A Digital Object Approach to IoT

P.Kirstein and S.Varakliotis (UCL)

The Background to IoT6 Gateways

- Large number of IoT devices by 2030 – Trillions
  - Even 100s billions controllers
  - Controllers and/or devices increasingly IP-enabled
- Authorisation and Security completely vitabut
  - Many end-devices weak, security protection complex
- IoT for legacy devices will continue for a long time
  - Computers or phones replaced every 3-5 years
  - Many industrial and infrastructures replace in decades
- Gateways best simple, functionality in servers

Network, Gateway and Device Needs

- Must be scalable to large numbers & global reach
- IoT applications/devices need configuration data
  - Identity and location of devices (in usable form)
  - May need descriptions how technologies act
  - Thus much use of repositories
- Security functions essential
  - Need to provide appropriate services
  - Need services to be implementable
  - Need provisions to be trustworthy
  - Need operations to be trusted

What does Scalability mean?

- Architecture must accommodate large numbers
  - Identifiers, numbers of devices and processes
  - Types of processes, devices and gateways
  - Addressing all above
  - Storing their properties
- Topology must allow large numbers of devices to be accessed, monitored and controlled
- Capable Reach should be global
  - Even though most applications more local

What Security services needed?

- Often authentication services required
  - Both of the requestor and of the target process/device
- Authorisation services vital
  - Directly from requestor or trusted intermediary
  - Achieved by trusted authentication
- May require confidentiality services
- May require non-repudiation of request or reply
- May require audit trail of transactions

Scenarios, Components and Systems

- Achieving scalability and security requires much more than for scenario or separate components
- Web services (REST) provide a scalable mechanism
- IPv6 gives a large Net space for addressing objects
- Passwords, symmetric and asymmetric encryption provide tools to enable security
- However all the above need an appropriate architecture and infrastructure to provide services

IOT6 and HANDLE

- The IOT6 considering how IPv6 addresses main IOT problems
- Have examined many features, and built system particularly for legacy technologies
- Most components not yet incorporate security
- Examining how properties of CNRI HANDLE system help appication composition, scalability, security
- Is existing, deployed, globally accessible technology
- Not necessarily use actual global systems

What is the DNS?

- The Domain Name Structure is a hierarchic naming structure for identifying uniquely network objects
- The DNS is Name Resolution Service resolves the name to network address of a network object
  - The basic DNS normally requires some credentials for locally modifying DNS entries; DNSSEC provides evidence that the DNS entries returned are authentic
  - The DNS system is globally distributed and replicated
  - There is no confidentiality on DNS entries, and no authorisation required to access them
- There are components for Registering, Storing and Rolving Domain Names
  - There may be a searchable database linking properties of a device to its domain name

What is HANDLE

- The HANDLE naming structure assigns a unique hierarchic identifier - a *Handle* - to a digital object
- A Handle is a record that associates the identifier with a list of attribute pairs *type/value*
  - As with Names in the DNS, from a certain level, the owner of the ID space can assign hierarchic *identifiers*
  - *A*ny desired meaning can be associated *with type*
- Like the DNS, the HANDLE system has subsystems
  - Resolving (HIR), registering (HR) and storing Handles (HS)
  - Permitting global access to the Handle store
  - Providing database systems linking properties to Handle
  - Requiring authorisation to update entries

DNS and HANDLE Differences

- There are significant differences also
  - Handle attributes are much richer including security tokens, access to processes and databases
  - Access to a Handle attribute store requires authorisation
  - Handle resolver requests can require authenticated and confidential responses
  - Handle attributes can even be recursive – meaning other Handles can be an attribute
  - HANDLE has better facilities for delegating functionality to subordinates, and for server distribution
    - Would be very important for scaling and management

HANDLE and Security

- HANDLE use can provide the basis for secure systems
  - Attributes of a Handle can be stored in encrypted form
  - They can include passwords and public/private key pairs
  - Access via HANDLE allows delegated authorisation
    - Application may use delegated authorisation – allowing complex authorisation with simple facilities in gateways or end-devices
    - Provide sophisticated facilities where processing power exists, simple acceptance in constrained devices with trustworthy coms
- Trustworthy secure process chaining and communications with DTLS
- Resource protection can rely merely on trustworthy authentication in protected resource
  - Providing authorisation process, inter- process chaining and inter-process communication are trustworthy
  - Normally number of authentication sources limited

Name Structure and Group Operations

- Another aspect is the possibility of different Handles implying different group operations, e.g.:
  - read all sensors in 4th floor
    - May map into IP multicast address
  - set all devices in room x for presence of person
- Some may reflect simple operation on multicast addresses, others more complex operations
- In any case the security parameters may impose authorisation constraints

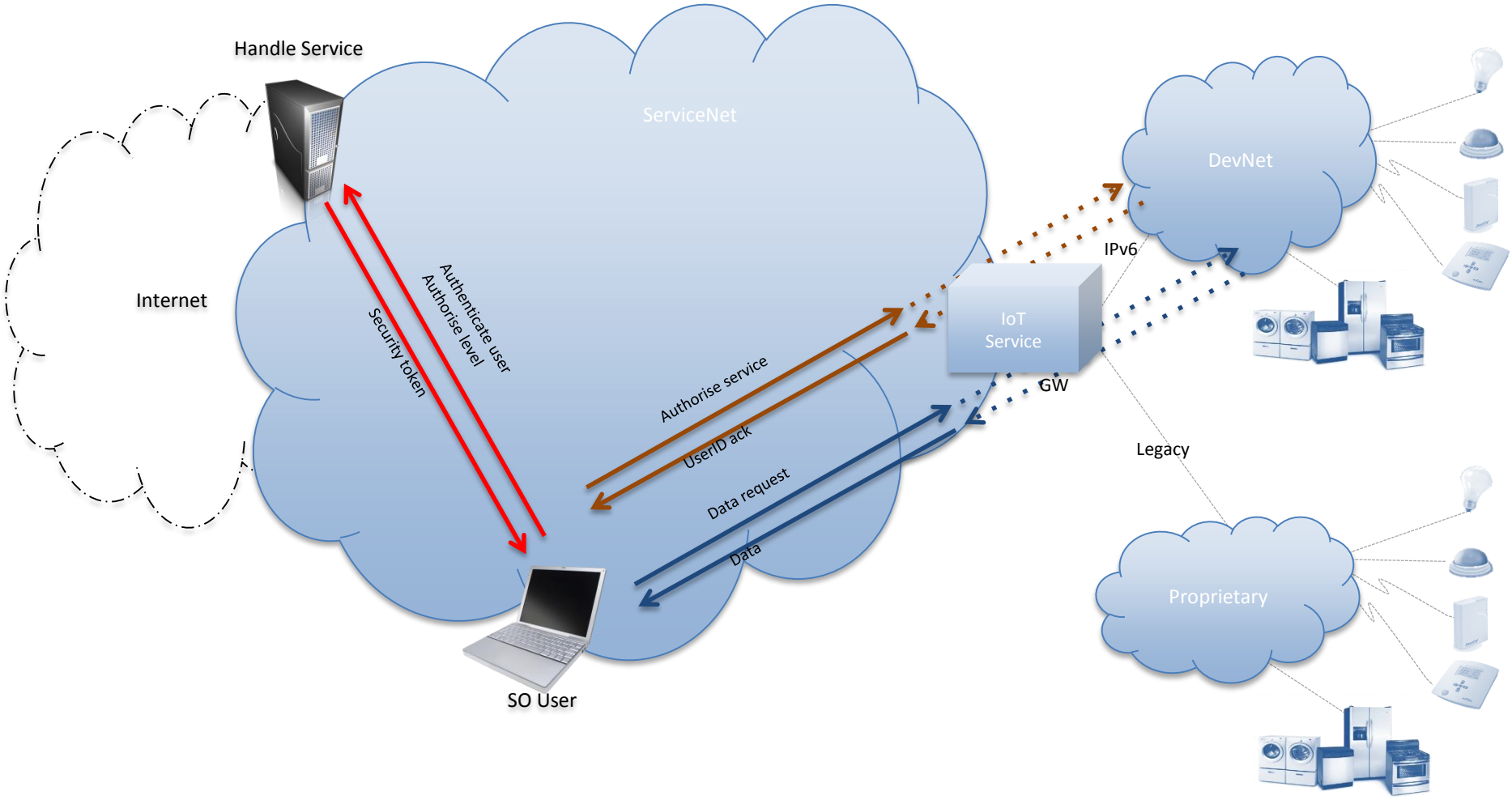
Application Composition

- Many applications are generic, but require configuration to make them specific, e.g.:
  - May call sub-processes
  - Require configuration data
  - Put over specific network stacks
  - Call security subroutines
- HANDLE can give some assistance there
  - E.g. Have Handles for DTLS with same prefix for Linux, Contiki, Windows, then WindowsXP/7/8 ….
  - Or Sensing and Actuation as a Service needs sensors
    - Sensor Type, Sensor location, sensor network access
    - Each may refer to complete Handles

Application-initiated operations

- To initiate an IoT operation, known by its Handle, the application accesses the HIR,
  - Giving the Handle ID, the application ID (AID), and Requestor ID (RID)
- The HIR accesses the HS and checks the authorisation of the AID
  - If AID is authorised, HIR returns Handle record, which may include parameters, resource address and security token (ST) forall possible actions, and authorisations for the RID
- Application may then access resource with authorised ST
- Resource need check only correctness of ST, and if satisfied will perform action requested
- Note much of the information may have been stored in the application during a set-up phase

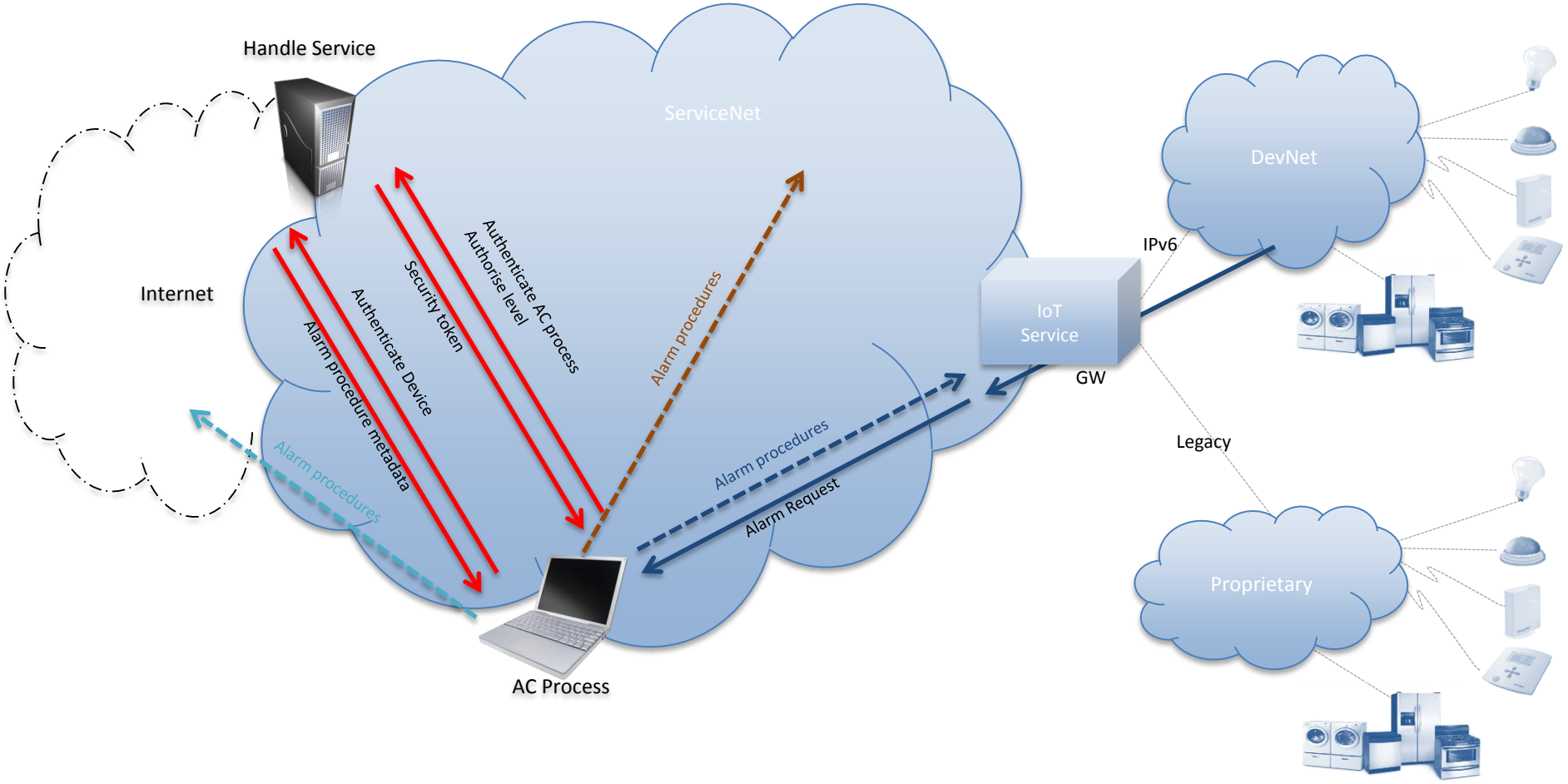
Application-initiated operations



Sensor-initiated operations – Sensor Alarm

- The sensor sends alarm parameters to an application, with its sensor authentication token (SAT) and Sensor ID (SID)
- The application accesses the HIR with the Alarm Handle, application ID, the SID, and the SAT
- The HIR accesses the HS and checks the authorisation of the application and the authenticity of the sensor
  - If both pass the check, HIR returns Handle record, which may include parameters, resource address and security token for the Alarm procedures
- Application may then access the Alarm process with the relevant security token (from HANDLE) and parameters from the sensor.
- The Alarm process need check only the Handle authorisation and, if satisfied, will perform the alarm procedures
- Note much of the information may have been stored in the application during a set-up phase

Sensor-initiated operations



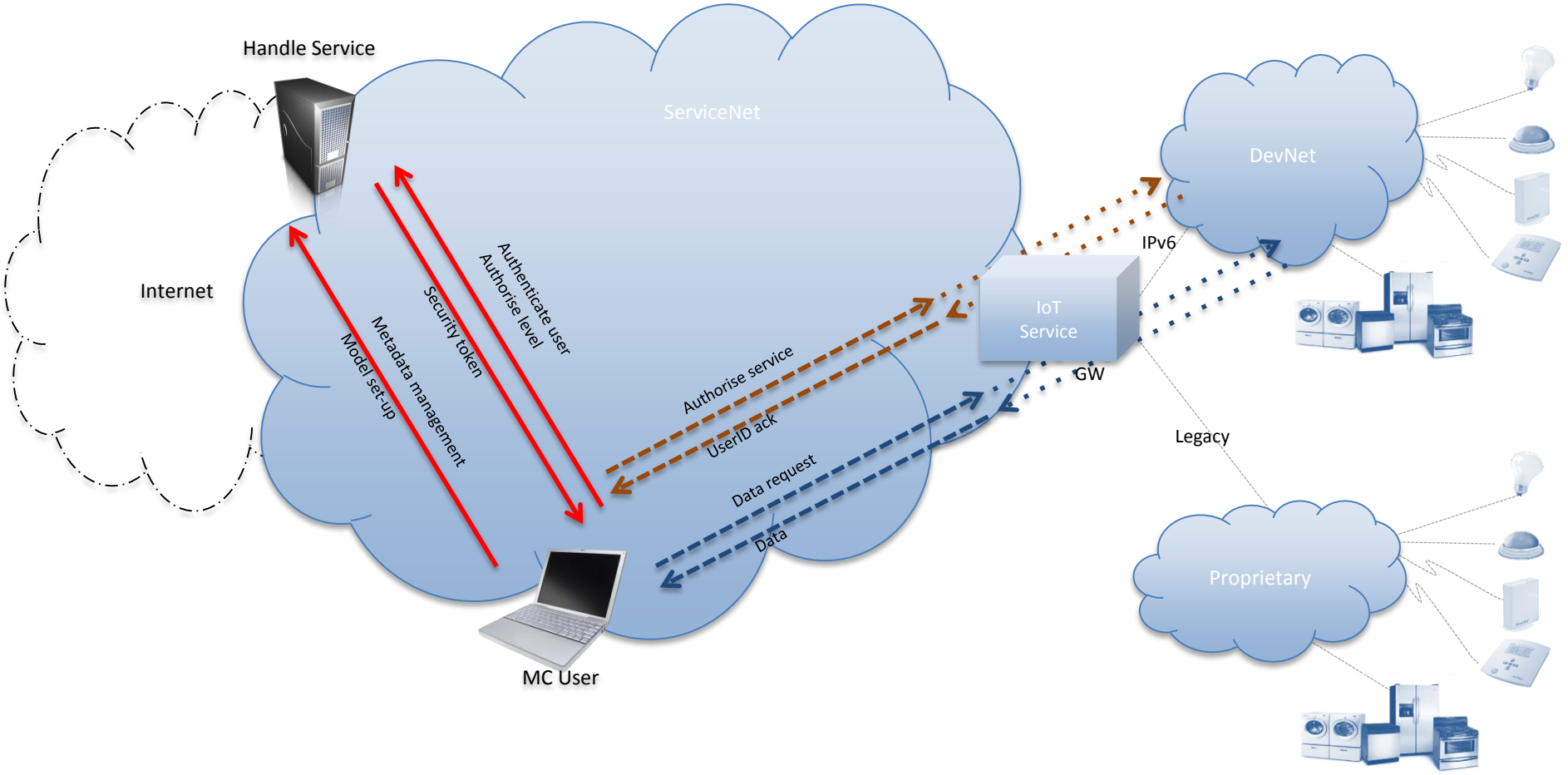
Configuring Applications

- In the previous discussion it was assumed Handle records existed and authorisation needs known
  - In fact a configuration stage is almost always needed
- For any application, there must be many models:
  - The Physical Environmint:
    - Rooms, location, contents to be managed
  - Exact contents: make, properties and serials of devices
  - Network addresses of all IP-enabled IoT devices
  - Model Addresses in its structure of legacy devices
  - Application: what and how devices are managed, what security features are needed, what group operations are envisaged
  - Security: how resources are protected, how usage is regulated, which information needs to be secured in what way
  - Authorisation: operations, configuration, re-configuration, security token authorisation, database entry update

Configuration with IOT6 Digcovery

- One could store Handles of room devices and their attributes in Digcoveryor other searchable databases
- Device description may even be accessed by Handle from an STIS database
- Different people, possibly distinguished by roles, can then be in another Digcovery database, with authentication token
- The application then takes these Digcovery-found resources (defined by Handles), adding authorisation priviledges and puts into HANDLE as new Handles
- It is then possible to do group operations with authorisation, by defining multiple Handles

Configuring Handles in HANDLE



Handle and Legacy Devices

- Use of HANDLE not constrained to security use
- With legacy devices, intermediate processes may translate technology dependent ←> independent
- Access to HIR with Handle may then return also Handle to intermediate process for that technology
- Application could then send parameters to intermediate process to provide correct technology-dependent parameters back for end resource (using DTLS both ways)
- This could be used with legacy IOT6 gateways, but would require effort by partners

Smart Office Set-up

- The smart office building has rooms 1 – n
- Room x has lights ID Lx, HVAC Hx, door actuator Dx, Access sensor Ax, Presence sensor Px
- Dx is type TDx, address ADx, needs token SDx to open
- Ax has serial number Asx, ……
- Users U1, U2, … are allowed to enter room x
- SECx is the process of reading the person's ID wanting to enter the room
- SETUPx is the process to open door, switch on lights, set temperature on HVAC, start monitoring room
  - Needs authorisation code Sx to initiate process

Smart Office HANDLE Usage - Entry

- A configuration Database is set up assigning the equipment in each room. Each item has a Handle to indicate type, net address, access code
- Each employee has entry card giving ID and access credential
- Person puts card on door sensor. Room Handle ID and person Handle ID sent to HANDLE Resolver
- The Handle Resolver retrieves person access code. Checks that Entry is permitted with that code. If permitted, the application sends Request to Room initiation routines (including door opening) with HANDLE authorisation code
  - The initiation routines are partly generic: *open door, put on light, put on HVAC*, and partly specific, even with further access to Handles
  - The application may also initiate recording routine to note the access
- Note most of the routines are those carried out in the normal scenario

Smart Office Alarm

- Many applications are run from a centralised script
  - Some like alarms, may be initiated by a sensor ready
- If an alarm is indicated by Sensor Ax, the sensor may issue an alarm message to a pre-set alarm routine giving the serial number Asx of the sensor and its authentication code.
  - The authentication code will be checked by the Alarm routine, and if it checks out, the application will initiate the relevant alarm procedures
  - Alarm message may include ID of person present from Presence sensor Px
- Again the routines are those undertaken without HANDLE. HANDLE merely gives addresses, checks authenticity, carries out any authorisation functions

Hospital Patient Set-up

- As before, there should be databases for equipment configuration in each room, patients, treatment and room monitoring stations
  - There would be a Handle for each piece of equipment - including authorisation code. If equipment is replaced, its Handle attributes must be updated
  - Handle for each patient including room and treatment
  - Handle for each duty monitoring station including rooms
  - Handle for each treatment routine
- Again duty nurse on coming on shift must authorise herself with system, enter which rooms will be monitored and gets authorisation codes
- Then enters sort of scenario now in system, except that addresses and authorisation codes must be put into treatment routines – as authorised from HANDLE system

Validation in IoT6

- The preceding shows that activities based on Digital Objects can be very powerful and may revolutionise both IoT and the Future Internet
- We will validate the concept in the smart office
- We will show the secure management of specific situations in the IoT6 demonstrations
  - Do not claim properly validated security
- We will use some 3 Motes, an application server and local Handle Server
- With these we will simulate authorised opening a doors, putting on lights and HIVACs, monitoring temperature and presence of room occupation
- We shut down the services when room empty room
- Details are given in short document circulated

Conclusions - Secure Services in IoT6

- The work described here shows how secure services can be introduced into IoT6 and even demonstrated integrated with the applications
  - Though the scenarios have had to be slightly modified to demonstrate these features
- Fundamental to the approach is the adoption of a secure, identifier-based, infrastructure with advanced features
  - The approach must be reflected in WP1 architecture
- The mechanisms are not novel, however they use an existing global, scalable, infrastructure

Conclusions – Scaling in IoT6

- There is huge discrepancy between 100s of objects demonstrated in IoT6 and trillions envisaged in IoT
  - The discrepancy cannot be bridged by looking at the scaling of single components of Use Cases
- Only by considering total systems architectures can scaling issues be tackled
- The architecture must incorporate the following
  - A hierarhic Identifier space distributed, unbounded and with application-specific globally accessible resolvers and registries
  - Repositories that can be replicated, globally accessed and with facilities to be specialised and expandable
  - Networks with the capability of addressing trillions of objects
- Since HANDLE is such a system, one needs to demonstrate only that its constituents have these properties

Conclusions

- We have described an approach that allows complex gateway functionality to be implemented in a way that has maximum flexibility and deployability into decomposed processes
  - The structure can provide scalability and security
- As suppliers adopt more standard functions in their controllers, the physical gateways may not need to change
  - Just some of the server processes are obviated
- Clearly this is start of a much wider activity