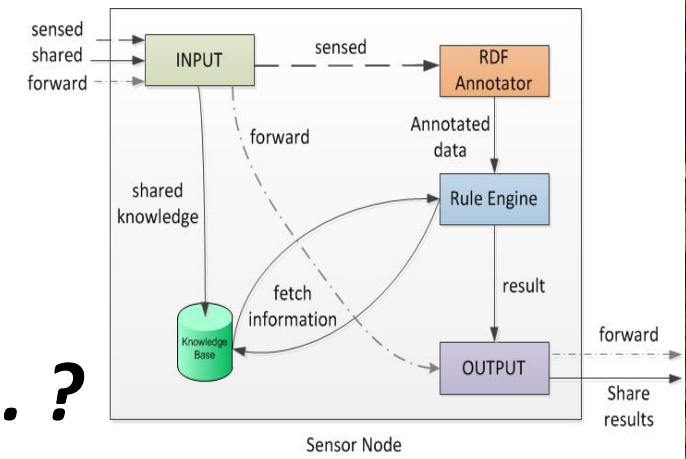
- Unpredictable, heterogeneous WSN “in the wild”.
- Processing at gateway nodes reduces WSN lifetime.

SEPSen

- uses processing capabilities of sensor nodes (in-network)
- aims for energy conservation in WSNs.
- simulation results promising

Next steps:

- automate Ontology infusion
- evaluate variation of complex events
- context-awareness at the sensor node level



Thank you. Questions ... ?

Contact:

Mumraiz Kasi, Annika Hinze

mk218, hinze@waikato.ac.nz

The University of Waikato
School of Science & Engineering
Private Bag 3105
Hamilton, New Zealand
info.sci.waikato.ac.nz

THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

References

[1] Y. Chen, H. V. Leong, M. Xu, J. Cao, K. C. C. Chan, and A. T. S. Chan. In-network data processing for wireless sensor networks. In Mobile Data Management (MDM’6), 2006.

[2] K. W. Chow and Q. Li. Issdm: an in-network semantic sensor data model. In Proc. of the Symposium on Applied computing, SAC ’07, pages 959–960, 2007.

[3] C. L. Forgy. Expert systems. chapter Rete: a fast algorithm for the many pattern/many object pattern match problem. IEEE press, 1990.

[4] L. Gu, D. Jia, and . Vicaire, Pascal et a. Lightweight detection and classification for wireless sensor networks in realistic environments. In SenSys’05, 2005.

[5] T. He, S. Krishnamurthy, J. A. Stankovic, and A. et al. Energy-efficient surveillance system using wireless sensor networks. In MobiSys ’04, 2004.

[6] A. Hinze, K. Sachs, and A. Buchmann. Event-based applications and enabling technologies. In Distributed Event-Based Systems (DEBS ’09), 2009.

References

[7] Q. Huaifeng and Z. Xingshe. Integrating context aware with sensornet. In Proc. of International Conference on Semantics, Knowledge and Grid, SKG ’05, page 83, 2005.

[8] V. Huang and M. K. Javed. Semantic sensor information description and processing. In Sensor Technologies and Applications, 2008.

[9] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: a scalable and robust communication paradigm for sensor networks. In MobiCom ’00, 2000.

[10] M. K. Kasi and A. Hinze. Cost analysis for complex in-network event processing in heterogeneous wireless sensor networks. In DEBS ’11, 2011.

[11] P. Levis, N. Lee, M. Welsh, and D. Culler. Tossim: accurate and scalable simulation of entire tinyos applications. In SenSys ’03, 2003.

[12] P. Levis, S. Madden, J. Polastre, R. Szewczyk, and A. W. et al. TinyOS: an operating system for sensor networks. In Ambient Intelligence. Springer Verlag, 2004.

References

[13] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong. Tag: a tiny aggregation service for ad-hoc sensor networks. SIGOPS Oper. Syst. Rev., 36:131–146, 2002.

[14] L. M. Ni, Y. Zhu, J. Ma, M. Li, Q. Luo, Y. Liu, S. C. Cheung, and Q. Yang. Semantic sensor net: An extensible framework. In ICCNMC, pages 1144–1153, 2005.

[15] A. P. Sheth. Changing focus on interoperability in information systems: From system, syntax, structure to semantics. In M. G. et al., editor, Interoperating Geographic Information Systems. Kluwer Academic Publishers, 1999.

[16] V. Shnayder, M. Hempstead, B.-r. Chen, G. W. Allen, and M. Welsh. Simulating the power consumption of large-scale sensor network applications. In SenSys ’04. ACM, 2004.

[17] A. Silberstein, R. Braynard, G. Filpus, G. Puggioni, A. Gelfand, K. Munagala, J.Yang, B.Kamachari, D.Estrin, and S. Wicker. Data driven processing in sensor networks. In Innovative Data Systems Research (CIDR ’07), 2007.

References

[18] X. Wang and J. Li. Precision constraint data aggregation for dynamic cluster-based wireless sensor networks. In Mobile Ad-hoc and Sensor Networks (MSN ’09), 2009.

[19] A. Wun, M. Petrovi, and H.-A. Jacobsen. A system for semantic data fusion in sensor networks. In Distributed Event-Based Systems(DEBS ’07), 2007.

The MICA mote uses an Atmel ATmega 128L processor running at 4 megahertz. The 128L is an 8-bit microcontroller that has 128 kilobytes of onboard flash memory to store the mote's program. This CPU is about as powerful as the 8088 CPU found in the original IBM PC (circa 1982).

This low power consumption allows a MICA mote to run for more than a year with two AA batteries.

MICA motes come with 512 kilobytes of flash memory to hold data. They also have a 10-bit A/D converter so that sensor data can be digitized. Sensors available include temperature, acceleration, light, sound and magnetic.

The final component of a MICA mote is the radio. It has a range of several hundred feet and can transmit approximately 40,000 bits per second. When it is off, the radio consumes less than one microamp. When receiving data, it consumes 10 milliamps. When transmitting, it consumes 25 milliamps. Conserving radio power is key to long battery life.

Software on MICA motes is built on an operating system called TinyOS. TinyOS is helpful because it deals with the radio and power management systems for you and makes it much easier to write software for the mote.