

Technische Universität Braunschweig  
Institut für Betriebssysteme und Rechnerverbund

Seminararbeit

Service Discovery in Ad-hoc-Netzen

von  
Stephan Gruschwitz

**Betreuung:**

M. Bechler

Braunschweig, den 24. Januar 2003

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>3</b>
<b>2</b>	<b>Service Location Protocol</b>	<b>3</b>
<b>3</b>	<b>Universal Plug and Play</b>	<b>6</b>
<b>4</b>	<b>Salutation</b>	<b>9</b>
<b>5</b>	<b>Fazit</b>	<b>11</b>
<b>6</b>	<b>Zusammenfassung</b>	<b>12</b>

# 1 Einleitung

Ad-hoc-Netze sind Netze, die ausschließlich von drahtlosen und mobilen Geräten gebildet werden. Diese Netze können auf keine Infrastruktur zurückgreifen und müssen sich daher selbst organisieren. [12]

Die Anzahl von mobilen Geräten und damit von Ad-hoc-Netzen hat in den letzten Jahren stark zugenommen und dieser Trend wird wahrscheinlich auch weiterhin anhalten. Damit ein Nutzer nun verschiedene Dienste in "fremden" Umgebungen nutzen kann, benötigt er gewisse Informationen (welche Services, welches Übertragungsprotokoll, ect.) über die Dienste, die in seiner Umgebung (Netzwerk) angeboten werden. Damit er nicht selber nach diesen Informationen suchen muss, sondern die Ressourcen einfach nutzen kann, gibt es Service Discovery Protocols (SDPs), wie z.B. das Service Location Protocol, Universal PlugAndPlay und Salutation. Diese übernehmen für den Nutzer das Suchen nach den angebotenen Diensten und die Konfiguration, damit auf selbige ohne weitere Einstellungen zugegriffen werden kann.

Die folgenden Beispiele verdeutlichen, welche Vorteile durch SDPs entstehen bzw. in welchen Situation sie dem Nutzer das Leben angenehmer gestalten können.

- Beispiel 1: Ein Vertreter besucht einen Kunden in dessen Büro. Dort will er seine neuen Produkte und deren technischen Daten vorstellen, welche auf seinem Handheld gespeichert sind. Da sein Handheld sowohl eine Wireless-Netzwerkkarte hat als auch UPnP unterstützt, findet es automatisch den Drucker des Kunden. Der Vertreter kann nun alle Unterlagen auf dem Drucker bequem ausdrucken. [1]
- Beispiel 2: Stellen Sie sich einen tragbaren Overhead-Projektor mit einem Bücherei-Client vor. Nach der Identifikation an dem System, kann ein Nutzer sich ein Dokument aussuchen, welches dann über den Projektor dargestellt wird. Dadurch ist es möglich, dass ein Nutzer nicht viele Folien zu einer Besprechung mitbringen muss, sondern einfach über den LAN-Server auf die Bücherei direkt zugreifen kann. [1]

In diesem Dokument wird zuerst die Funktionalität der drei schon erwähnten SDPs für Ad-hoc-Netze beschrieben. Anschließend werden diese Protokolle gegenübergestellt, um so ihre Vor- und Nachteile besser vergleichen zu können.

## 2 Service Location Protocol

Das Service Location Protocol (SLP) ist ein Standard der Internet Engineering Task Force (IETF), um netzwerkbasierte Dienste automatisch zu lokalisieren und konfigurieren. SLP benutzt URLs, um eine Dienststart und deren Lokalisation zu beschreiben. Beispielsweise ist "service:printer:lpr://hostname" die Dienst-URL für einen Druckservice, der unter "hostname" anzusprechen ist.

Für die Kommunikation zwischen und das Suchen von Diensten unterscheidet SLP die Anwendungen in drei Hauptelemente, den User Agent (UA), den Service Agent (SA) und den Directory Agent (DA).

- User Agent: Der User Agent ist eine Software, die auf Anfrage (z.B. eines Anwendungsprogramms) “Service Requests” sendet. Ein UA sucht also für eine Anwendung die benötigten Dienste.
- Service Agent: Service Agents machen die Position und die Eigenschaften eines Dienstes im Netz bekannt.
- Directory Agent: Ein Directory Agent “verwaltet” alle Dienste in einem Netzwerk (Abb. 1). Wenn ihn ein “SA Advertisement” erreicht, registriert er den Dienst und

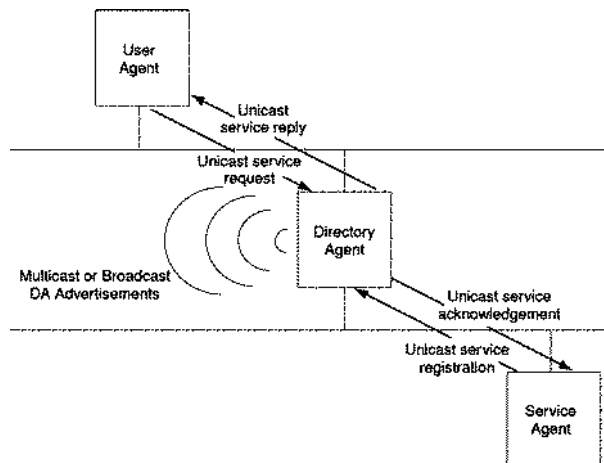


Abbildung 1: SLP Kommunikation

schickt dem SA ein “Service Acknowledgment”. Um zu verhindern, dass ein Dienst noch registriert ist, obwohl er nicht mehr verfügbar ist, muss sich jeder Dienst nach einer gewissen Zeit erneut registrieren. Registriert sich ein Dienst nicht wieder, geht der DA davon aus, dass der Dienst nicht mehr im Netz vorhanden ist und löscht ihn aus seinem Register. “Service Requests” werden von einem DA mit den entsprechenden Dienst-URLs beantwortet.

Ein DA ist optional. Ist kein DA vorhanden, kommunizieren UAs und SAs direkt miteinander (Abb. 2). Ein UA sendet solange Multicast-Nachrichten mit seiner Anfrage, bis ein SA antwortet. Ein SA, der eine Anfrage empfängt, die auf einen seiner Dienste zu trifft, antwortet mit einer Unicast-Nachricht an den UA. SAs schicken in regelmäßigen Abständen Multicast-Nachrichten, so dass UAs immer neue Dienste kennen lernen können.

Für die Kommunikation zwischen den einzelnen Agenten sind folgende Nachrichten vorgesehen:

- Service Request: Ein Service Request wird von einem UA gesendet, um einen Dienst zu lokalisieren. Die Nachricht enthält einen String mit dem gesuchten Dienst, z.B. “service:printer”. Für die Suche nach einem passenden Service ist es möglich, Einschränkungen, wie z.B. “AND”, “OR”, und Vergleiche ( $=$ ,  $<$ ,  $\leq$ ,  $>$ ,  $\geq$ ), einzugeben.

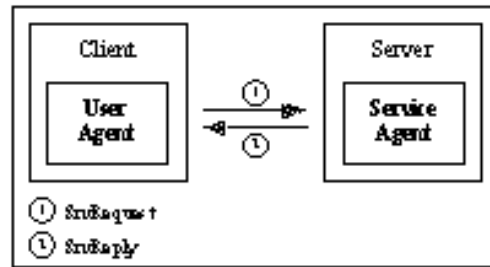


Abbildung 2: SLP ohne DA

- Service Reply: Enthält die Dienst-URLs der Dienste, die auf die Anfrage “Service Request” zutreffen. Ein Service Reply wird von einem DA oder SA gesendet.
- Service Registration: Ein SA sendet eine Service Registration, um sich bei einem DA zu registrieren. Die Nachricht enthält die Dienst-URL.
- Service Acknowledgment: Ein DA sendet ein Service Acknowledgment als Bestätigung auf eine Service Registration.
- Directory Agent Advertisement: Directory Agent Advertisements werden von DAs gesendet, um ihre Dienst-URL bekannt zu geben.
- Service Agent Advertisement: Service Agent Advertisements werden von SAs gesendet, um ihre Anwesenheit im Netz bekannt zu geben und ihre Dienst-URL zu veröffentlichen.
- Service Type Request (optional): Durch einen Service Type Request kann ein UA alle Dienstypen in einem Netzwerk erfragen.
- Service Type Reply (optional): Ein Service Type Reply ist die Antwort auf einen Service Type Request.
- Attribute Request (optional): UAs können mit einem Attribute Request die Attribute eines Dienstes erfragen.
- Attribute Reply (optional): Ein SA antwortet auf ein Attribute Request mit einem Attribute Reply, welcher eine Liste der Attribute enthält.
- Service Deregister (optional): Ein SA kann sich durch ein Service Deregister bei einem DA deregistrieren.

Um SLP-Netzwerke besser verwalten zu können, wurden so genannte Scopes eingeführt. Durch Scopes ist es möglich, die einzelnen Dienste zu gruppieren. So kann man z.B. Drucker A dem Scope “zwölfteEtage” und Drucker B dem Scope “achteEtage” zuweisen. Anschließend können die UAs so konfiguriert werden, dass alle UAs der Mitarbeiter aus der zwölften Etage auf Drucker A und alle Mitarbeiter aus der achten Etage auf Drucker B zugreifen.

### 3 Universal Plug and Play

Universal Plug and Play (UPnP) baut für die automatische Verbindung bzw. Kommunikation verschiedener Anwendungen auf bekannte Standards wie z.B. TCP/IP, HTTP und XML auf. Wie schon aus dem herkömmlichen "Plug and Play" bekannt, sollen UPnP-Anwendungen direkt nach dem Einschalten ohne weitere Konfiguration funktionsfähig sein: Ein dem Netzwerk neu hinzugefügtes Gerät bekommt eine IP-Adresse, stellt auf Anfrage seine Dienste zur Verfügung und sucht nach anderen Geräten und deren Anwendungen. Die IP-Adresse wird durch Auto-IP bestimmt. Dabei wird zuerst versucht, die IP-Adresse durch einen DHCP-Server zu erhalten. Ist kein DHCP-Server vorhanden, wird die IP-Adresse zufällig gewählt und durch eine ARP-Anfrage getestet, ob sie schon vergeben ist.

UPnP weist allen Diensten in einem Netz eine der beiden folgenden Rollen zu: "Control Points" oder "Device Services" (Abb. 3) .

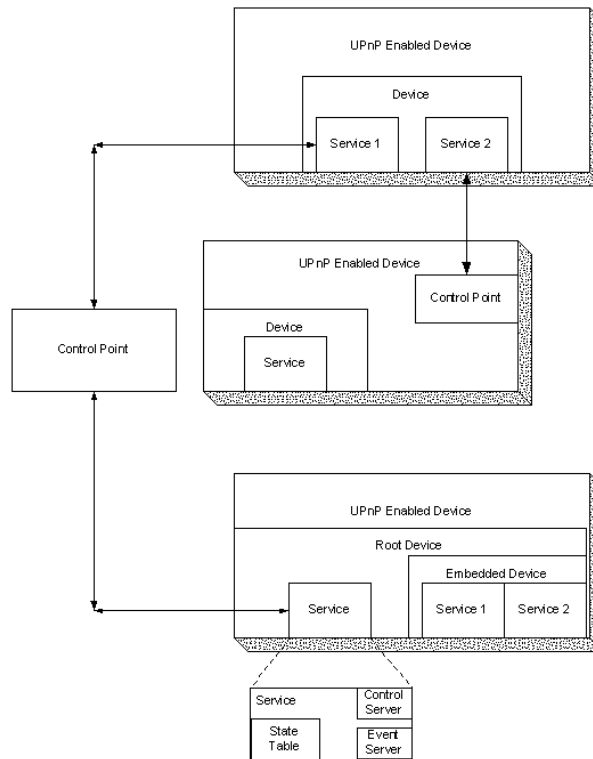


Abbildung 3: UPnP

- Control Points: Ein Control Point ist eine Anwendung, die Geräte- und Servicebeschreibungen abrufen, Anweisungen an Dienste schicken und Ereignisse von Diensten entgegennehmen. Geräte, die es dem Nutzer erlauben, Dienste auf ihnen auszuführen (wie z.B. ein CD-Spieler) werden als Control Points bezeichnet.

- Device Services: Ein Device Service ist eine funktionale Einheit. Jede Anwendung, die in dem Netzwerk einen Dienst erbringt, ist ein Device Service.

UPnP verwendet für die Service Discovery das “Simple Service Discovery Protocol” (SSDP) [2]. SSDP ermöglicht es, Geräten in einem UPnP-Netzwerk ihre Dienste bekannt zu geben und nach fremden Diensten zu suchen. Hierfür verwendet SSDP multi- und unicast HTTP-Verbindungen (HTTPMU bzw. HTTPU). Es wird XML verwendet, um die Fähigkeiten der einzelnen Dienste/Geräte zu beschreiben.

Eine typische UPnP-Kommunikation besteht aus den folgenden Elementen: Addressing, Discovery, Description, Control, Eventing und Presentation.

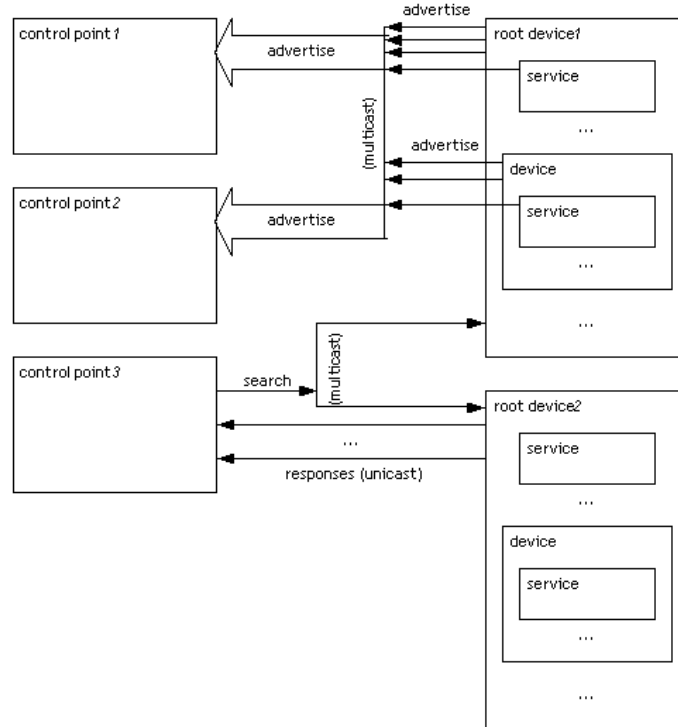


Abbildung 4: UPnP Discovery

- UPnP Addressing: Um einem UPnP-Netzwerk “beizutreten”, benötigt jedes Gerät eine IP-Adresse. Diese erhält das Gerät, wie schon beschrieben, durch Auto-IP.
- Discovery: Dienste müssen sich bei den Control Points bekannt machen. Hierzu sendet ein Gerät eine Multicast-ssdp:notify-Nachricht, in welcher es seine Dienste den Control Points bekannt macht.  
Wenn ein neuer Control Point in einem Netzwerk erreichbar ist, sendet er ein Multicast-ssdp:m\_search, um nach verfügbaren Geräten und deren Diensten zu suchen. Diese Suchanfrage kann Einschränkungen enthalten, es kann z.B. nur eine ganz bestimmte Art von Diensten gesucht werden. Alle Geräte, die einen solchen

Dienst anbieten, antworten dem Control Point mit einem Unicast. Diese Antwort enthält eine URL, unter der eine genauere Beschreibung des Gerätes zu finden ist (Abb. 4).

- **Description:** Ist einem Control Point ein Gerät bekannt, weiß er immer noch relativ wenig über das Gerät. Um mehr über das Gerät zu erfahren, muss dessen Beschreibung analysiert werden. Die Beschreibung kann über eine URL, die in der "Discovery-Nachricht" enthalten ist, abgerufen werden. Für sie wird XML verwendet. Die Beschreibung enthält eine Liste von Diensten, die von dem Gerät angeboten werden und die URLs, über die welche Dienste kontrolliert werden (Control, Eventing, Presentation).
- **Control:** Wenn einem Control Point die Funktionsweise eines Gerätes bekannt ist, kann er "Control Messages" an die Control-URL des Dienstes senden. Die Control Nachrichten sind auch XML-Nachrichten, welche das Simple Object Access Protocol (SOAP) [9] nutzen. Wie bei Funktionsaufrufen üblich, antwortet ein Dienst mit einer spezifischen Nachricht auf eine Control Message (Abb. 5).

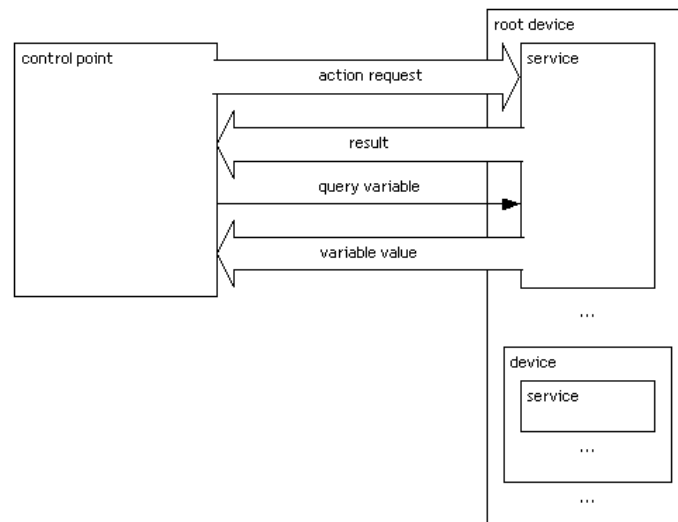


Abbildung 5: UPnP Control

- **Eventing:** Eine UPnP Beschreibung eines Gerätes enthält auch eine Liste von Variablen, die den Zustand (State) der Dienste beschreiben. Die Dienste verschicken eine Nachricht, wenn eine oder mehrere dieser Variablen geändert werden. Dieses geschieht über "Event Messages", die auch in XML geschrieben sind. Ein Control Point kann sich bei einem Dienst anmelden, um die Event Messages zu erhalten.
- **Presentation:** Die UPnP-Dienste bieten eine Presentation in HTML als Graphical User Interface (GUI) an. Durch eine "HTTP GET" Anfrage kann diese Seite geladen bzw. betrachtet werden.



## 4 Salutation

Salutation wurde von einem Konsortium (mehr als 30 Firmen, unter anderem Sun, Cisco, HP, IBM) entwickelt. Die Spezifikation ist frei verfügbar. Bei der Entwicklung von Salutation wurde besonders darauf Wert gelegt, dass es weder plattform- noch netzwerkabhängig ist. Salutation bietet Anwendungen und Geräten eine Möglichkeit, ihre Dienste zu beschreiben, und andere Anwendungen oder Geräten können so auch auf selbige zugreifen.

Die Architektur besteht aus zwei Hauptkomponenten: Dem Salutation Manager und dem Transport Manager. Dieses beiden “Manager” können in einer Schichtenansicht betrachtet werden, in welcher der Salutation Manager auf dem Transport Manager liegt (Abb. 6).

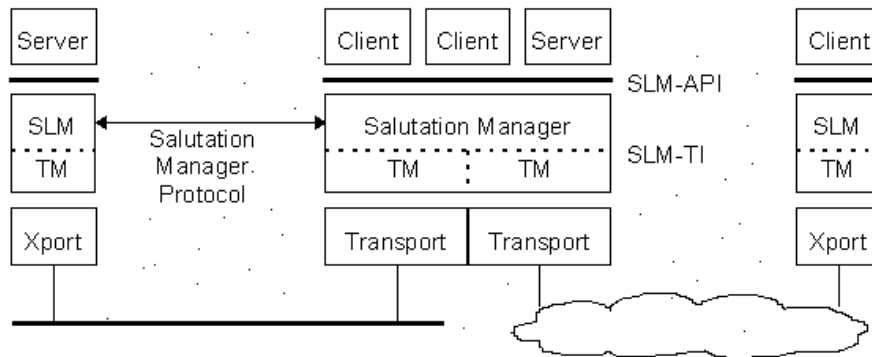


Abbildung 6: Salutation

- **Salutation Manager:** Der Salutation Manager kann als eine Art erweitertes Inhaltsverzeichnis gesehen werden. Wenn eine Anwendung einen Dienst bereitstellen will, registriert sie sich bei einem Salutation Manager. Ein Client fragt seinen Salutation Manager nach dem Dienst, den er nutzen möchte; die eigentliche Suche nach dem Dienst wird dann von dem Salutation Manager durchgeführt. Der Client kann dann “einfach” den zurückgelieferten Dienst nutzen. Der Salutation Manager bietet eine netzwerkunabhängige Kommunikationsschnittstelle zwischen Client und Server. Nicht jeder Client oder Server braucht einen Salutation Manager. Die Architektur ist so ausgelegt, dass ein Salutation Manager in “Umgebung A” als Proxy für einen Client in “Umgebung B” dienen kann (Abb. 7).
- **Transport Manager:** Der Transport Manager realisiert die Kommunikation über das Netzwerk, er ist also netzwerkabhängig. Aus diesem Grund kann ein Salutation Manager auch mehr als nur einen Transport Manager haben, um über verschiedene Netzwerktypen kommunizieren zu können. Ein Salutation Manager sendet Nachrichten also nicht direkt über ein Netzwerk, sondern sendet und empfängt alle Nachrichten über ein netzwerkunabhängiges Interface (SLM-LI) durch den Transport Manager.

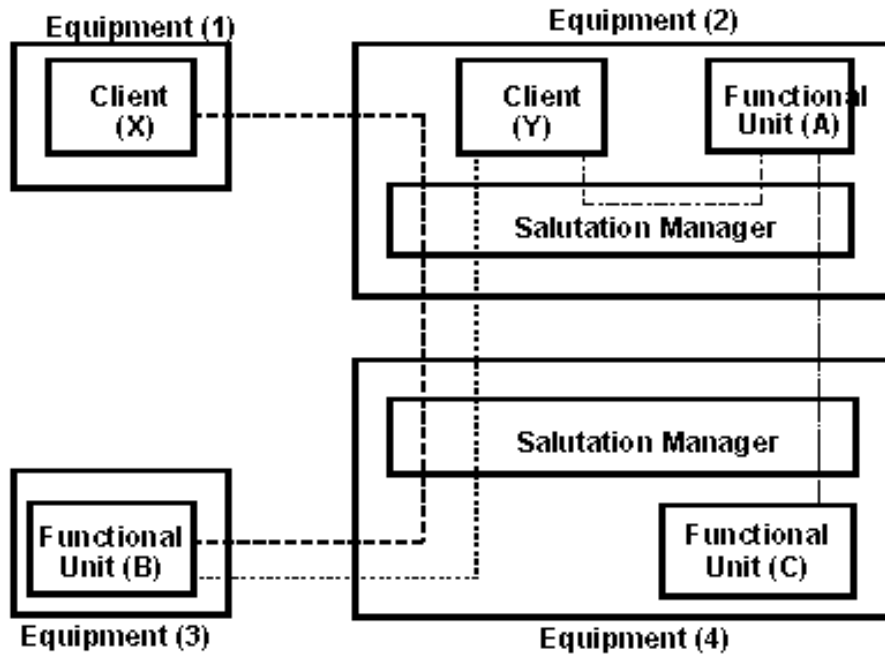


Abbildung 7: Salutation-Manager-Proxy

- “Functional Unit”: Functional Units wurden von dem Salutation Technical Committee (STC) definiert. Das Komitee hat verschiedene Geräte und Anwendungen untersucht und deren Grundfunktionen herausgearbeitet. Für jede Grundfunktion wurde eine Functional Unit definiert. Functional Units sind die kleinsten funktionalen Einheiten, die von Clients oder Servern (wie z.B. Printservice oder Scanservice) zur Verfügung gestellt werden. Für jede Functional Unit wird von den STC noch eine Reihe von Attributen definiert, um den Service besser zu beschreiben. Ein Drucker hat eine Print Functional Unit, und diese Functional Unit hat Attribute wie z.B. Pixel Density, Paper Size und Duplex.

Die Hauptaufgaben, die ein Salutation Manager behandelt, lassen sich durch Service Registration, Service Discovery, Service Availability und Service Session Management beschreiben. Der Salutation Manager verwaltet ein “Inhaltsverzeichnis”, in dem alle Informationen über die vorhandenen Dienste gespeichert sind. Alle Clients melden sich selber an bzw. ab. Diese Vorgänge werden Service Registration genannt. Um auf andere Dienste zu greifen zu können, sucht ein Salutation Manager beim Service Discovery nach anderen Salutation Managern und den dort verwalteten Diensten. Auf Anfrage eines Clients kann ein Salutation Manager in regelmäßigen Abständen die Verfügbarkeit einzelner Dienste prüfen. Eine Sitzung wird etabliert, wenn ein Client einen Dienst durch ein “Service Discovery” anfordert. Eine Sitzung kann auf drei unterschiedlichen Arten ablaufen: entweder im “Salutation Mode”, dem “Emulated Mode” oder dem “Native Mode”.

- Salutation Mode: Im “Salutation Mode” ist der Salutation Manager nicht nur für die Weiterleitung der Pakete verantwortlich, sondern definiert auch die Formate, die in dieser Sitzung verwendet werden.
- Emulated Mode: Im “Emulated Mode” leitet der Salutation Manager die Pakete nur weiter.
- Native Mode: Die Kommunikation zwischen Client und Server läuft über ein eigenes Protokoll, dadurch wird der Salutation Manager hierbei nicht weiter benötigt.

Der Vollständigkeit halber sei hier auch noch Salutation-Lite erwähnt. Hierbei handelt es sich um eine Salutation-Version, die für “kleinere” Geräte (wie z.B. Handhelds) optimiert wurde. Es eignet sich auch besonders für Übertragungsmedien mit kleiner Bandbreite, wie Infrarot oder Bluetooth.

## 5 Fazit

Nachdem die drei SDPs SLP, UPnP und Salutation kurz vorgestellt wurden, ist es nun möglich, sie vergleichend gegenüber zu stellen und so ihre Vor- und Nachteile zu analysieren.

SLP ist schon sehr gut beschrieben und durch die IETF standardisiert. Dadurch, dass es sowohl mit DAs als auch ohne sie funktionsfähig ist, lässt es sich sehr gut für kleine und große Netzwerke nutzen. Auch lassen sich durch das Einbinden neuer DAs relativ leicht verschiedene Netze zu einem zusammen schließen. Durch das Leasing-Konzept wird verhindert, dass Dienste, die das Netz verlassen und sich nicht abmelden, für immer und ewig als verfügbar gelten. Allerdings ist SLP auf ein TCP/IP-Netzwerk angewiesen.

Auch UPnP ist auf ein TCP/IP-Netzwerk angewiesen, da es unter anderem HTTP GET und POST Nachrichten versendet. Durch die Beschreibung der einzelnen Dienste in XML, hat UPnP aber eine sehr mächtige Sprache zur Beschreibung zur Verfügung. Mit Auto-IP besitzt UPnP eine einfache Möglichkeit zum Suchen einer IP-Adresse. Da jeder Control Point, wenn er seine Suchanfrage nicht spezifiziert, von jedem Device Service eine Antwort erhält, kann ein großes Datenaufkommen auftreten.

Der größte Vorteil von Salutation ist, dass es netzwerkunabhängig funktioniert. Es muss nur ein entsprechender Transport Manager vorhanden sein. SLP und UPnP sind hingegen auf ein TCP/IP-Netzwerk angewiesen. Leider unterstützt Salutation kein “Remote Event Notification”.

	SLP	UPnP	Salutation
Home Page	www.svrloc.org	www.upnp.org	www.salutation.org
Hersteller	IETF	Microsoft	Salutation Consortium
Lizenz	open source	open (für Mitglieder)	open source
Netzwerk Transport	TCP/IP	TCP/IP	unabhängig
Dienstverwaltung	Directory Agent (DA) (optional)	nein	Salutation Manager
Dienstbekanntgabe	Service Registration	ssdp:alive	Registrieren bei einem Salutation Manager
Dienstsuche	Über DA oder multicast an alle SAs	Über Control Point oder auf Advertisements warten	Über Salutation Manager
Dienste nutzen	Dienst URL	SOAP	Service Session Management
Service Beschreibung	Service Type und Attribute	in XML	Functional Unit und Attribute
Service Gruppen	Scopes	nein	nein
Event Notification	nein	“Service publishes event” wenn eine Variable einen neuen Wert erhält	Availability Checking
Leasing	nein	ja	ja
Sicherheit	IP basiert	IP basiert	Nutzeridentifizierung

## 6 Zusammenfassung

Service Discovery wird in den nächsten Jahren immer wichtiger werden, da die Zahl der mobilen Geräte zunehmen wird. Dadurch wird die Anzahl der Ad-Hoc-Netze stark wachsen. Ad-Hoc-Netze sind, wie der Name schon andeutet, keine statischen Netze, sondern unterliegen einer ständigen Veränderung: Geräte schließen sich dem Netz an und andere verlassen das Netz. Wegen dieses ständigen Wandels werden Service-Discovery-Protokolle benötigt, welche mit dieser Spontanität umgehen können: Dies sind Protokolle, die automatisch Dienste, welche in einem Netz vorhanden sind, lokalisieren und konfigurieren bzw. Dienste, die das Netz verlassen, aus der Liste der erreichbaren Services streichen. In dieser Arbeit wurden SLP, UPnP und Salutation, die bekanntesten Service Location Protocols für Ad-Hoc-Netze, kurz vorgestellt und ihre Vor- und Nachteile gegenüber gestellt.

Neben diesen hier beschriebenen Protokollen gibt es noch eine große Anzahl weiterer Service Discovery Protokolle. Es ist auch davon auszugehen, dass noch viele andere entwickelt

und implementiert werden. Daher ist es wichtig Protokolle zu entwickeln, die zwischen den verschiedenen SDPs vermitteln können. So kann gewährleistet werden, dass verschiedene Geräte miteinander kommunizieren können, auch wenn sie nicht die selben SDPs benutzen.

## Literatur

- [1] Lee, Choonhwa; Helal, Sumi: "Protocols for service discovery in dynamic and mobile networks". International Journal of Computer Research Volume 11, Number 1, 2002.
- [2] Goland, Yaron Y.; Cai, Ting; Leach, Paul; Gu, Ye; Albright, Shivaun: "Simple Service Discovery Protocol" IETF Draft draft-cai-ssdp-v1-03.txt, October 28, 1999. [http://www.upnp.org/download/draft\\_cai\\_ssdp\\_v1\\_03.txt](http://www.upnp.org/download/draft_cai_ssdp_v1_03.txt) [Stand: Dezember 2002].
- [3] MSDN Library: <http://msdn.microsoft.com/library/default.asp> [Stand: Dezember 2002].
- [4] Network Working Group: Guttman, E.; Perkins, C.; Veizades, J.: "Service Location Protocol, Version 2" RFC2608, IETF, June 1999.
- [5] Network Working Group: Guttman, E.: "Service Location Protocol, Modifications for IPv6" RFC3111, IETF, May 2001.
- [6] Project Miriam, SLP: <http://www.ipsi.fraunhofer.de/mobile/projects/miriam/slp/> [Stand: Januar 2003].
- [7] Salutation Consortium: <http://www.salutation.org> [Stand: Dezember 2002].
- [8] SLP-Homepage: <http://www.srvloc.org/> [Stand: Januar 2003]
- [9] Simple Object Access Protocol (SOAP) 1.1: <http://www.w3.org/TR/SOAP/> [Stand: Dezember 2002].
- [10] Universal Plug and Play Forum: <http://www.upnp.org/> [Stand: Dezember 2002].
- [11] UPnP NAT Traversal FAQ:  
<http://www.microsoft.com/windowsxp/pro/techinfo/planning/networking/natfaq.asp> [Stand: Dezember 2002].
- [12] Schiller, J.: "Mobilkommunikation - Techniken für das allgegenwärtige Internet", Addison-Wesley, 2000.
- [13] Yu, Mengrong: "Service Discovery Protokolle für Ad-Hoc Netze" Seminar zur Datenkommunikation und verteilten Systemen, [http://www-i4.informatik.rwth-aachen.de/content/teaching/seminars/sub/2002\\_ss\\_docs/adhoc\\_sdp.pdf](http://www-i4.informatik.rwth-aachen.de/content/teaching/seminars/sub/2002_ss_docs/adhoc_sdp.pdf) [Stand: Januar 2003].