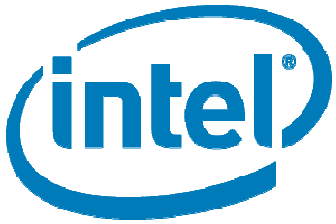


Worldwide Sensor Web: WWW + Sensors?

Phillip B. Gibbons

Intel Research Pittsburgh

Worldwide Sensor Web

- **Planet-wide in-situ data collection: live & historical**
  - collect & store in parallel near source
- **Push data processing & filtering to sensors**
  - reduce the raw data to derived info, in parallel near source
- **Real-time adaptation of collection & processing**
  - data-content-triggered, model-based, event-driven, live
- **Data as a single queriable unit**
  - global queries, aggregating queries, non-local triggering
- **Queries posed anywhere on planet**
  - but exploit any locality
- **Robustness**

A Worldwide Sensor Web is a Distributed Systems Nightmare

- Write-intensive, time-critical, high volume, heterogeneous, subject to harsh conditions, mobile, actuation
- Continuous/Global/aggregating queries (macroscope)

- Programmed by many developers for many applications
- …dynamically changing mix
- …sensors shared by many applications

Want to hide complexity from application developers

PERVASIVE computing
OCTOBER–DECEMBER 2003

IrisNet: An Architecture for a Worldwide Sensor Web

Today's common computing hardware—Internet connected desktop PCs and inexpensive, commodity off-the-shelf sensors such as Webcams—is an ideal platform for a worldwide sensor web. IrisNet provides a software infrastructure for this platform that lets users query globally distributed collections of high-bit-rate sensors powerfully and efficiently.

Wide-area architectures for pervasive sensing will enable a new generation of powerful distributed sensing services. Such architectures have so far received little attention but are increasingly relevant because of a confluence of technological trends. Commodity off-the-shelf sensors—including video cameras (Webcams), microphones, and motion services on this distributed infrastructure.

Imagine the following scenario: after an oil spill, an ecologist wishes to know all locations where oil has significantly encroached on the coastal habitat. She queries a coastal-monitoring service, which collects data from video cameras directed at the coastline. In response, she receives both images of these contaminated sites and their geographic locations. The same coastal-moni-

IrisNet

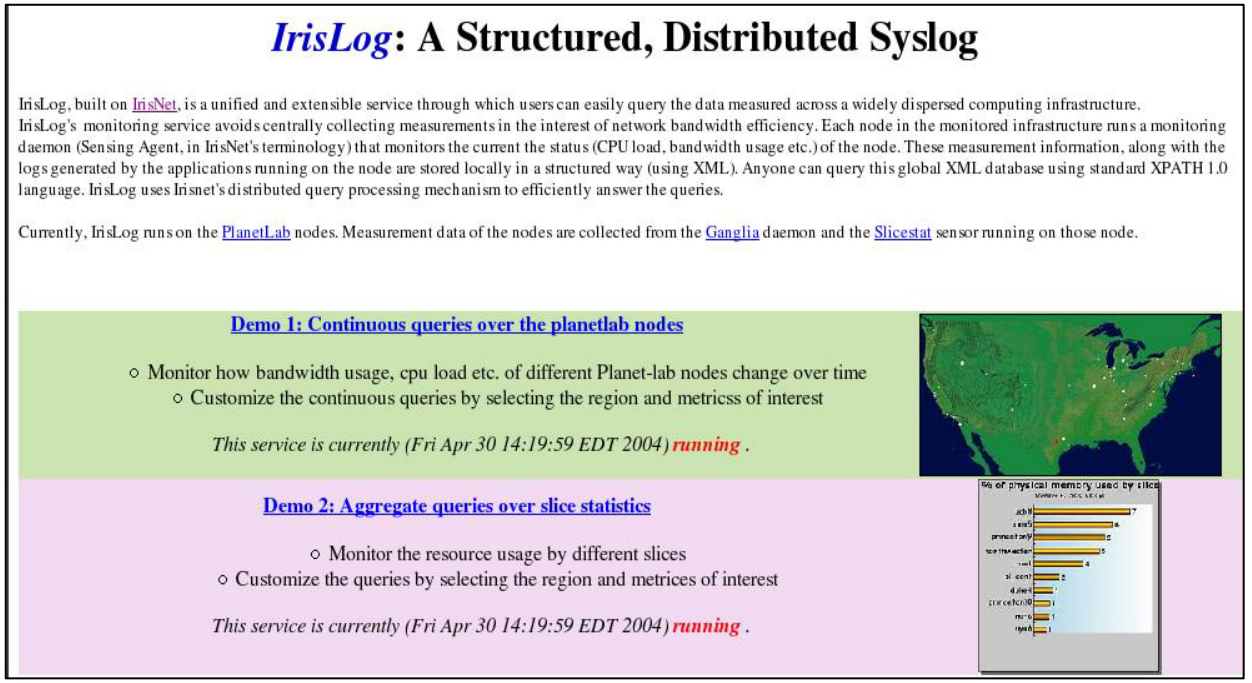
- Exports centralized view of database to queries
  - Distributed query processing behind the scenes
- Provides common runtime environment for sensor feed processing

Suman Nath's dissertation

IrisNet on PlanetLab

We deployed IrisNet on PlanetLab network

- 450+ nodes, 5 continents
- "Sensor" feeds are CPU, memory & network measurements at each node



***IrisLog*: A Structured, Distributed Syslog**

IrisLog, built on IrisNet, is a unified and extensible service through which users can easily query the data measured across a widely dispersed computing infrastructure. IrisLog's monitoring service avoids centrally collecting measurements in the interest of network bandwidth efficiency. Each node in the monitored infrastructure runs a monitoring daemon (Sensing Agent, in IrisNet's terminology) that monitors the current the status (CPU load, bandwidth usage etc.) of the node. These measurement information, along with the logs generated by the applications running on the node are stored locally in a structured way (using XML). Anyone can query this global XML database using standard XPATH 1.0 language. IrisLog uses Irisnet's distributed query processing mechanism to efficiently answer the queries.

Currently, IrisLog runs on the PlanetLab nodes. Measurement data of the nodes are collected from the Ganglia daemon and the Slicestat sensor running on those node.

Demo 1: Continuous queries over the planetlab nodes

- Monitor how bandwidth usage, cpu load etc. of different Planet-lab nodes change over time
- Customize the continuous queries by selecting the region and metricss of interest

This service is currently (Fri Apr 30 14:19:59 EDT 2004) ***running*** .

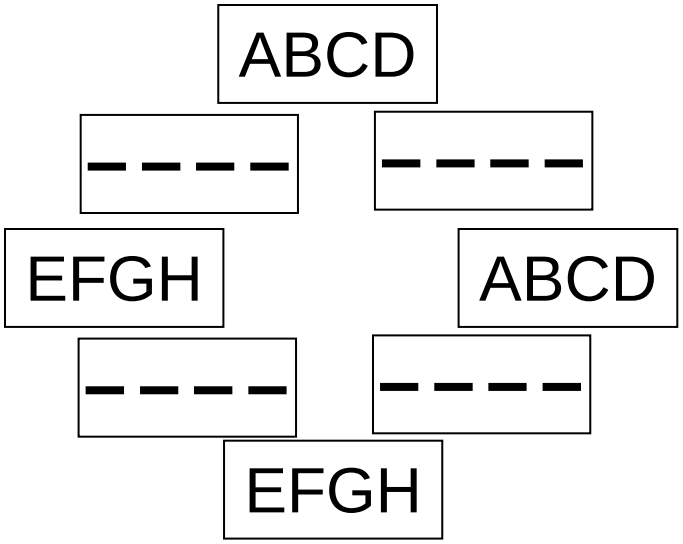
Demo 2: Aggregate queries over slice statistics

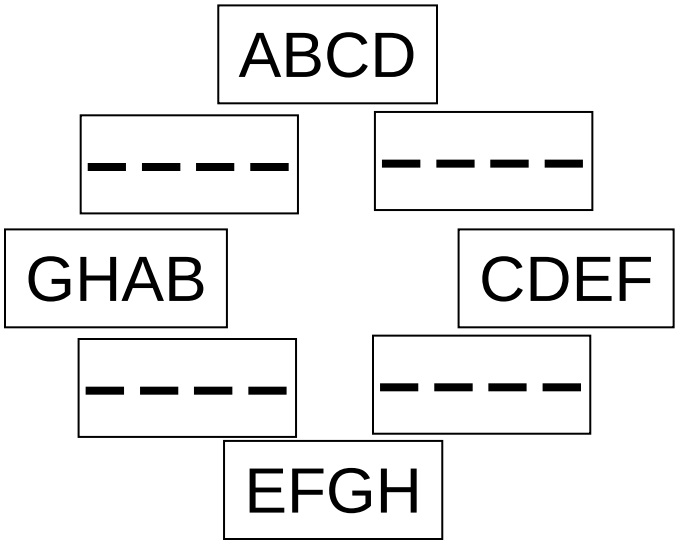
- Monitor the resource usage by different slices
- Customize the queries by selecting the region and metrices of interest

This service is currently (Fri Apr 30 14:19:59 EDT 2004) ***running*** .

Some Lessons Learned from IrisNet's Deployment on PlanetLab

- Robustness, robustness, robustness
- Platform heterogeneity => VM-based?
- Load balancing
- Processes die; connectivity flaky (on re-connect incorporate into running aggregate query)
- Correlated failures:
  - replication scheme matters (used Signed Quorum Systems)
  - subtleties arise (can predict 99% of failure patterns and still not improve availability)
- Importance of relative object assignment





Relative Object Assignment

- Do they give us the same availability? NO, Left better
- How large is the difference? Over 2 nines for TPC-H

Conclusions

- Work on Distributed Systems / Databases is an important part of the agenda
  - Meet functionality & performance demands
  - Robustness
  - Hidden from the app developer
- What are the challenges in a worldwide sensor web beyond WWW + good centralized sensor platform/database?