

# Ad Hoc Pervasive Computing

Dr. Dirk Husemann  
IBM Zurich Research Lab  
[hud@zurich.ibm.com](mailto:hud@zurich.ibm.com)

28 November 2000

# How Ad Hoc is Ad Hoc?

**Ad hoc:** . . . fashioned from whatever is immediately available. (1)

# How Ad Hoc is Ad Hoc?

**Ad hoc:** . . . fashioned from whatever is immediately available. (1)

- Ad hoc networking: “plug and play”

# How Ad Hoc is Ad Hoc?

**Ad hoc:** . . . fashioned from whatever is immediately available. (1)

- Ad hoc networking: “plug and play”
  - Auto configuration of network interfaces
  - Route discovery (where to send your packets)
  - Service discovery (what is available)

# How Ad Hoc is Ad Hoc?

**Ad hoc:** . . . fashioned from whatever is immediately available. (1)

- Ad hoc networking: “plug and play”
  - Auto configuration of network interfaces
  - Route discovery (where to send your packets)
  - Service discovery (what is available)
- Plug in, stay, and play: **Long-term ad hoc networking**

# How Ad Hoc is Ad Hoc?

**Ad hoc:** . . . fashioned from whatever is immediately available. (1)

- Ad hoc networking: “plug and play”
  - Auto configuration of network interfaces
  - Route discovery (where to send your packets)
  - Service discovery (what is available)
- Plug in, stay, and play: **Long-term ad hoc networking**
- “Butterflying” & diversity: **Transient ad hoc networking**



# Scenarios

- Office scenario
- Refrigerator & shopping scenario
- Tourist scenario
- Airport scenario

# Issues

- Fast network discovery
  - Modern RF: **spread spectrum frequency hopping**



# Issues

- Fast network discovery
  - Modern RF: **spread spectrum frequency hopping**
- Network range
  - Large range: Connectivity
  - Short range: **Proximity information**

# Issues

- Fast network discovery
  - Modern RF: **spread spectrum frequency hopping**
- Network range
  - Large range: Connectivity
  - Short range: **Proximity information**
- Routing

# Issues

- Fast network discovery
  - Modern RF: **spread spectrum frequency hopping**
- Network range
  - Large range: Connectivity
  - Short range: **Proximity information**
- Routing
- Fast service discovery

# Issues

- Fast network discovery
  - Modern RF: **spread spectrum frequency hopping**
- Network range
  - Large range: Connectivity
  - Short range: **Proximity information**
- Routing
- Fast service discovery
- Service description

# Issues

- Fast network discovery
  - Modern RF: **spread spectrum frequency hopping**
- Network range
  - Large range: Connectivity
  - Short range: **Proximity information**
- Routing
- Fast service discovery
- Service description
- Scalability

# Networking technology

- Wired technologies: Suitable for long-term ad hoc networking

# Networking technology

- Wired technologies: Suitable for long-term ad hoc networking
- **Transient ad hoc networking: Wireless technology a necessity**

# Networking technology

- Wired technologies: Suitable for long-term ad hoc networking
- **Transient ad hoc networking: Wireless technology a necessity**
- Wireless: RF & Infrared
  - Infrared: IrDA, **AIR**



# Networking technology

- Wired technologies: Suitable for long-term ad hoc networking
- **Transient ad hoc networking: Wireless technology a necessity**
- Wireless: RF & Infrared
  - Infrared: IrDA, **AIR**
  - RF: **Bluetooth, IEEE 802.11**, HomeRF, DECT

# Wireless technologies

Data rate

Range

Nodes

Discovery

Topology

Spectrum

Target

# Wireless technologies

	AIR
Data rate	4 Mbps
Range	1.5–7.6m
Nodes	10
Discovery	< 250 $\mu$ s
Topology	network, pt-2-pt
Spectrum	850-900nm
Target	ad hoc, cable repl.

# Wireless technologies

	AIR	Bluetooth
Data rate	4 Mbps	1 Mbps
Range	1.5–7.6m	10m (100m +20dB)
Nodes	10	8
Discovery	< 250 $\mu$ s	> 10 – 30s
Topology	network, pt-2-pt	pt-2-multi pt
Spectrum	850-900nm	2.4GHz
Target	ad hoc, cable repl.	cable replacement

# Wireless technologies

	AIR	Bluetooth	IEEE 802.11b
Data rate	4 Mbps	1 Mbps	11 Mbps
Range	1.5–7.6m	10m (100m +20dB)	100m
Nodes	10	8	127
Discovery	< 250μs	> 10 – 30s	immediate
Topology	network, pt-2-pt	pt-2-multi pt	network
Spectrum	850-900nm	2.4GHz	2.4GHz
Target	ad hoc, cable repl.	cable replacement	wireless LAN

# Service Discovery (1)

- **What is a service?**

# Service Discovery (1)

- **What is a service?**
  - **Processor concept:** input || processing || output

# Service Discovery (1)

- **What is a service?**
  - **Processor concept:** input || processing || output
  - **API concept:** set of interface descriptions



# Service Discovery (1)

- **What is a service?**

- **Processor concept:** input || processing || output
- **API concept:** set of interface descriptions
- **Component concept:** piece of software implementing specific interfaces (proxied, direct)

# Service Discovery (1)

- **What is a service?**
  - **Processor concept:** input || processing || output
  - **API concept:** set of interface descriptions
  - **Component concept:** piece of software implementing specific interfaces (proxied, direct)
- **Service description**
  - **Simple to generate & parse, compact**

# Service Discovery (1)

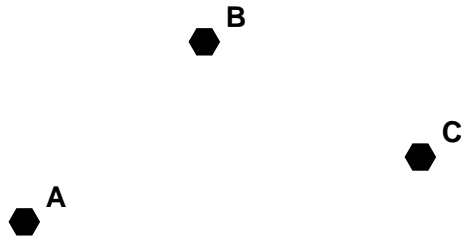
- **What is a service?**

- **Processor concept:** input || processing || output
- **API concept:** set of interface descriptions
- **Component concept:** piece of software implementing specific interfaces (proxied, direct)

- **Service description**

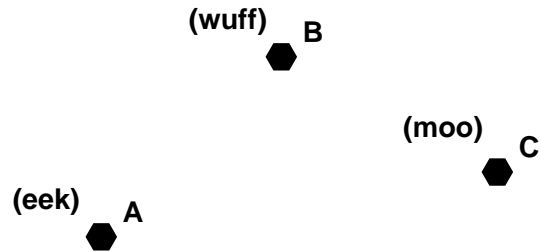
- **Simple to generate & parse, compact**
- Support *delta descriptions*

## Service Discovery (2)



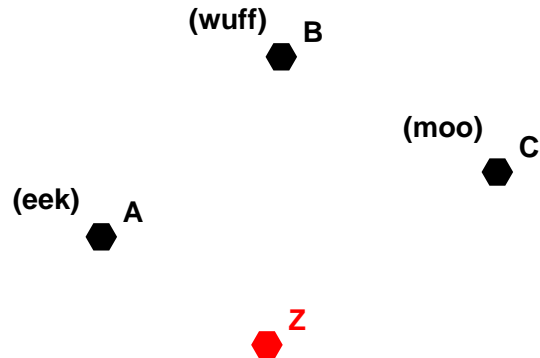
- **A, B, and C** present

## Service Discovery (2)



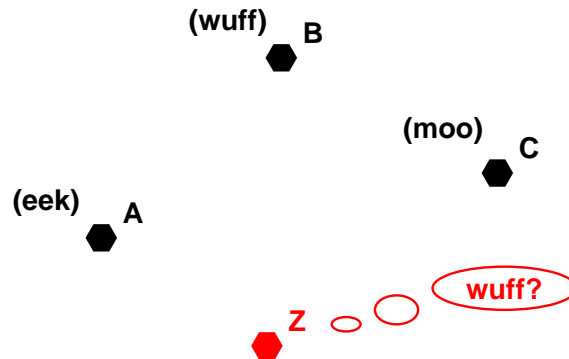
- **A**, **B**, and **C** present
- Devices offer some kind of **service**

## Service Discovery (2)



- **A**, **B**, and **C** present
- Devices offer some kind of **service**
- **Z** joins the group,

## Service Discovery (2)



- **A**, **B**, and **C** present
- Devices offer some kind of **service**
- **Z** joins the group, requires **wuff** service

## Service Discovery (3)

- Service discovery: Which device provides a required service?



## Service Discovery (3)

- Service discovery: Which device provides a required service?
- Two approaches:
  - **Pull model:** query the environment when service required

## Service Discovery (3)

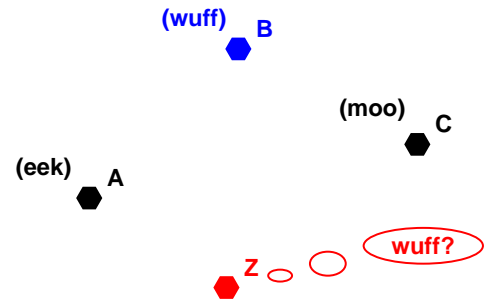
- Service discovery: Which device provides a required service?
- Two approaches:
  - **Pull model:** query the environment when service required
  - **Push model:** push service information into the environment

## Service Discovery (3)

- Service discovery: Which device provides a required service?
- Two approaches:
  - **Pull model:** query the environment when service required
  - **Push model:** push service information into the environment
- In depth: **Distributed push, centralized push, & pull**

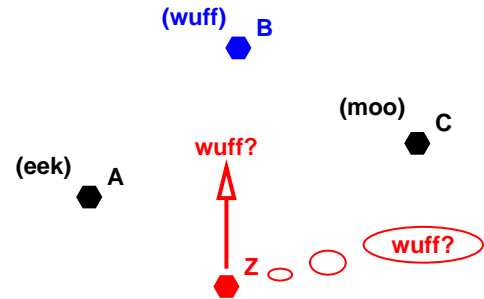
# Service Discovery: Distributed Pull

- Node **Z** requires service **wuff**



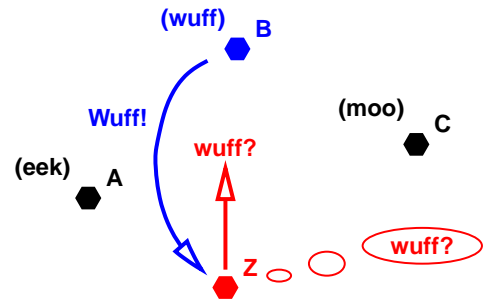
# Service Discovery: Distributed Pull

- Node **Z** requires service **wuff**
- Node **Z** **broadcasts** request for **wuff**



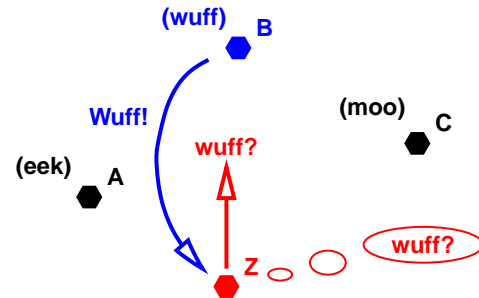
# Service Discovery: Distributed Pull

- Node **Z** requires service **wuff**
- Node **Z** **broadcasts** request for **wuff**
- Node **B** responds with service information



# Service Discovery: Distributed Pull

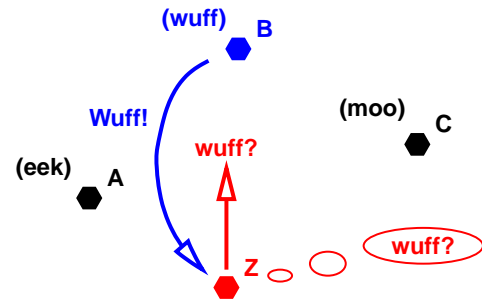
- Node **Z** requires service **wuff**
- Node **Z** **broadcasts** request for **wuff**
- Node **B** responds with service information



- Problem: **RF broadcast** rather **unreliable**

# Service Discovery: Distributed Pull

- Node **Z** requires service **wuff**
- Node **Z** **broadcasts** request for **wuff**
- Node **B** responds with service information

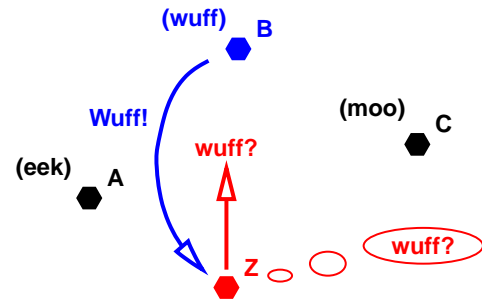


- Problem: **RF broadcast** rather **unreliable**
- Problem: B in **sleep mode?**



# Service Discovery: Distributed Pull

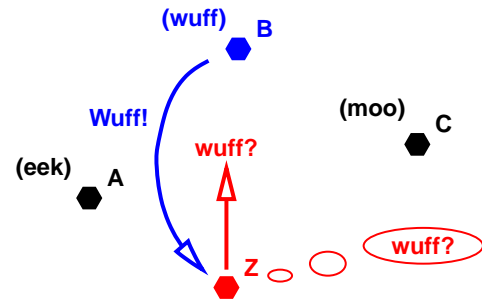
- Node **Z** requires service **wuff**
- Node **Z** **broadcasts** request for **wuff**
- Node **B** responds with service information



- Problem: **RF broadcast** rather **unreliable**
- Problem: B in **sleep mode?**
- Problem: Reachability

# Service Discovery: Distributed Pull

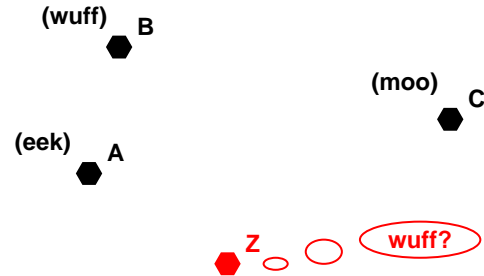
- Node **Z** requires service **wuff**
- Node **Z** **broadcasts** request for **wuff**
- Node **B** responds with service information



- Problem: **RF broadcast** rather **unreliable**
- Problem: B in **sleep mode?**
- Problem: Reachability
- Service discovery slow

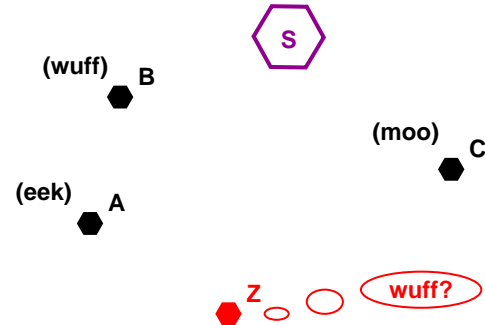
# Service Discovery: Centralized Pull

- Node **Z** requires service **wuff**



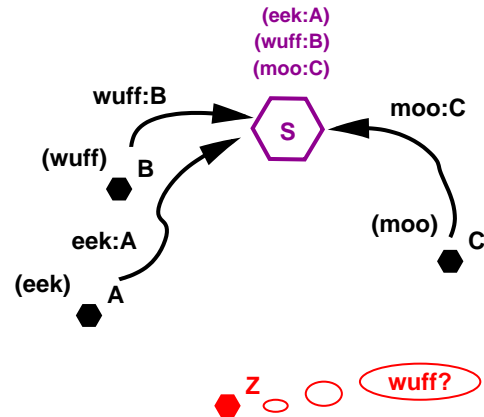
# Service Discovery: Centralized Pull

- Node **Z** requires service **wuff**
- Central repository **S**



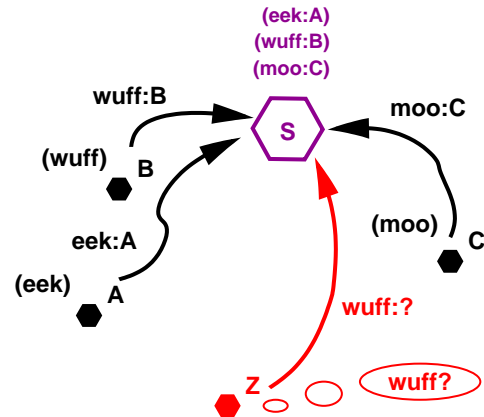
# Service Discovery: Centralized Pull

- Node **Z** requires service **wuff**
- Central repository **S**
- Nodes **A**, **B**, **C** register with **S**



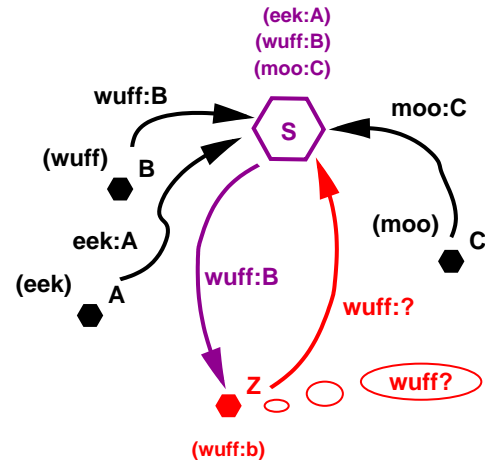
# Service Discovery: Centralized Pull

- Node **Z** requires service **wuff**
- Central repository **S**
- Nodes **A**, **B**, **C** register with **S**
- Node **Z** queries **S** for **wuff**



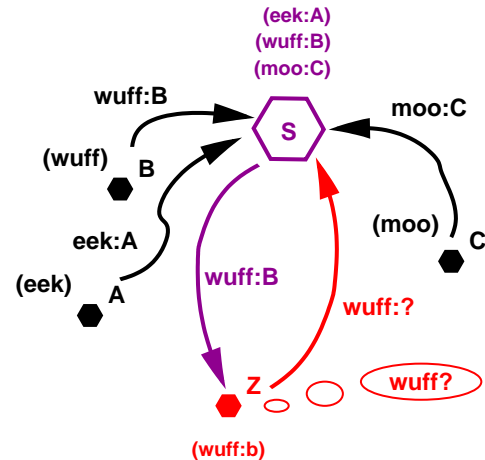
# Service Discovery: Centralized Pull

- Node **Z** requires service **wuff**
- Central repository **S**
- Nodes **A**, **B**, **C** register with **S**
- Node **Z** queries **S** for **wuff**
- Node **S** responds with service information



# Service Discovery: Centralized Pull

- Node **Z** requires service **wuff**
- Central repository **S**
- Nodes **A, B, C** register with **S**
- Node **Z** queries **S** for **wuff**
- Node **S** responds with service information

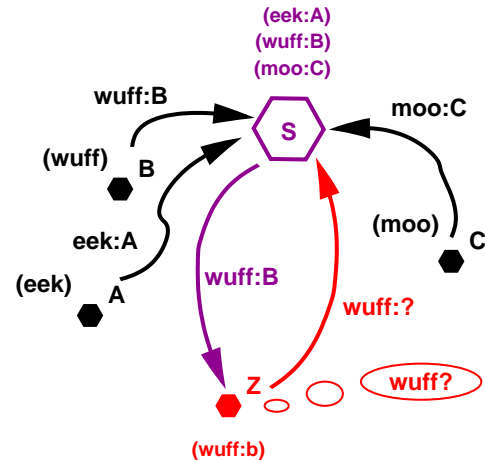


- Problem: **Finding central repository**



# Service Discovery: Centralized Pull

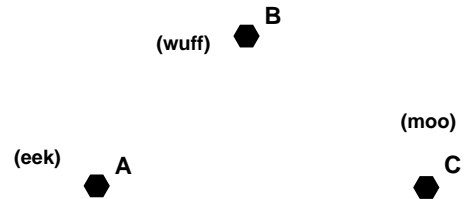
- Node **Z** requires service **wuff**
- Central repository **S**
- Nodes **A, B, C** register with **S**
- Node **Z** queries **S** for **wuff**
- Node **S** responds with service information



- Problem: **Finding central repository**
- Problem: Length of **lease**?

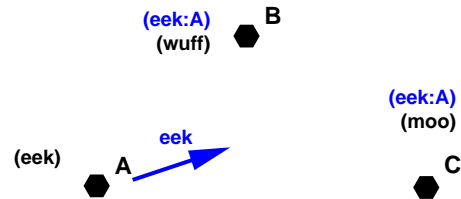
# Service Discovery: Distributed Push

- Node **Z** requires service **wuff**



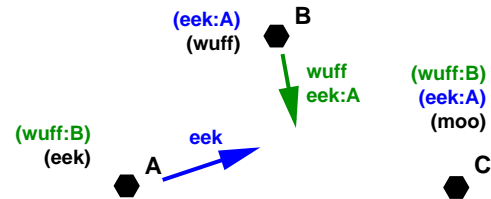
# Service Discovery: Distributed Push

- Node **Z** requires service **wuff**
- Nodes **periodically broadcast** service lists



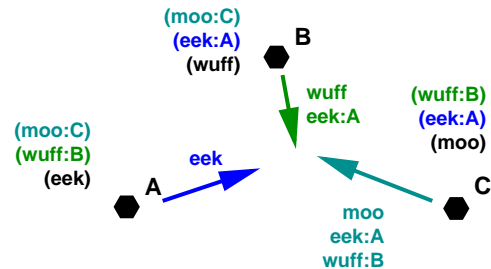
# Service Discovery: Distributed Push

- Node **Z** requires service **wuff**
- Nodes **periodically broadcast** service lists
- Received service lists update existing lists



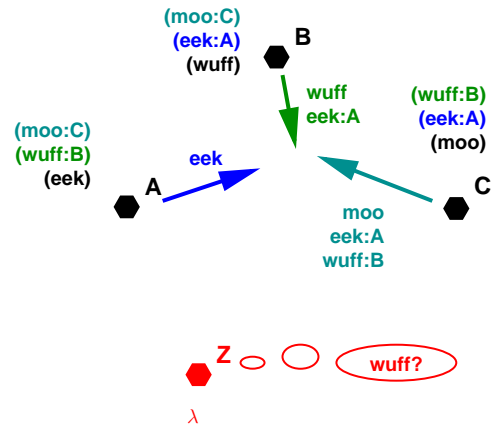
# Service Discovery: Distributed Push

- Node **Z** requires service **wuff**
- Nodes **periodically broadcast** service lists
- Received service lists update existing lists



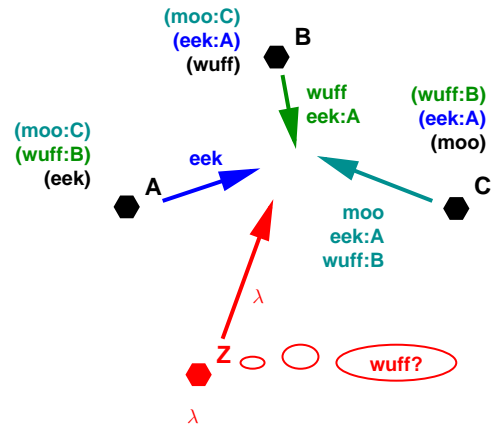
# Service Discovery: Distributed Push

- Node **Z** requires service **wuff**
- Nodes **periodically broadcast** service lists
- Received service lists update existing lists
- (New) node **Z** triggers rebroadcast



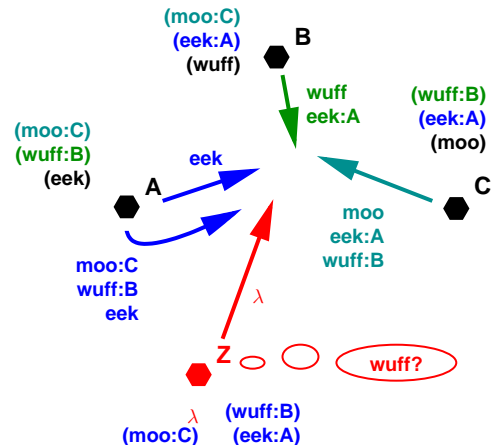
# Service Discovery: Distributed Push

- Node **Z** requires service **wuff**
- Nodes **periodically broadcast** service lists
- Received service lists update existing lists
- (New) node **Z** triggers rebroadcast



# Service Discovery: Distributed Push

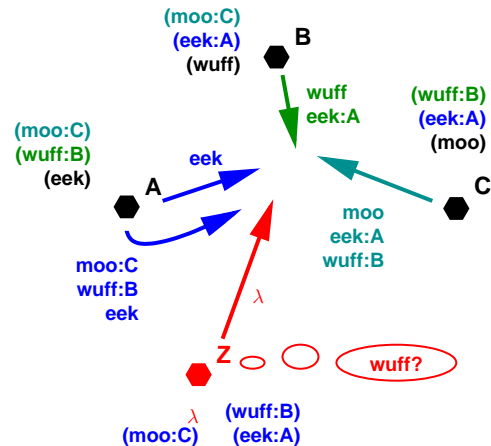
- Node **Z** requires service **wuff**
- Nodes **periodically broadcast** service lists
- Received service lists update existing lists
- (New) node **Z** triggers rebroadcast
- **Z** updated





# Service Discovery: Distributed Push

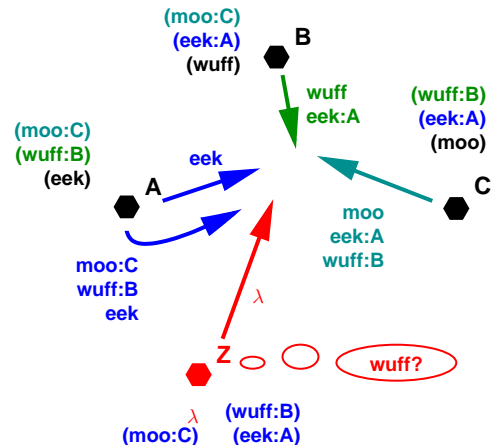
- Node **Z** requires service **wuff**
- Nodes **periodically broadcast** service lists
- Received service lists update existing lists
- (New) node **Z** triggers rebroadcast
- **Z** updated



- **“Always” updated view of environment!**

# Service Discovery: Distributed Push

- Node **Z** requires service **wuff**
- Nodes **periodically broadcast** service lists
- Received service lists update existing lists
- (New) node **Z** triggers rebroadcast
- **Z** updated



- **“Always” updated view of environment!**
- Problem: Length of **lease** and **broadcast interval**?

# Service Description

- Ad Hoc nature imposes restrictions
  - **Compact:** conserve network bandwidth & device memory
  - **Easy to parse:** limited CPU & memory on devices
  - **Deltas**

# Service Description

- Ad Hoc nature imposes restrictions
  - **Compact:** conserve network bandwidth & device memory
  - **Easy to parse:** limited CPU & memory on devices
  - **Deltas**
- Textual description: Strings, XML, SOAP

# Service Description

- Ad Hoc nature imposes restrictions
  - **Compact:** conserve network bandwidth & device memory
  - **Easy to parse:** limited CPU & memory on devices
  - **Deltas**
- Textual description: Strings, XML, SOAP
- Code description: Java interfaces

# Service Description

- Ad Hoc nature imposes restrictions
  - **Compact:** conserve network bandwidth & device memory
  - **Easy to parse:** limited CPU & memory on devices
  - **Deltas**
- Textual description: Strings, XML, SOAP
- Code description: Java interfaces
- Encoding

# Scalability, Efficiency

- “Moore’s law takes care of scalability”

# Scalability, Efficiency

- “Moore’s law takes care of scalability” . . . well, not quite



# Scalability, Efficiency

- “Moore’s law takes care of scalability” . . . well, not quite
- Moore’s law provides for ever shrinking “devices”

# Scalability, Efficiency

- “Moore’s law takes care of scalability” . . . well, not quite
- Moore’s law provides for ever shrinking “devices”
- Possibility to drive pervasive computing further and further

# Scalability, Efficiency

- “Moore’s law takes care of scalability” . . . well, not quite
- Moore’s law provides for ever shrinking “devices”
- Possibility to drive pervasive computing further and further
- (At least) exponential increase in number of possible PvC devices

# Scalability, Efficiency

- “Moore’s law takes care of scalability” . . . well, not quite
- Moore’s law provides for ever shrinking “devices”
- Possibility to drive pervasive computing further and further
- (At least) exponential increase in number of possible PvC devices
- **Costs** will always be an issue: RF bandwidth, memory, CPU

# Scalability, Efficiency

- “Moore’s law takes care of scalability” . . . well, not quite
- Moore’s law provides for ever shrinking “devices”
- Possibility to drive pervasive computing further and further
- (At least) exponential increase in number of possible PvC devices
- **Costs** will always be an issue: RF bandwidth, memory, CPU
- Scalability & efficiency will always be an issue

# Security & Privacy

- “Good question!”

# Security & Privacy

- “Good question!” . . . “Next question, please”

# Security & Privacy

- “Good question!” . . . “Next question, please”
- Authentication, authorization, protection of data & messages



# Security & Privacy

- “Good question!” . . . “Next question, please”
- Authentication, authorization, protection of data & messages
- Difficult to solve
  - Small platforms, lots of devices
  - Symmetric key scheme: manageability?
  - PKI: not even in the Internet world fully there
  - PKI on small devices?

# Security & Privacy

- “Good question!” . . . “Next question, please”
- Authentication, authorization, protection of data & messages
- Difficult to solve
  - Small platforms, lots of devices
  - Symmetric key scheme: manageability?
  - PKI: not even in the Internet world fully there
  - PKI on small devices? How to check certificates?

# Security & Privacy

- “Good question!” . . . “Next question, please”
- Authentication, authorization, protection of data & messages
- Difficult to solve
  - Small platforms, lots of devices
  - Symmetric key scheme: manageability?
  - PKI: not even in the Internet world fully there
  - PKI on small devices? How to check certificates?
- How to model “trust”?

# Security & Privacy

- “Good question!” . . . “Next question, please”
- Authentication, authorization, protection of data & messages
- Difficult to solve
  - Small platforms, lots of devices
  - Symmetric key scheme: manageability?
  - PKI: not even in the Internet world fully there
  - PKI on small devices? How to check certificates?
- How to model “trust”?
- No convincing approach yet

# Routing

- **MANET Routing protocols**

# Routing

- **MANET Routing protocols**
  - **Proactive** or **table driven:**
    - ◆ Periodical updating of routing tables
    - ◆ Every node up-to-date
    - ◆ constant & heavy routing information exchange

# Routing

- **MANET Routing protocols**
  - **Proactive** or **table driven:**
    - ◆ Periodical updating of routing tables
    - ◆ Every node up-to-date
    - ◆ constant & heavy routing information exchange
  - **Reactive** or **on-demand:**
    - ◆ Update only when routing fabric changes
    - ◆ lot of overhead to determine route

# Routing

- **MANET Routing protocols**
  - **Proactive** or **table driven:**
    - ◆ Periodical updating of routing tables
    - ◆ Every node up-to-date
    - ◆ constant & heavy routing information exchange
  - **Reactive** or **on-demand:**
    - ◆ Update only when routing fabric changes
    - ◆ lot of overhead to determine route
- DIRC networking (2)

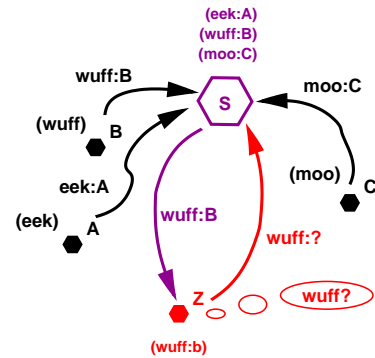


# Routing

- **MANET Routing protocols**
  - **Proactive** or **table driven:**
    - ◆ Periodical updating of routing tables
    - ◆ Every node up-to-date
    - ◆ constant & heavy routing information exchange
  - **Reactive** or **on-demand:**
    - ◆ Update only when routing fabric changes
    - ◆ lot of overhead to determine route
- DIRC networking (2)
- **Security!?** (3)

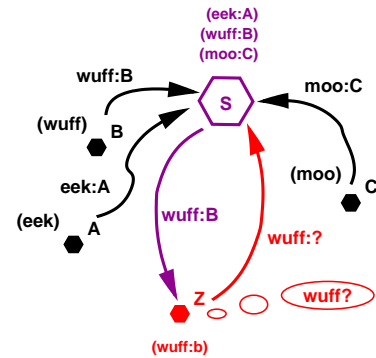
# Case Study: Jini

- Centralized pull model



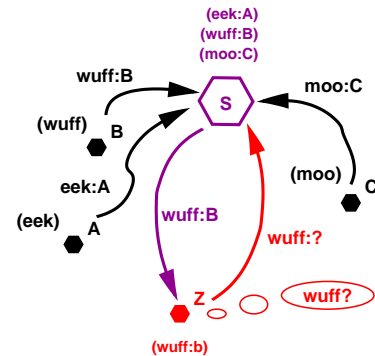
# Case Study: Jini

- Centralized pull model
- Service: **Java proxy object**



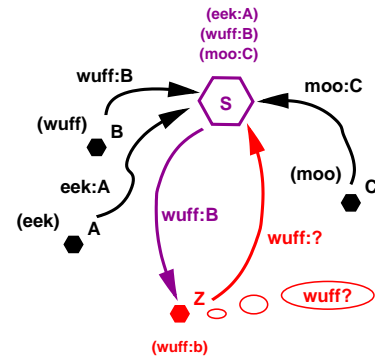
# Case Study: Jini

- **Centralized pull model**
- Service: **Java proxy object**
- Service description: **Java interfaces**



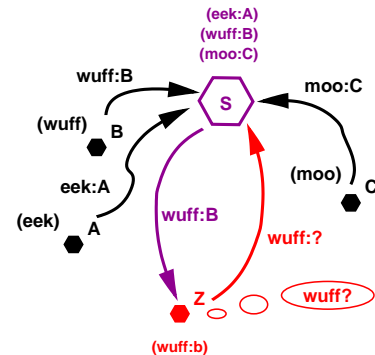
# Case Study: Jini

- **Centralized pull model**
- Service: **Java proxy object**
- Service description: **Java interfaces**
- Proxy object encapsulates invocation contract



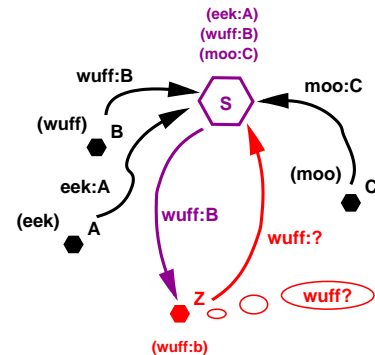
# Case Study: Jini

- **Centralized pull model**
- Service: **Java proxy object**
- Service description: **Java interfaces**
- Proxy object encapsulates invocation contract
- Event notification supported



# Case Study: Jini

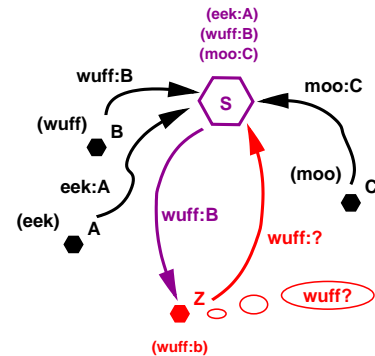
- **Centralized pull model**
- Service: **Java proxy object**
- Service description: **Java interfaces**
- Proxy object encapsulates invocation contract
- Event notification supported



- Well-advertized, some support, **restrictive JCP license, tied to Java runtime**

# Case Study: Universal Plug and Play (UPnP)

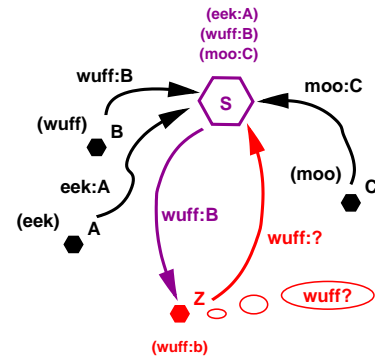
- Hybrid: **Centralized& distributed pull model**





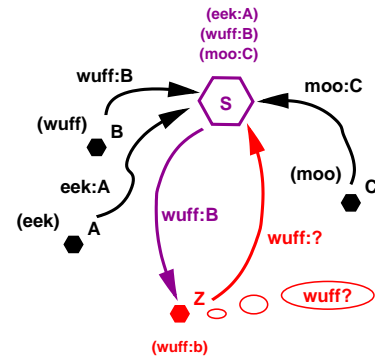
# Case Study: Universal Plug and Play (UPnP)

- Hybrid: **Centralized& distributed pull model**
- Service: **SOAP**



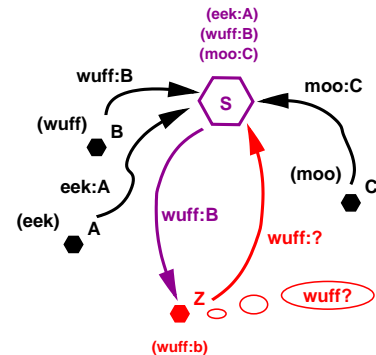
# Case Study: Universal Plug and Play (UPnP)

- Hybrid: **Centralized& distributed pull model**
- Service: **SOAP**
- Server: “proxy”



# Case Study: Universal Plug and Play (UPnP)

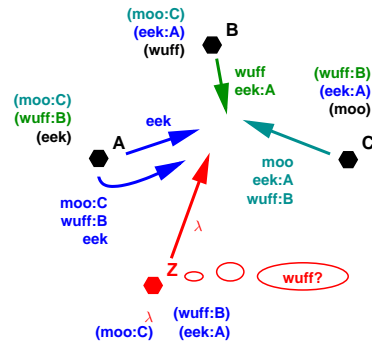
- Hybrid: **Centralized& distributed pull model**
- Service: **SOAP**
- Server: “proxy”



- Microsoft initiated; but IETF working group: **open standard**

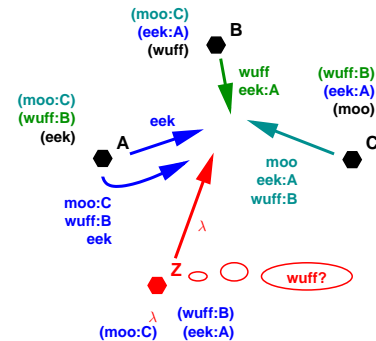
# Case Study: DEAPspace

- Transient Ad-Hoc Networking



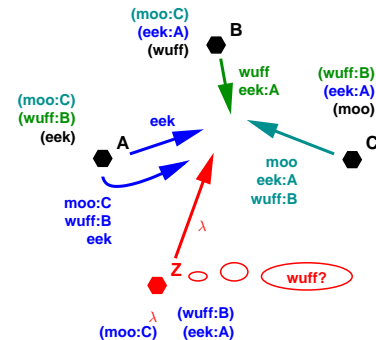
# Case Study: DEAPspace

- Transient Ad-Hoc Networking
- Service: Processor concept



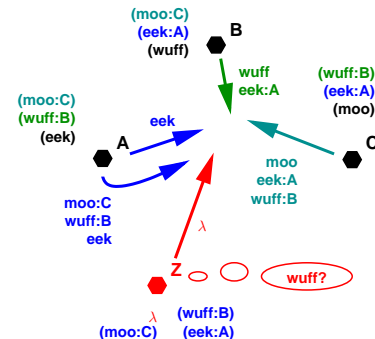
# Case Study: DEAPspace

- **Transient Ad-Hoc Networking**
- Service: **Processor concept**
- Service description: **Textual, XML, tree-based, MIME extensions**



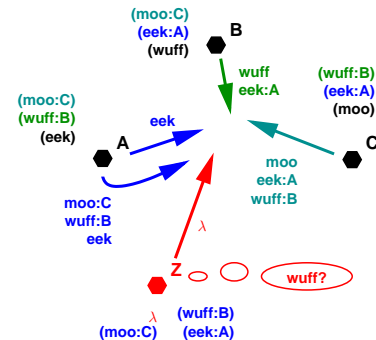
# Case Study: DEAPspace

- **Transient Ad-Hoc Networking**
- Service: **Processor concept**
- Service description: **Textual, XML, tree-based, MIME extensions**
- Service discovery: **Push model**



# Case Study: DEAPspace

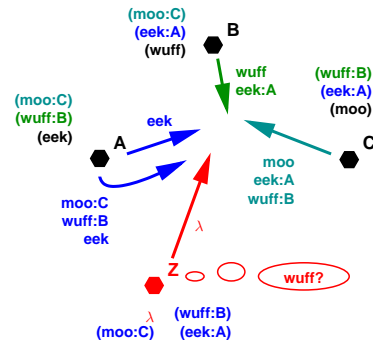
- **Transient Ad-Hoc Networking**
- Service: **Processor concept**
- Service description: **Textual, XML, tree-based, MIME extensions**
- Service discovery: **Push model**
- Research vehicle implemented at IBM Zurich Research Lab
  - Analytical, simulator, emulator





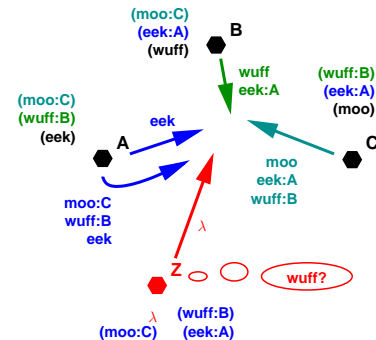
# Case Study: DEAPspace

- **Transient Ad-Hoc Networking**
- Service: **Processor concept**
- Service description: **Textual, XML, tree-based, MIME extensions**
- Service discovery: **Push model**
- Research vehicle implemented at IBM Zurich Research Lab
  - Analytical, simulator, emulator
- Result: Push-model more efficient than pull



# Case Study: DEAPspace

- **Transient Ad-Hoc Networking**
- Service: **Processor concept**
- Service description: **Textual, XML, tree-based, MIME extensions**
- Service discovery: **Push model**
- Research vehicle implemented at IBM Zurich Research Lab
  - Analytical, simulator, emulator
- Result: Push-model more efficient than pull
- Result: RF support for low-bandwidth information exchange not there (yet?)



# Ad Hoc PvC Applications?

- Standard application:

# Ad Hoc PvC Applications?

- Standard application: **Build a better mouse trap**

# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**

# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**
  - Ubiquitous: coffee machines

# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**
  - Ubiquitous: coffee machines
  - Sensors, machine monitoring

# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**
  - Ubiquitous: coffee machines
  - Sensors, machine monitoring
  - Resource discovery in LAN



# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**
  - Ubiquitous: coffee machines
  - Sensors, machine monitoring
  - Resource discovery in LAN
  - All: Long term ad hoc

# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**
  - Ubiquitous: coffee machines
  - Sensors, machine monitoring
  - Resource discovery in LAN
  - All: Long term ad hoc
- Transient ad hoc:

# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**
  - Ubiquitous: coffee machines
  - Sensors, machine monitoring
  - Resource discovery in LAN
  - All: Long term ad hoc
- Transient ad hoc: ???

# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**
  - Ubiquitous: coffee machines
  - Sensors, machine monitoring
  - Resource discovery in LAN
  - All: Long term ad hoc
- Transient ad hoc: ???
  - Ideas, scenarios, first steps

# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**
  - Ubiquitous: coffee machines
  - Sensors, machine monitoring
  - Resource discovery in LAN
  - All: Long term ad hoc
- Transient ad hoc: ???
  - Ideas, scenarios, first steps
  - RF technology lacking

# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**
  - Ubiquitous: coffee machines
  - Sensors, machine monitoring
  - Resource discovery in LAN
  - All: Long term ad hoc
- Transient ad hoc: ???
  - Ideas, scenarios, first steps
  - RF technology lacking
- Multi device applications:

# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**
  - Ubiquitous: coffee machines
  - Sensors, machine monitoring
  - Resource discovery in LAN
  - All: Long term ad hoc
- Transient ad hoc: ???
  - Ideas, scenarios, first steps
  - RF technology lacking
- Multi device applications:
  - SyncML & Bluetooth/802.11 (?)

# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**
  - Ubiquitous: coffee machines
  - Sensors, machine monitoring
  - Resource discovery in LAN
  - All: Long term ad hoc
- Transient ad hoc: ???
  - Ideas, scenarios, first steps
  - RF technology lacking
- Multi device applications:
  - SyncML & Bluetooth/802.11 (?)
- PvC Scenarios



# Ad Hoc PvC Applications?

- Standard application: **Build a better remote control**
  - Ubiquitous: coffee machines
  - Sensors, machine monitoring
  - Resource discovery in LAN
  - All: Long term ad hoc
- Transient ad hoc: ???
  - Ideas, scenarios, first steps
  - RF technology lacking
- Multi device applications:
  - SyncML & Bluetooth/802.11 (?)
- PvC Scenarios ???



# Wrapping Up

- Ad hoc pervasive computing only slowly becoming a reality

# Wrapping Up

- Ad hoc pervasive computing only slowly becoming a reality
- Wireless networking technology *sine qua non* for ad hoc PvC

# Wrapping Up

- Ad hoc pervasive computing only slowly becoming a reality
- Wireless networking technology *sine qua non* for ad hoc PvC
- . . . however, not yet suited for ad hoc PvC
- Service discovery & service description important for efficiency

# Stay Tuned...

- Next in our trilogy:

# Stay Tuned...

- Next in our trilogy: **Current Pervasive Computing Research**
  - CoolTown, PEN, i-wear: Intelligent Clothing, DABWeb...

# Thank You...



**Thank You...**

**... And “See you later!”**

# References

- (1) Britannica.com, "Merriam Webster's Collegiate Dictionary: Main Entry: Ad hoc." [World Wide Web site](#). Cited 23 November 2000.
- (2) DIRC Technology GmbH & Co. KG, "DIRC: Digital Inter Relay Communication." [World Wide Web site](#). Cited 20 November 2000.
- (3) V. Kärpijoki, "Signalling and routing security in mobile and ad-hoc networks." [World Wide Web site](#), May 2000. Cited 23 November 2000.

# Credits

I am indebted to my colleagues [Reto Hermann](#) (for the compilation of characteristics of wireless network technologies) and [Christian Rohner](#) for the comparison of Jini and UPnP.

Created with pdfLaTeX, FoilTeX, XFig, hyperref and PPower4.