

global scale event notification

Bob Briscoe, BT Research

Sep 2004

Acknowledgements: Jon Crowcroft (Uni Cam)
Jane Tateson, Andrea Soppera, Trevor Burbridge (BT)



event notification?

- event:
(the representation of)
some asynchronous occurrence
- asynchronous:
at a time unpredictable to the observer
- occurrence:
change in the state of an object



my point

- the Internet of things depends on widespread event notification handlers



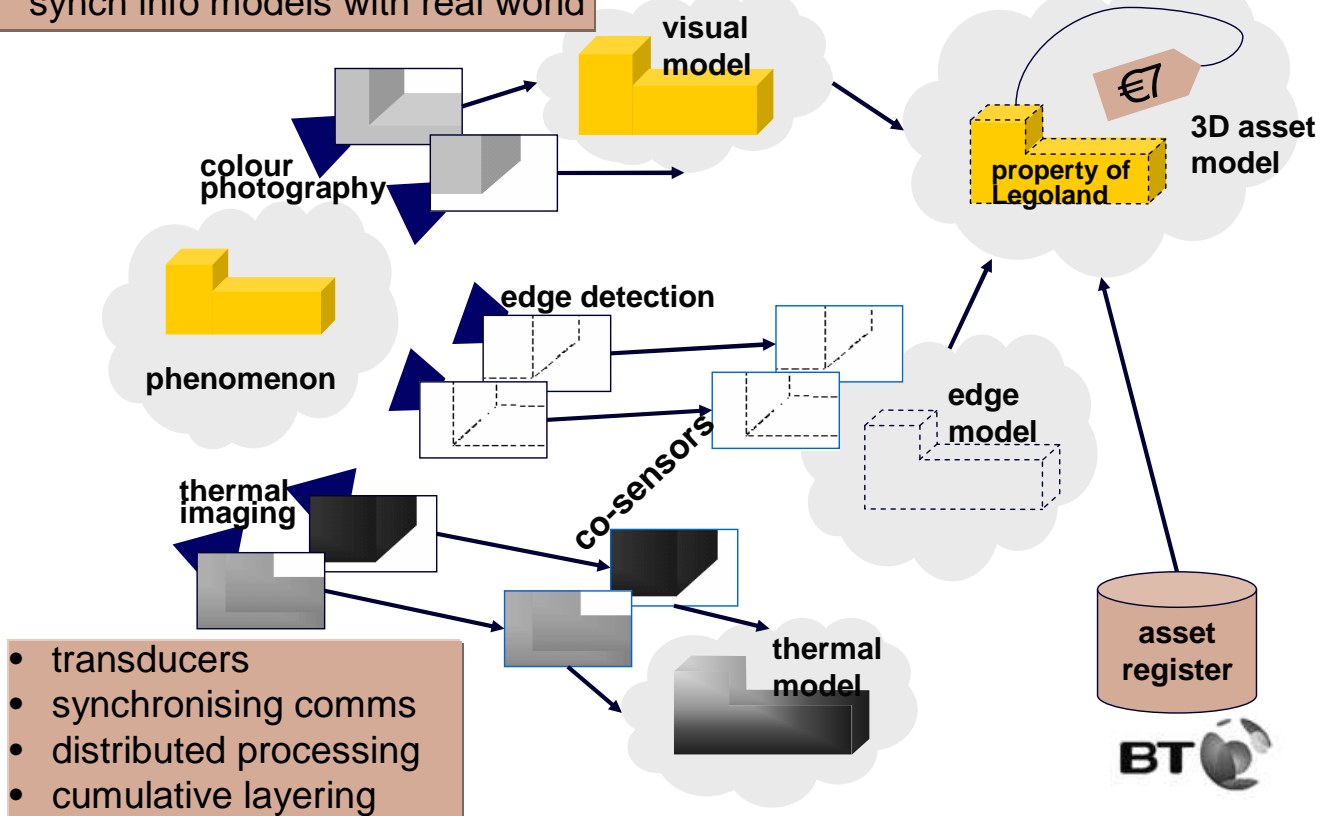
- far from consensus on outstanding hard problems
 - hard to make endpoints reliant only on themselves
 - too onerous for challenged hardware
 - but alternatives require unscalable state in comms infrastructure



conceptual model

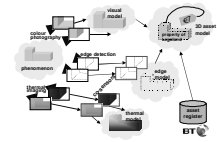
- the hard comms problem
- synch info models with real world

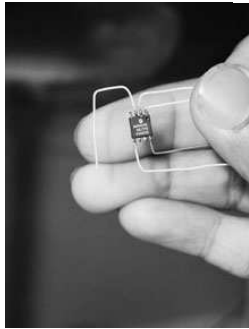
≡ care in the community, home, automotive, supply chain, Internet zero, sensor nets



comms modes

asynchronous communications



- iPic Web server [Shrikumar02]
 - impressive but...
- 
- do we continually ask everything physical to report its state?
 - asynch event notification more applicable for sensors [Shrikumar01]
 - polling never better: not timely, not efficient
 - cascade of event notification over polling loses timeliness

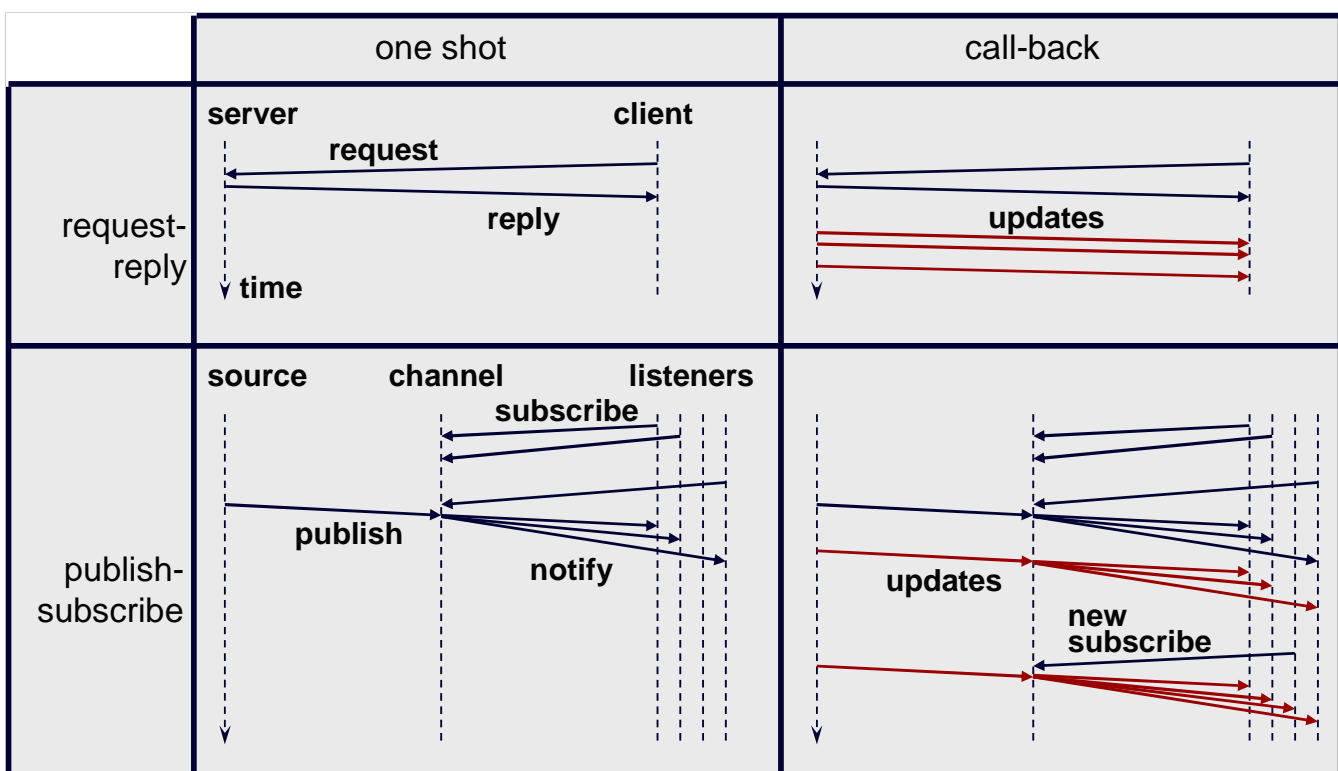
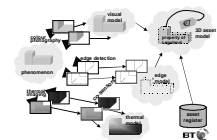
storing & reporting state can be decomposed

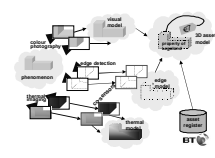
“The iPic demo server is connected by a serial link, which is currently experiencing a load up to its full design capacity...”

Please visit the mirror site below.”



communication modes

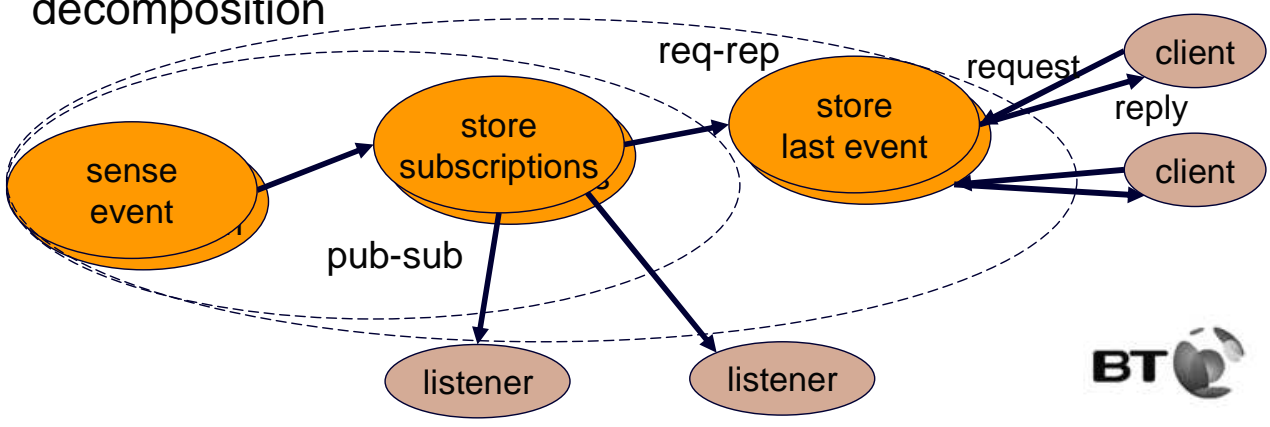




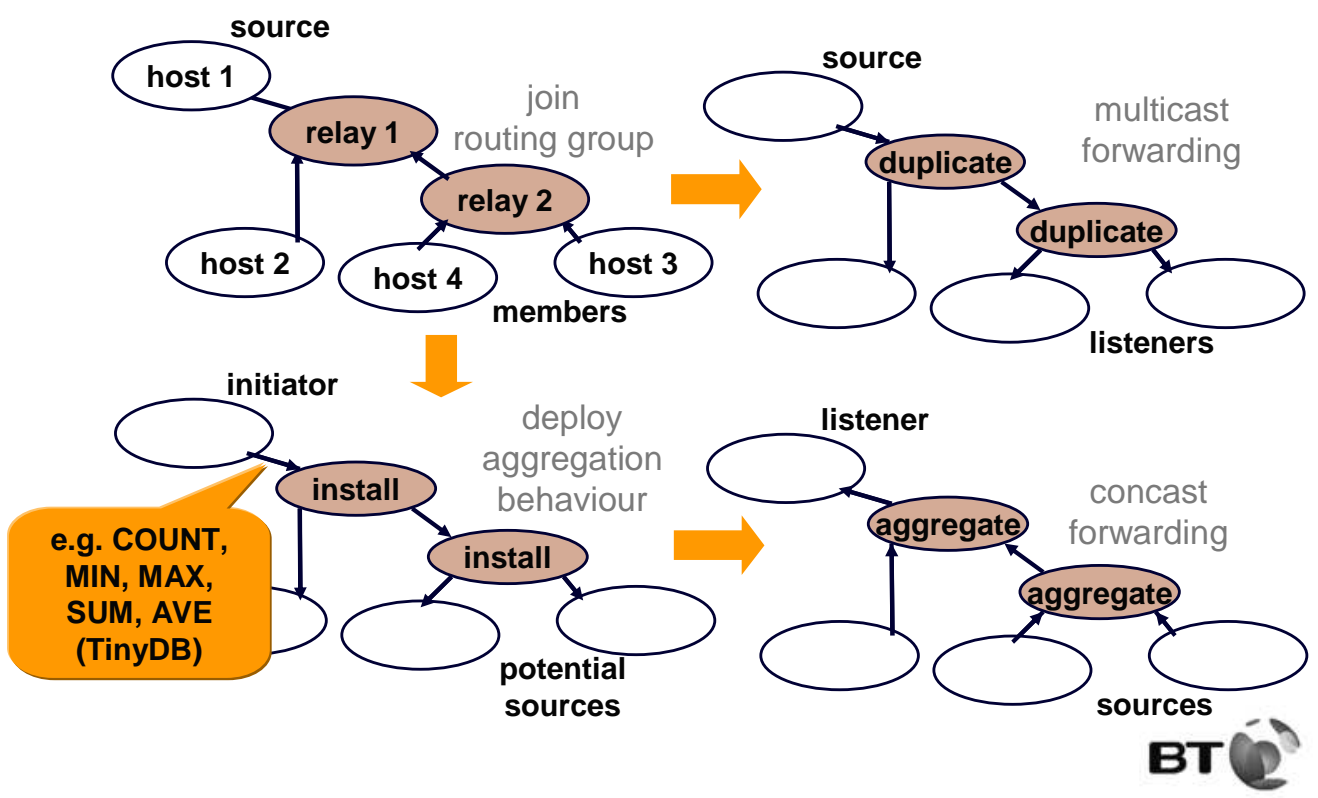
comms modes publish-subscribe

- inherently point to multipoint (group communications)
 - feeds from real world maintain plethora of views of the world
- no need for radio listener at source: power hungry
- but no control over subscription memory demand

decomposition



group formation and forwarding



channelisation problem [Adler01]

- each group's channel requires stored resource
 - *either* distributed group routing tables
 - group routing tree created by receiver interest (app or net layer)
 - each relay stores list of neighbour interest per routing group
 - near-linear complexity: little inherent topological correlation?
 - *or* channel allocations
 - each group in each 'cell' allocated spectrum/timeslot/code/ etc
- if aggregate channel resource
 - must then filter at receiver wasting b/w, interrupts and processing
 - or filter in network (equivalent to channelisation problem)
 - or index-based dynamic creation of groups [Soppera:watchcast]
- creates an economic limit to pervasive computing

everywhere in network between event sources and group interest



the unexpected didn't happen – I think

- if pub-sub, avoid ack → implosion
& sender doesn't know receiver list anyway
- nack preferred (SRM/concast etc avoids implosion)
- rcvr cannot nack asynch msg
 - until receives next in sequence (msec or years later)
- solutions:
 - hop by hop ack [Rowstron01:SCRIBE]
 - e2e index beacon [Soppera:watchcast]
 - note: hop by hop ack doesn't imply e2e delivery (cf TCP)
 - for sensor nets, e2e = across concast & multicast parts



open but closable

- pub-sub has a nice 'business model'
- basic model: open publication of data on a channel
- limit visibility with crypto or scoping of msg routing
 - rights can be changed out of band at run-time
 - can maintain relationship with listeners, which pub-sub hides
- doesn't lock in zero config devices
 - zero config device's packet destination is a neutral 'channel'
 - listeners join channel at run-time to complete msg routing config

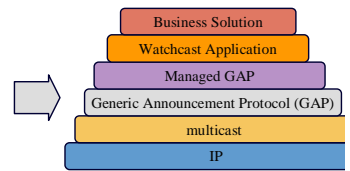


attempts at solutions

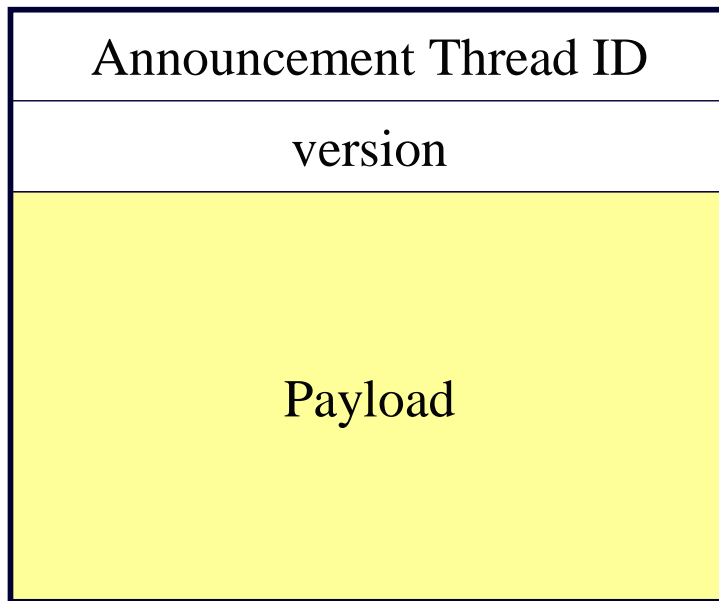
global scale
event notification



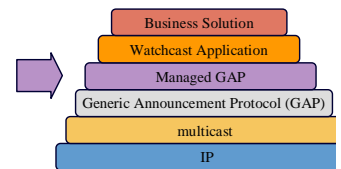
Generic Announcement Protocol



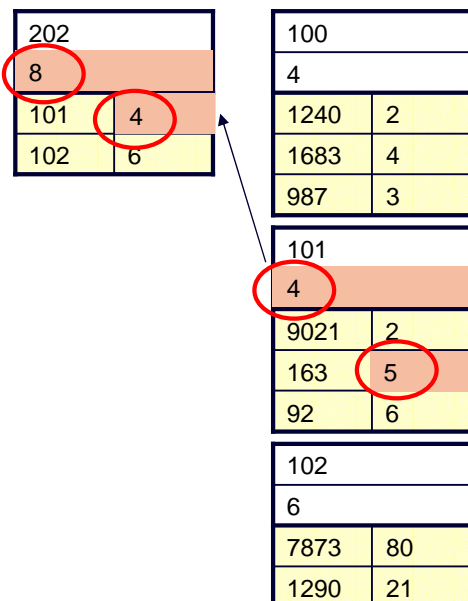
<ath:URL=http://www.hosting.org/AthID?set=farm\$31425>



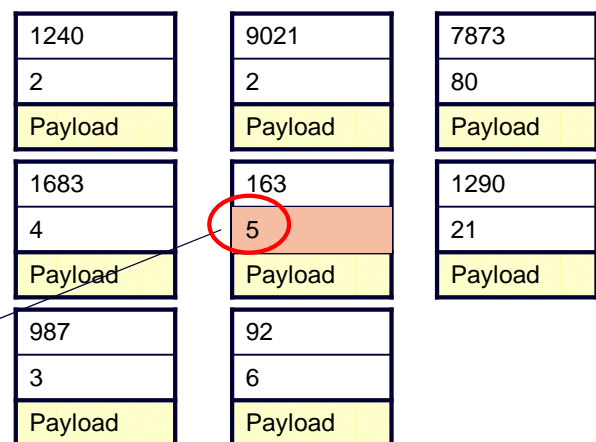
index-based event notification



Index channels



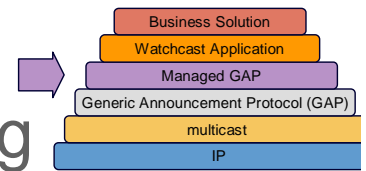
Application channels



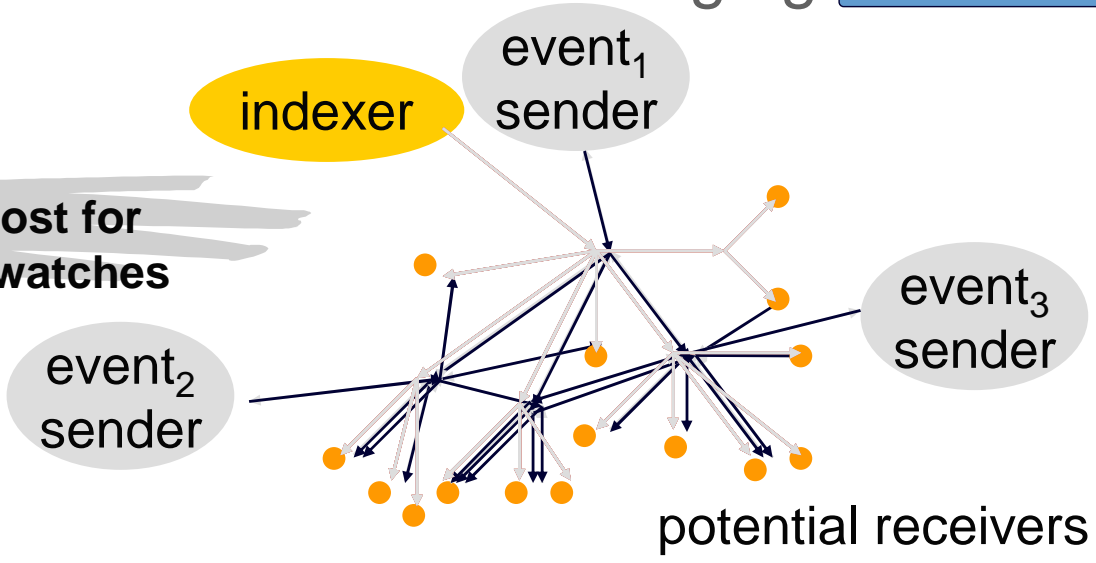
efficient & flexible



index-based event messaging



zero cost for extra watches



multipoint request-reply

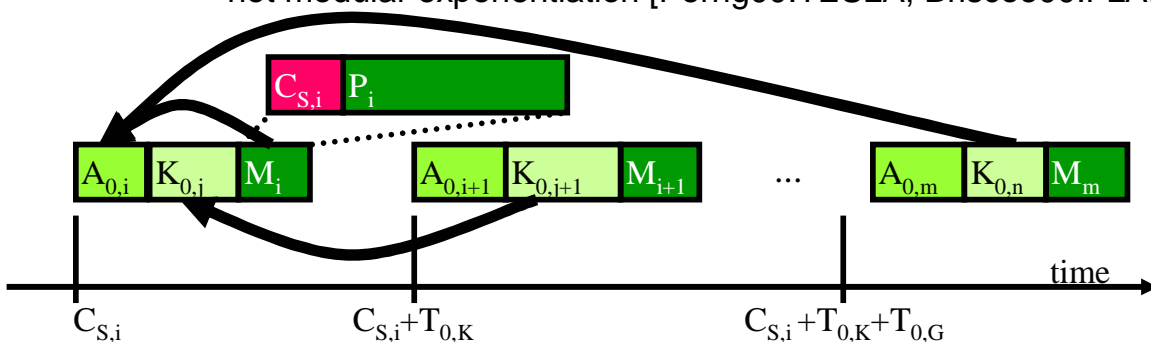
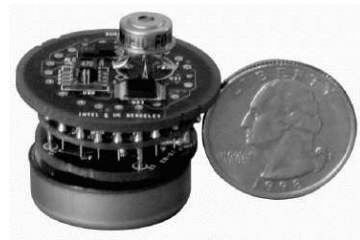
joins (routing)

data (forwarding)



SPINS [Perrig01]

- implemented on Berkeley motes
- group security, not just 1-1
- based on two primitives:
 - SNEP for message encryption
 - μ TESLA for message authentication
 - TESLA derives asymmetry from passage of time, not modular exponentiation [Perrig00:TESLA, Briscoe00:FLAMeS]

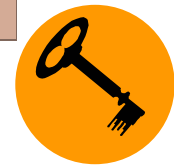


- strong crypto but light processing & msg overhead

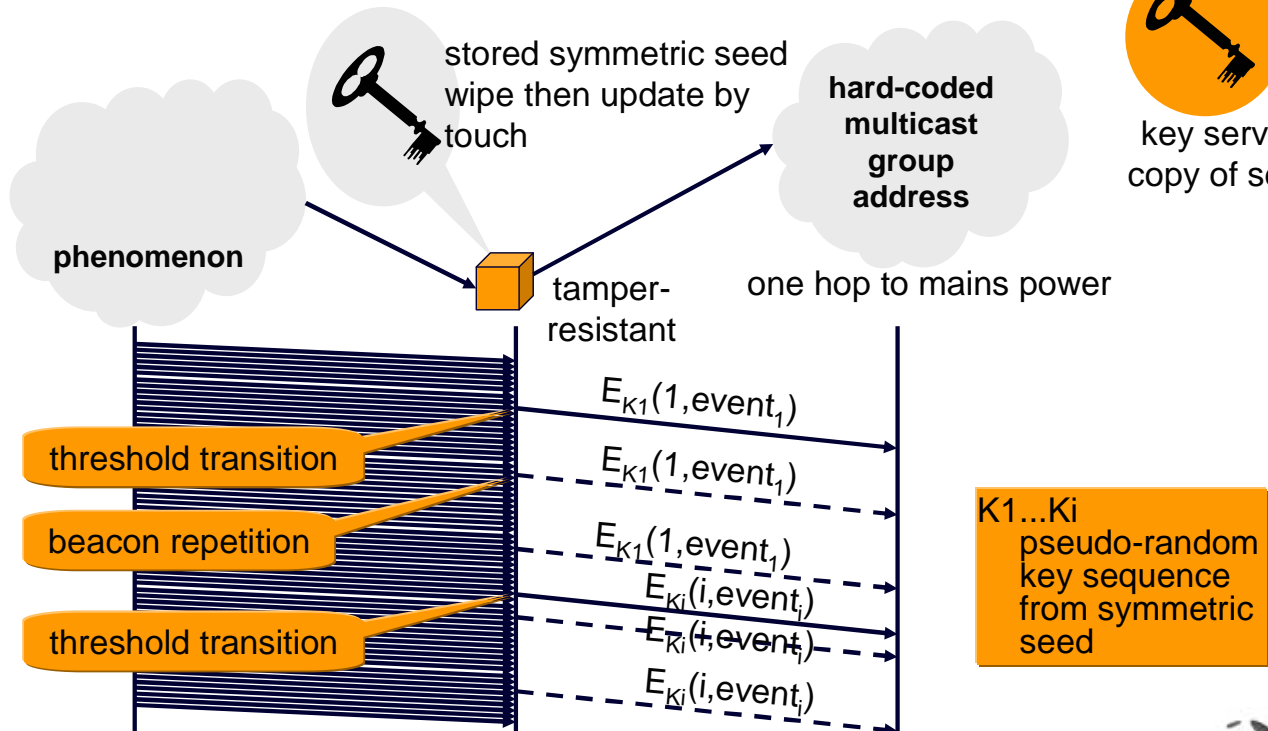


straw man proposal

- design goals
 - zero rcv
 - zero config



key server
copy of seed



prerequisites for Internet of things

- ubiquitous pub-sub
- but also...
 - group creation facilities capable of 10^6 group /sec worldwide
 - infrastructure investment incentives
 - if p2p infrastructure, solve free-riding
 - solve privacy without limiting commercial potential

all our efforts here now
privacy is the gating factor
(what you've seen is 2-4yrs old)



more info

- strange links, ad hoc connectivity creation, routing across sensor databases, addressing events, message traffic profiles, unusual congestion control, security in the wild, key establishment without RSA and more...

Bob Briscoe, "The Implications of Pervasive Computing on Network Design" BT Technology Journal 22 (3) pp. 170--190 URL: <http://www.btexact.com/publications/bttj/bttjissues/> (July, 2004) (but deliberate journal on-line publication delay)

- Bob.Briscoe@bt.com

- questions?



global scale event notification

spare slides

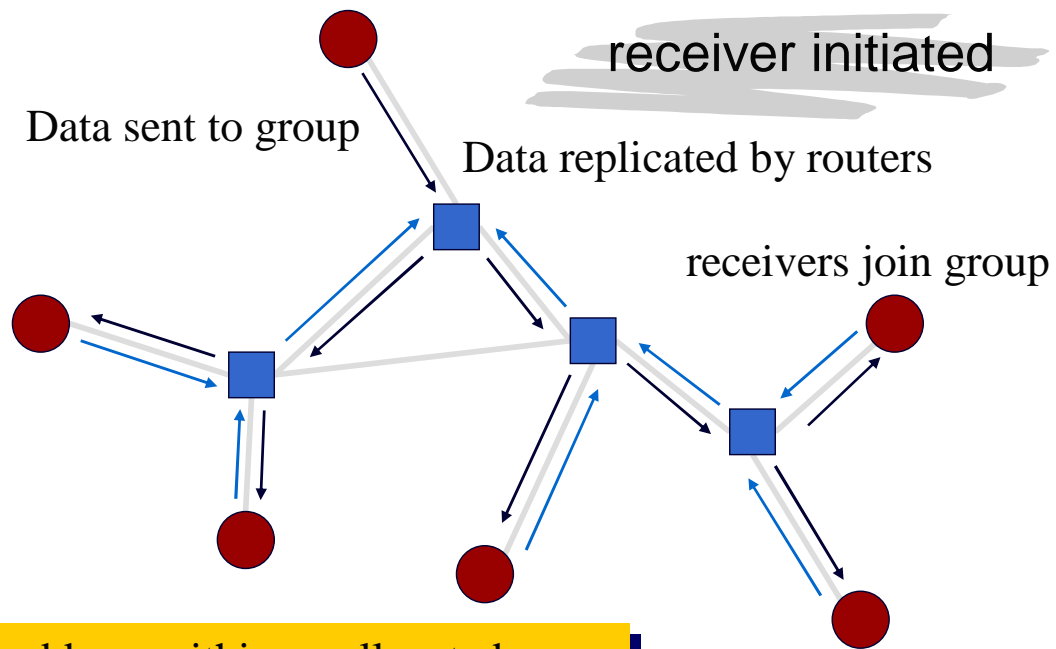
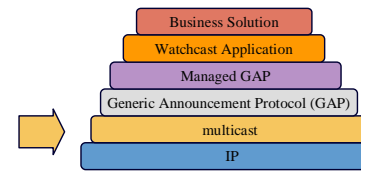


just do it?

- only if we break principles we hold dear
- aren't cheap micro-devices disposable?
- forget research into perfect comms stack?
- No!
 - *device* is disposable
 - its *design* embodies huge investment



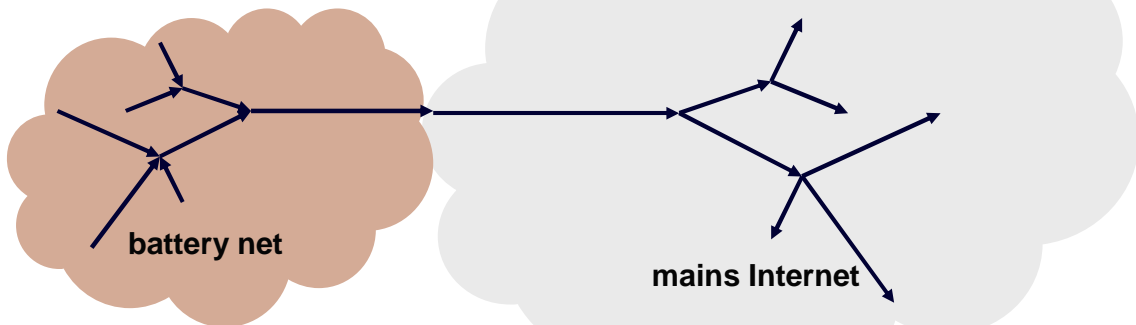
IP Multicast - recap



IP address within an allocated range represents a 'group' not a host



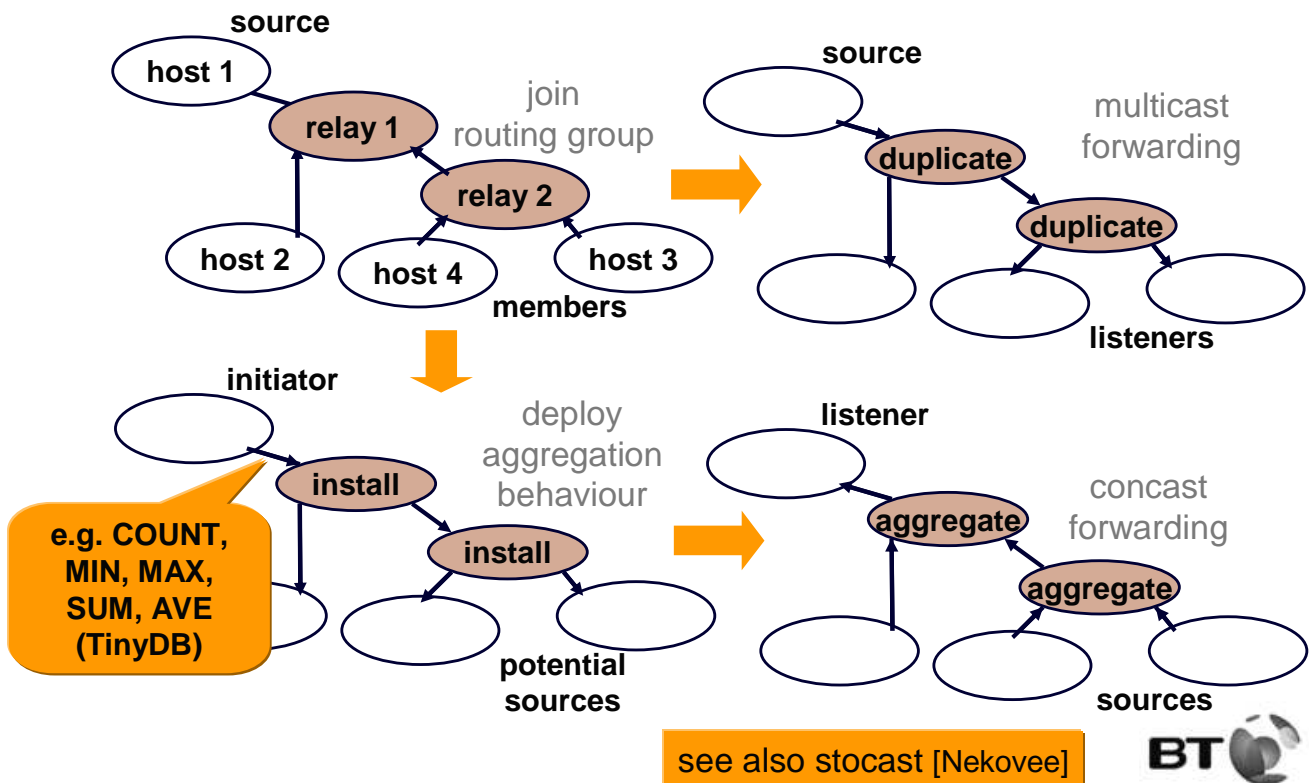
energy constraint reverses rules



- don't multicast until mains
 - minimise message.links
- can do better
 - aggregation of multiple messages (directed diffusion) [Estrin00,01]
 - concast
 - cf generic router assist (GRA) (cisco - generalisation of nack aggregation (PGM))
- receiver initiated multicast
 - normal rules apply
 - but gateway is proxy source (e.g. for re-transmit)
 - relay doesn't need meaning
 - encrypt end to end



group formation and forwarding



connectivity of everything

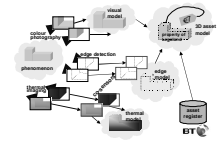
- as a statement of scope
 - otherwise implications on networking uninteresting
- as a statement of recommendation
 - universally present software:
 - TCP/IP, event notification?, higher... HTTP, XML parser?
 - why? if *relative* cost small, *potential* benefit is large
 - TCP/IP cost:
 - 200B code (cf. TinyOS 3.5kB, mote 8kB)
 - memory smaller, cheaper, energy efficient $O(2^{1/D})$
 - processing costs energy
 - headers cost bandwidth esp. IPv6 (header compression helps)
 - potential benefit: $O(n^2)$ [Metcalfe]
 - avoid constraining new uses by locality

n = no. of connectable nodes

D = doubling time



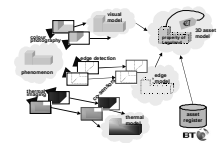
connectivity creation motivating force



- no grand plan for the model of everything
 - mini-models required in their own right
- re-sell for others to build bigger models
 - retail then wholesale
- or open publication?



connectivity creation connectivity by arrangement?



- how did the model's connectivity arise?
 - by arrangement: frequencies, formats, codings, protocols, languages
 - created within another application: discovery and configuration
- classic example:
 - personal digital assistant seeks attractive monitor
 - love at first byte? straight to layer 7 on the first date?
- an alternative
 - cyberspace as chaperone and matchmaker (pre-connected)
 - new requirements on cyberspace
 - proximity model(s)
 - are you a flatscreened Sony or a 53yr-old divorcee from Hounslow in a rain-coat?



proximity awareness

- more advantages
- coverage
- augmented reality

