

# Enabling IoT Platform via MobilityFirst Future Internet Architecture

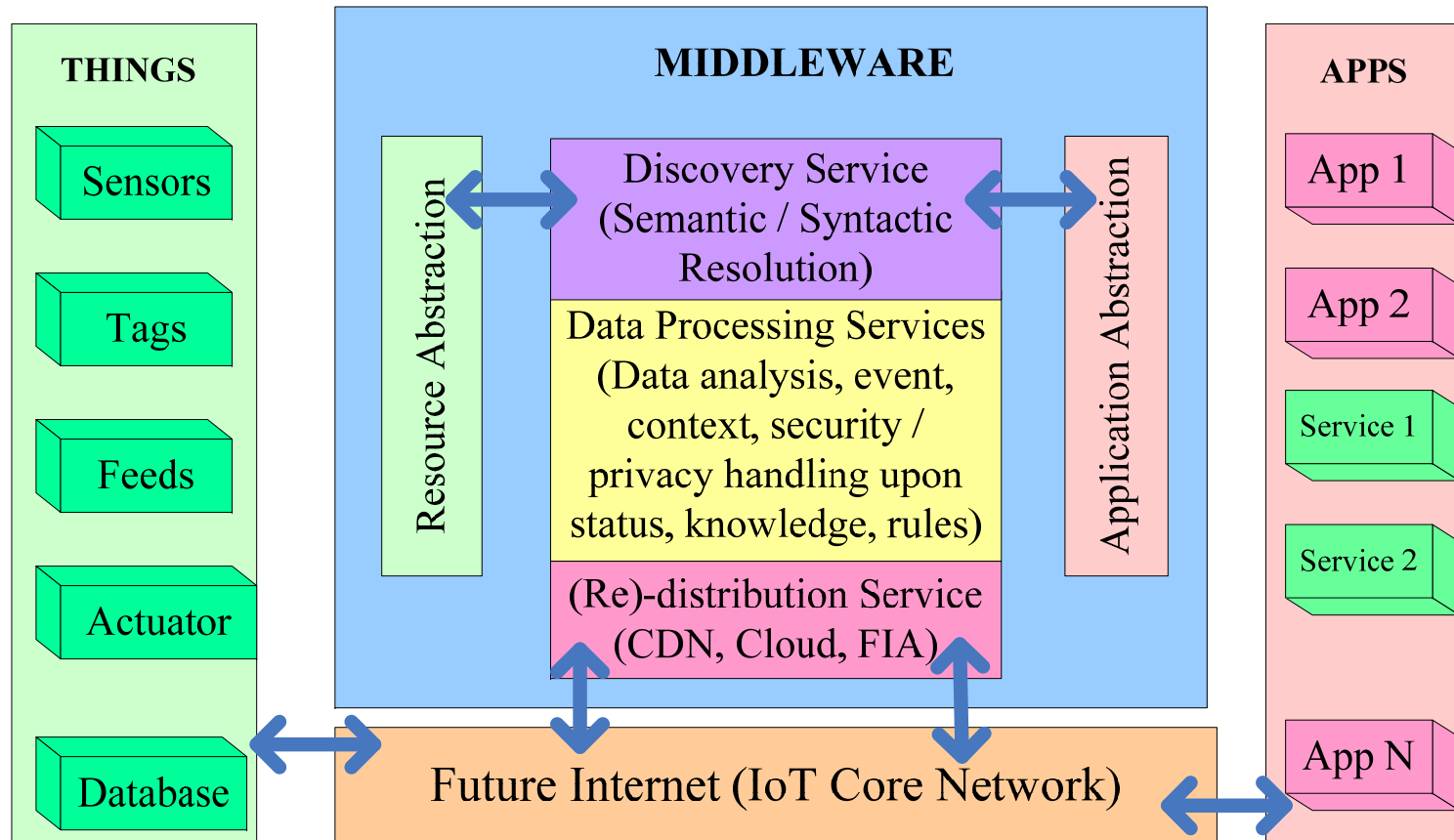
Jun Li

WINLAB, Rutgers University

# Motivations

- A big trend of Internet– Internet of Things (IoT)
  - Transforming today’s network of computers to tomorrow’s network of Things
- A big gap between physical objects and apps
  - Sensors (actuators, tags) are much more heterogeneous than computers, no standard interfaces
- Need IoT Middleware to bridge the gap
- What is the role of future Internet architecture?

# IoT Middleware Functions



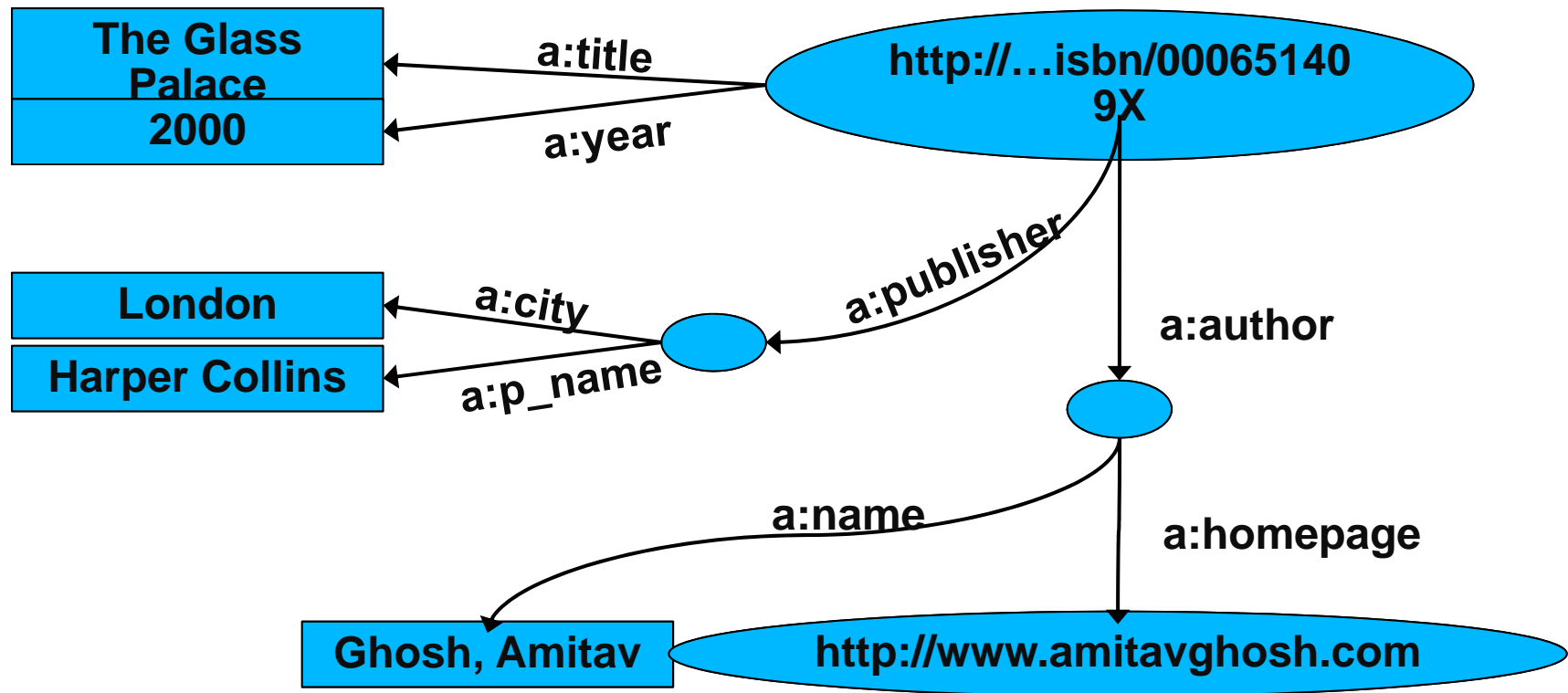
# Today's Middleware

- M2M Middleware
  - Specialized to vertical market apps
  - Low interoperability and scalability
- IoT Middleware
  - Generalized to any application
  - Increased interoperability and scalability
- Linked Data
  - Linking all databases created with semantic web technology in the cyberspace
  - A global information space

# Semantic Web Technology

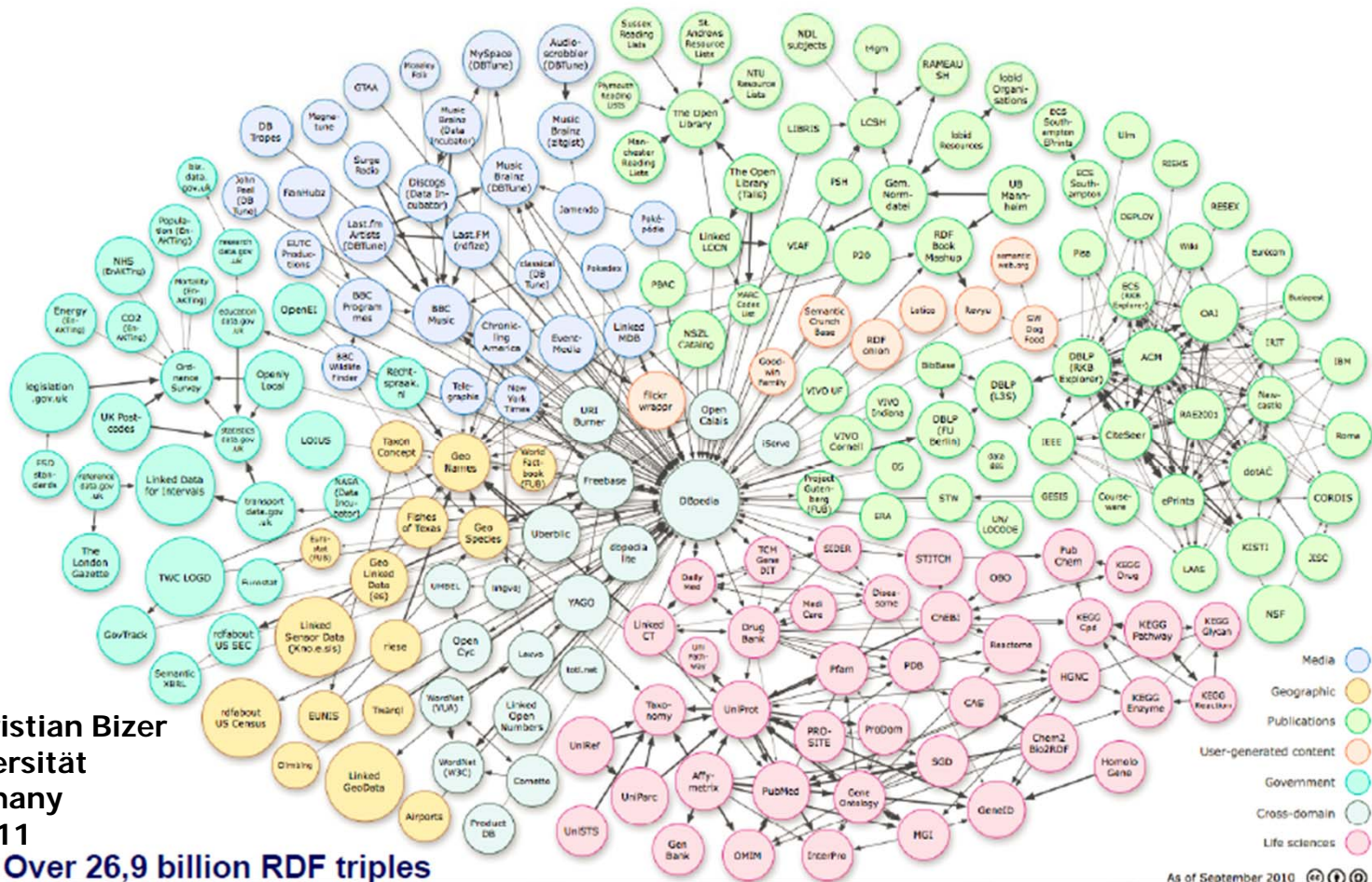
- Building up the relationships between data
  - Store web data with semantic links
  - Discover data from semantic query
- Basics
  - The relationship of data is represented in RDF (resource description framework) triples and graphs
  - The data source with semantic attributes can be query by SPARQL (an RDF query language)
- Linked Data
  - A huge collection of semantic databases over web
  - Sensors can also be linked data, live streaming data

# An RDF graph sample



- Source: Ivan Herman W3C, Oct. 2011

# Linked Data (Sept. 2010)

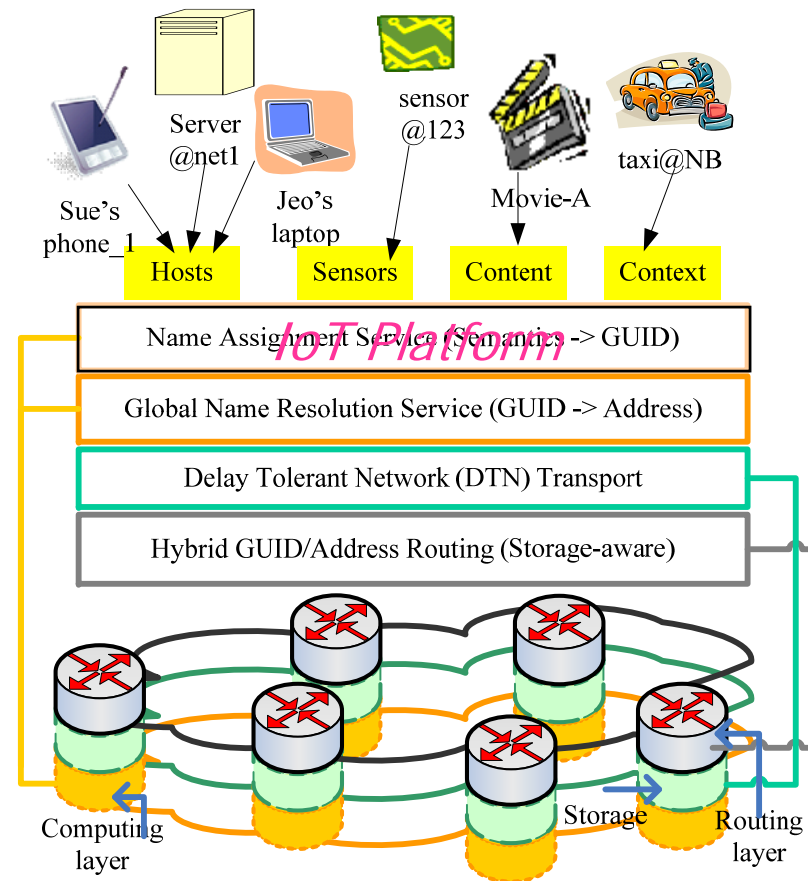


11/30/2011

IoT Platform and MobilityFirst FIA

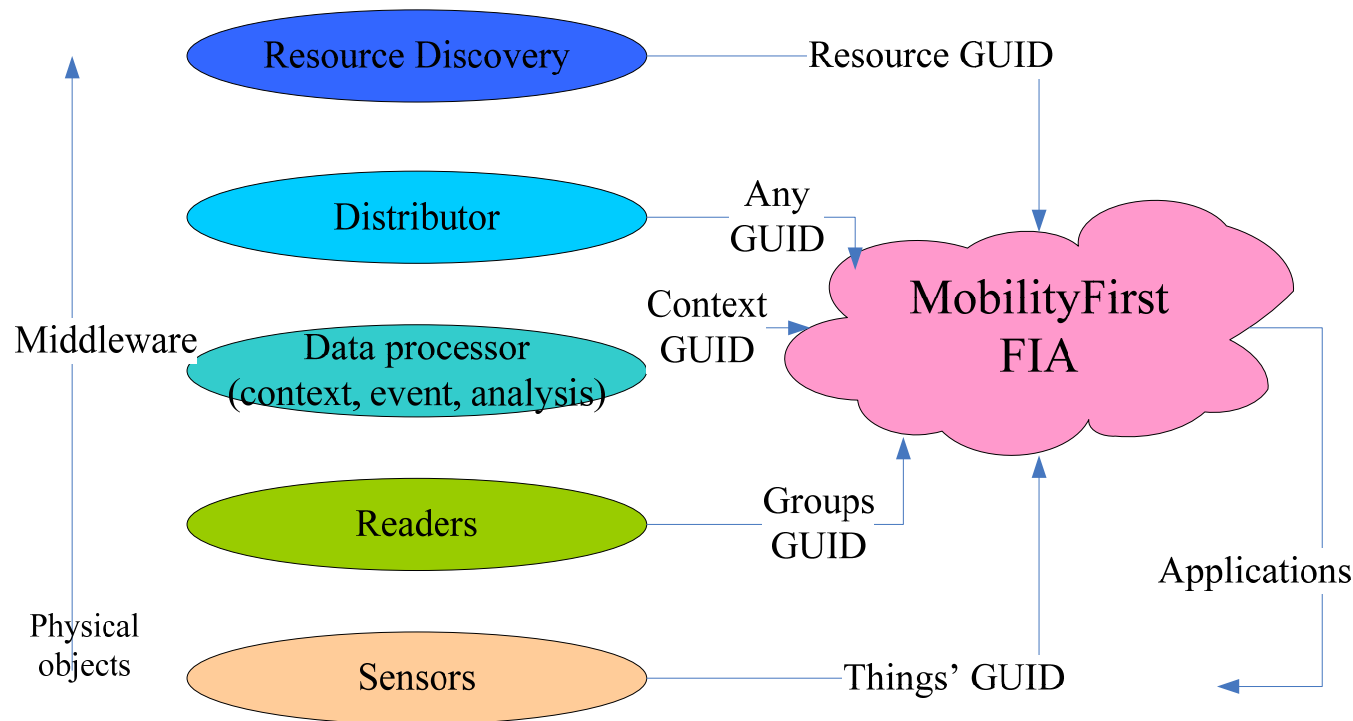
# MobilityFirst Architecture

- Three protocol layers (baseline): Routing, DTN transport, GNRS
- Name assignment service (NAS)
- How MobilityFirst runs as an IoT platform?





# GUID for Middleware Services



- Sensors and Middleware services are visible in core network

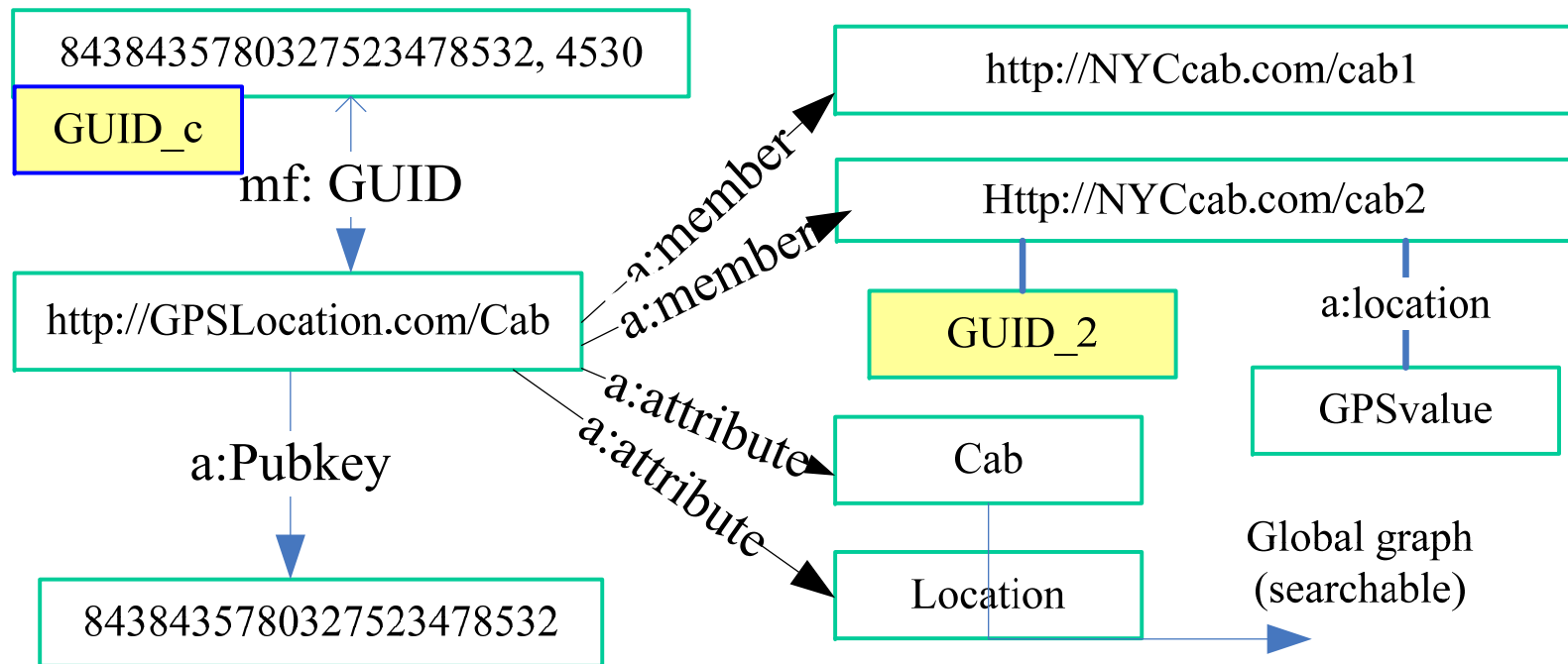
# GUID named context service

- Basic benefits from MobilityFirst
  - Multi-homing and mobility
    - One GUID to multiple URIs or IPs
  - Requests and data can be multicast, anycast
    - Request to GUID going to all mapped addresses
  - No end-to-end connectivity required
    - Hop-by-hop and late binding
  - Anti-spoofing
    - Verifying authentication with public key in GUID

# A context-aware application

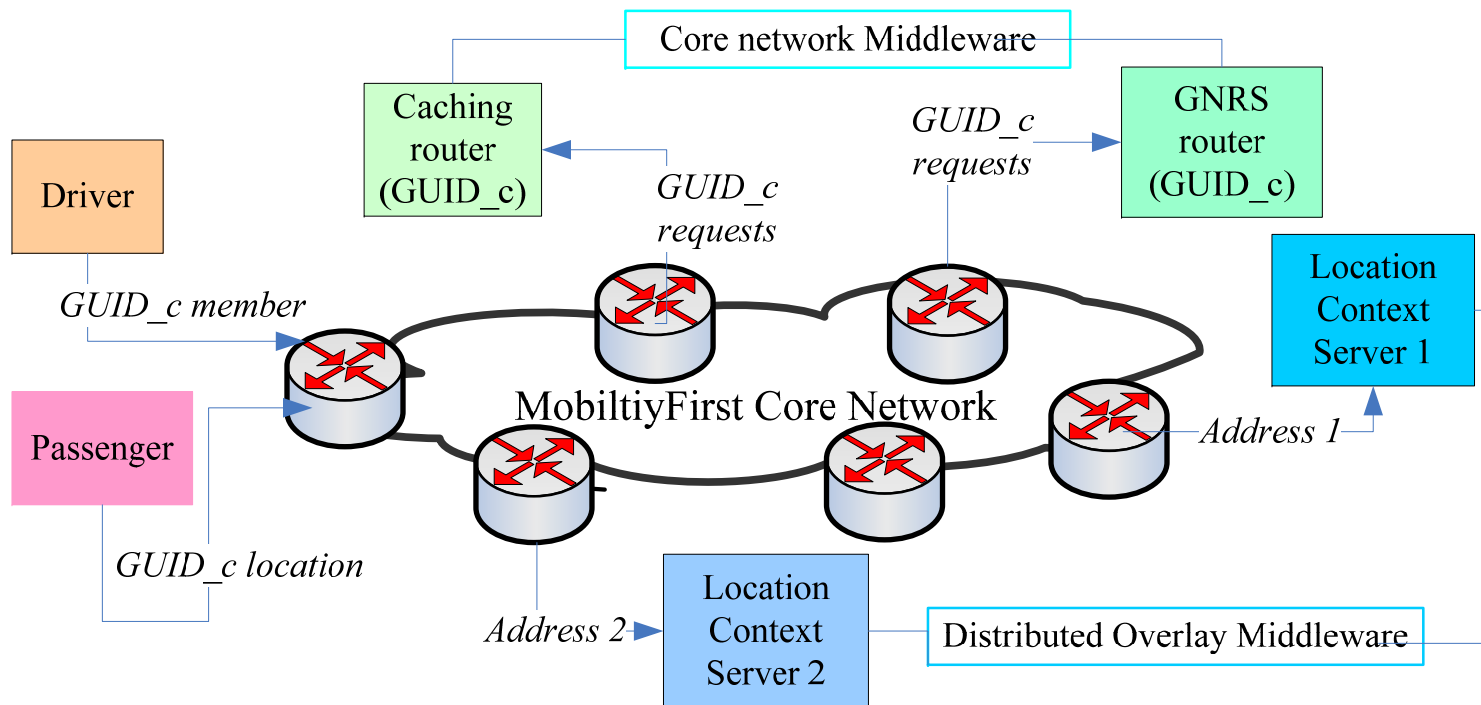
- UbiCab example
  - *“James walks on NYC streets and wants to find an empty cab closest to his location”.*
- Resources – phones with GPS values as data
- A location context service, as middleware, collects GPS sensor data and provides to application (a voice call app)
- What does it mean resources and service being assigned with GUIDs?

# Location context publishing



- An RDF graph published via MobilityFirst NAS

# Caching – from overlay to core

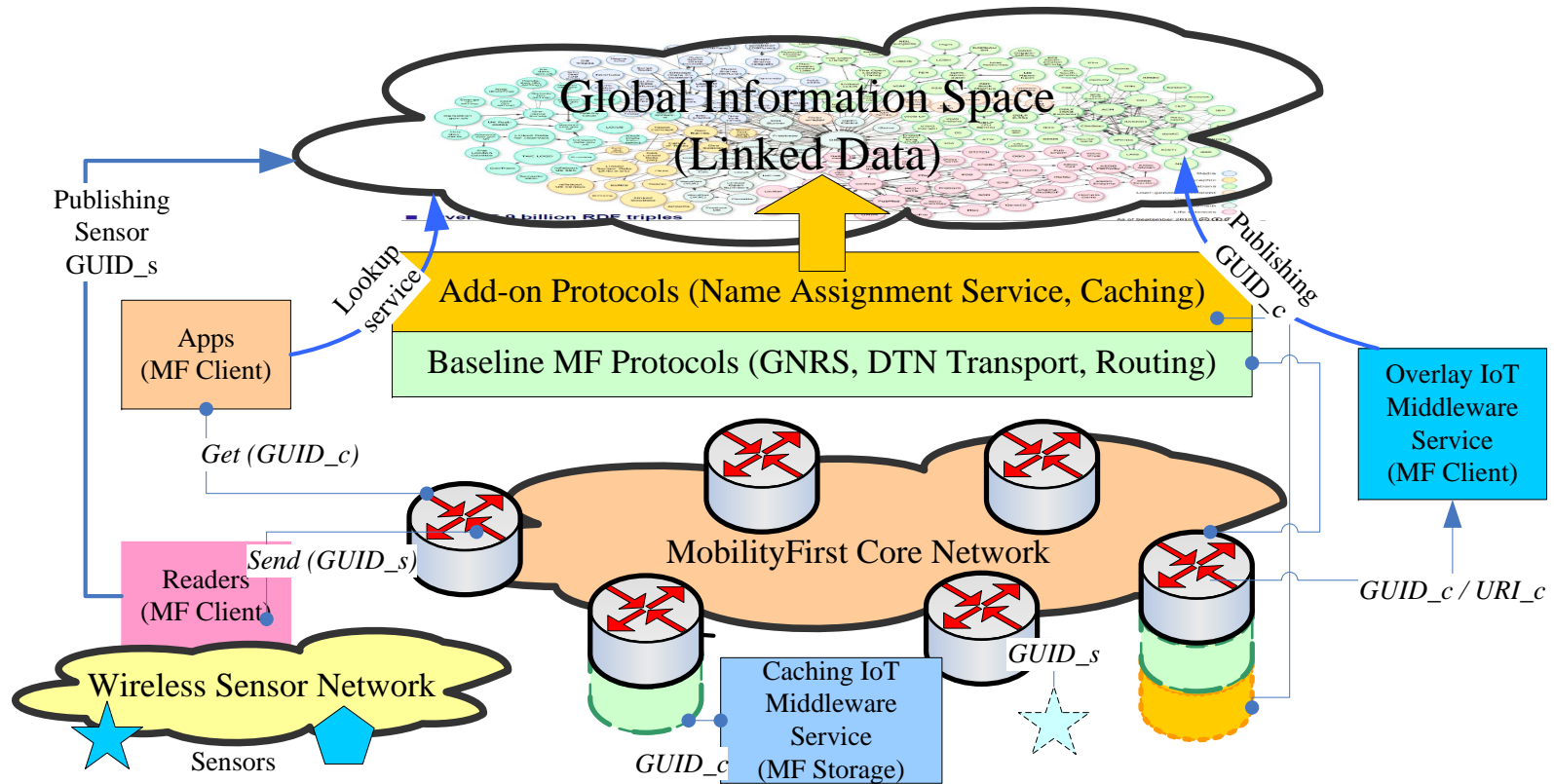


- Service is operated on GUIDs inside core network

# Cached context service

- Requirements on routers
  - Storage available and support standard RDF SPARQL or a default query language
  - Cached service is a dynamic database – in RDF graph
- Benefits
  - Reduce traffic load, lower latency
  - Available when disconnected to overlays due to mobility or ad-hoc conditions
  - Efficiency for location dependent and high data rate services

# MobilityFirst as IoT Platform



- Resource (GUID\_s) , Service (GUID\_c) visible and cacheable in core
- Interface to Linked Data via MF NAS (Name Assignment Service)

# Conclusions

- MobilityFirst can provide an IoT platform to connect Things
- Things are identifiable at MF core network layer via GUIDs
- Middleware functions (services) are identifiable at MF core network layer via GUIDs
- Middleware functions can be cached to enhance the efficiency of the service delivery