



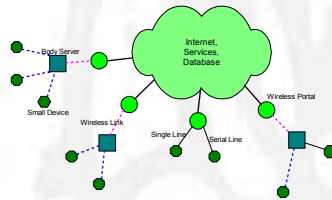
Data Centric Networking for Invisible Computing

Jong Hee Kang, Gaetano Borriello

Invisible Computing

To help user's tasks with low cognitive load on the user
By connecting physical world to virtual world

We have to use zillions of small devices to do that
Devices must not interfere with users



Small Devices

- Short range wireless communication
- RF, IR, Body Area Network
- Intermittent connection
- Mobile
- Frequently turned off for power management
- Limited resources
- Limited computing power
- Limited storage space
- Limited communication bandwidth
- Examples
- Small sensor, actuator, etc.

Issues

- How to configure these devices?
- How do the devices find out appropriate services?
- How do the services get appropriate data from these devices?

"How to connect many small devices to the Internet?"

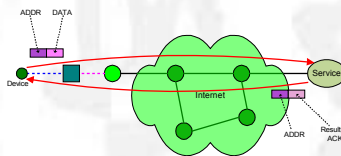
Care more about **data** (generated by device) than device itself

Put intelligence into the wired infrastructure and make small devices as simple as possible

Let devices inject only data into the network and the network figure out what to do with the injected data

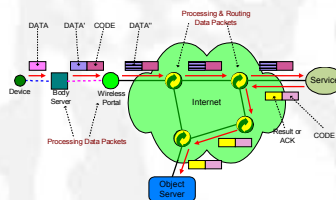
Put a delegate server for every real world object (e.g. person, location, etc.) inside the network

Drawbacks of TCP/IP-based Approach



- Need to configure each device
- Each device must be configured every time it is reconnected to the Internet
- Device must have enough intelligence to manage connection
- Need more computation power, and TCP/IP stack.
- End-to-end connection is not always available
- Fragile wireless link
- Intermittent connection

Our 'Data Centric Networking' Approach

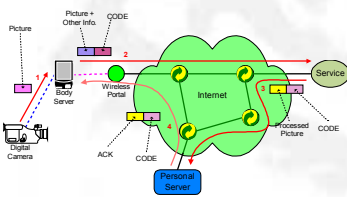


- Devices don't have to maintain any state
- Devices can be as simple as possible
- Devices can be used immediately without any configuration
- Device doesn't need to maintain connection
- Appropriate for intermittently connected low power devices

Put intelligence into the wired infrastructure
Devices generate data and inject them to the network

- Code is added to the data
- Data generated by devices is processed and routed in the network infrastructure
- Every real world object (e.g. person, location, etc.) has its own delegate server in virtual world
- The object (delegate) server acts like a home agent for its associated real world object

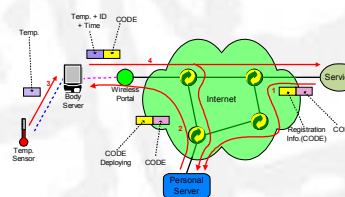
Example 1. Photo Service



Alice borrows a digital camera from Bob, and takes a picture. The camera sends the photo image to her body server. Alice's body server adds more information such as her ID, location to the image data and sends it to nearest wireless portal with a code describing how to process the photo. And, her body server preserves the photo for backup.

The data packet is routed to the photo service by the packet processing nodes inside the network. The photo service processes the photo image and sends the result to Alice's personal server. Alice's personal server sends acknowledgement to her body server later when the body server is connected to any wireless portal. When the body server gets the acknowledgement, it erases the photo from its storage.

Example 2. Medical Service



Bob subscribes to a medical service, and the medical service wants to receive Bob's body temperature periodically. In addition, the service wants to store the temperature information to Bob's personal server. So, the service registers itself with Bob's personal server. (The service sends code to Bob's personal server.) Bob's personal server sends the code that is received from the medical server to Bob's body server.

The temperature sensor in Bob's pocket sends temperature data to Bob's body server. Bob's body server attaches the code to the temperature data and sends it to a wireless portal. A node inside the Internet executes the code and duplicates the data and sends one to the medical service and the other to Bob's personal server.