Network Data Management – Usage (NDM-U) For IP-Based Services

Version 1.0

March 24, 2000

Preface

Contact

For general questions regarding this document and referrals to technical experts for detailed questions, please contact:

Steven A. Cotton
Project Manager/Chief Editor
IPDR, Inc.
E-mail: scotton@ipdr.org

BSS Group Editor
Ramzi Yehia
Daleen Technologies
E-mail: raye@daleen.com

Protocol Group Editor
Jeff Meyer
HP
E-mail: jeffm@cup.hp.com

Acknowledgements

The following member companies contributed materially to the creation of this document:

Charter Members
ACE*COMM
Amdocs
Andersen Consulting
Apogee Networks
Applied Protocol Engines
Aptis, Inc.
AT&T Labs
Clarent
Convergys Corporation
CSG Systems
Daleen Technologies
DST Innovis
HP
IP23
Narus, Inc.
Savera Systems Inc.
TeleStrategies, Inc.
XACCT Technologies

Supporting Members
Comptel Plc
Computer Generation, Inc.
MSI
MetraTech Corp.
Solect Technology Group
TCSI Corporation

Associate Members
Microsoft

Abstract

In order to develop the IPDR record format it is required that a framework be in place that formally classifies the “IP network and service elements” and “support systems.” Next, the relationship between such subsystems needs to be in place to determine the flow of information between components. Finally, the requirements of each subsystem must be determined in order to specify the type of “IP resource and service usage information” that will be exchanged.

This document proposes a reference model to satisfy the above requirements. Section 2 proposes a general IP support system framework. This section is intended to identify key components found in production IP networks, present the relationship between them, and define basic terminology. This section focuses on the network and service element layer (NSE), and illustrates the concepts with example ISP network infrastructures. Section 3 details the business requirements that the structures and systems defined in Section 2 must satisfy. Next, section 4 describes the information flow requirements between the NSE layer and the various support systems identified. Section 5 concludes with a proposed structure of the IPDR record for various IP-based services, and motivates this structure using the business requirements identified in section 3 and the technical requirements identified in section 4.

This revision of the document represents the state of the work at a point determined by the working groups to be useful for broader review and validation. Many issues have been identified for further work and are not fully addressed in this issue. Additionally, it is anticipated that domain experts will contribute via liaison relationships currently being established. The services represented in this version may be substantially altered once these contributions are considered, even if at the expense of backward compatibility.

Change History

1.0 WDA	12/14/1999 Working Draft Revision A - Circulated to Membership for Comment
1.0.WDB	1/17/2000 Working Draft Revision B - Circulated to Membership for Comment
1.0 RDA	3/6/2000 Review Draft Revision A – Circulated to Steering Committee for Comment

Table of Contents

Figures

Tables

1. Introduction

1.1. Purpose

This document is intended to specify technical information that is sufficient for practical implementations of interchange of usage data among service elements participating in the delivery of IP-based services, either within a single enterprise or across multiple enterprises.

The IPDR organization intends to submit this specification to selected accredited organizations for consideration as an approved standard.

1.2. Scope

This document is limited to the discussion of issues as defined by the mission statement of IPDR.org, namely:

> *Define an open, extensible record format for exchanging IP resource and service usage information (1) from IP network and service elements to any support system that uses this data (OSS, BSS, etc.), and (2) between such support systems.*

1.3. References

[1] Telecom Operations Map - Evaluations Version 1.1, GB910, *TeleManagement Forum (TMF - http://www.tmforum.org)*, April 1999.

[2] Extensible Markup Language (XML) 1.0, W3C Recommendation REC-xml-19980210, W3C (http://www.w3c.org), 10-February-1998.

[3] Open Settlement Protocol (OSP) for Inter-domain pricing, authorization, and usage exchange, RTS/TIPHON-03004.2, *European Telecommunications Standards Institute (ETSI - http://www.etsi.org)*, 1999.

1.4. Overview

This specification is divided into four major chapters:

- IPDR Reference Model - a definition of the abstract and operational relationships between entities involved in the generation, recording, storage, transport, and processing of usage attributes.
- Business Requirements - a definition of business requirements to be addressed by the protocol specification and specific scenarios for the major process flows anticipated in actual application.
- Protocol - the notation, data unit syntax, and dynamic procedures involved in the operation of the interfaces specified in the reference model.
- Structures - the specific usage and document attributes and collections of such attributes associated with services for which accounting occurs.

The Protocol and Structures chapters represent the specific design produced through analysis of the Business Requirements chapter, consistent with the Reference Model chapter.

1.5. Terminology

Term	Definition
Accounting	The process of collecting and analyzing **service** and **resource usage** metrics for the purposes of capacity and trend analysis, cost allocation, auditing, and billing, etc. Accounting management requires that resource consumption be measured, rated, assigned, and communicated between appropriate business entities.
Mediation	In view of Figure 5, Mediation refers to the combination of the logical entities IPDR recorder, IPDR transmitter, and IPDR store.
Resource	A quantifiable asset employed by a **Service Provider**, or on behalf of a **Service Provider** by another Service Provider, to fulfill a request of a **Service Consumer**. (Examples include: files, communications, goods, etc).
Roaming	Service usage initiated by a service consumer and provided by a service provider other than the one with which the service consumer have business relationship.
Service	Network and/or application operation that provides the **Service Consumer** with the requested **resource.**
Service Consumer	The beneficiary (human or system) of a **service**.
Service Element	Any element that is responsible for fulfilling a **Service Consumer** request. (Examples include: network equipment and system processes)
Service Provider	An enterprise that provides communications-based **services**.
Session	A set of related service usages; service usages may or may not be time based in the unit of measurement.
Usage	Consumption of **resources** and **services** by a **Service Consumer**.
Usage Attribute	A parameter whose value indicates some aspect of **usage** of a given **service** and/or **resource.**
Usage Event[1]	A **Service**-specific trigger resulting in the generation by a **Service Element** of a set of **Usage Attribute** values related to **Usage** specific to a given **Service Consumer**

[1] Because of legacy issues, Usage Events from many Service Elements will not conform to an IPDR specification or, in some cases, may never conform. To be considered a Usage Event the information presented or made available by inference from the Service Element must minimally contain attributes from some of the general attribute categories.

1.6. Glossary:

ANI - Automatic Number Identification
ASP - Application Service Provider
BSS - Business Support Systems
CRM - Customer Relationship Management
DSS - Decision Support Systems
DTD - Data Type Definition
EP - End Point
ESN - Electronic Serial Number
GK - Gate Keeper
IMSI - International Mobile Subscriber Identity
IP - Internet Protocol
IS - IPDR Store
ISP - Internet Service Provider
IT - IPDR Transmitter
NDM - Network Data Management
NSE - Network Service Element
OSS - Operations Support System
PSTN - Public Switched Telephone Network
QoS - Quality of Service
RADIUS – Remote Access Dial-In Usage Server
RAS - Remote Access Server
SC - Service Consumer
SE - Service Element
SP - Service Provider
TMF - TeleManagement Forum
TOM - Telecommunications Operations Map
UA - Usage Aggregators
UC - Usage Collectors
VoIP - Voice over IP
VPN - Virtual Private Network
XML - eXtensible Markup Language

2. IPDR Reference Model

The IPDR organization has adopted the Telecommunication Management Forum's (TMF) telecommunications operation map (TOM) for the purposes of motivating the functional role and interfaces of the IPDR specifications relative to operations support systems (OSS). We have chosen the TOM because it is a well-known, industry-accepted organizational model of telecommunications support systems used by carriers and service providers today. The TMF Model is useful as a model of typical systems, and as motivation for design decisions. However, the TMF Model itself is not part of IPDR, and the data structures and interfaces of IPDR may be used in systems that vary substantially from the TMF Model. See [1] for more details.

2.1. IPDR and the TMF Model

The TOM, shown in Figure 1, identifies the core operation support processes found in a production carrier business operation. The systems that implement the customer care, services development/operations and network/systems management processes each provide a well-defined set of services that enable a carrier to successfully deploy and manage telecommunications services. As the model shows, these systems are organized in a layered fashion. Thus, each component builds on the services provided at a lower layer (and possibly adjacent components) to deliver the required functionality.

The IPDR organization's charter is to facilitate the integration of IP-based network elements into billing, reporting and assurance systems. In particular, one key goal is to define a common usage record format and exchange protocol to facilitate the flow of usage information from IP network elements managers to support systems. In the TOM, the network data management (NDM) component (defined as part of the network and systems management processes) defines the device-independent collection mechanism for such purposes. As such, the work of this specification falls into the definition of the Network Data Management component.

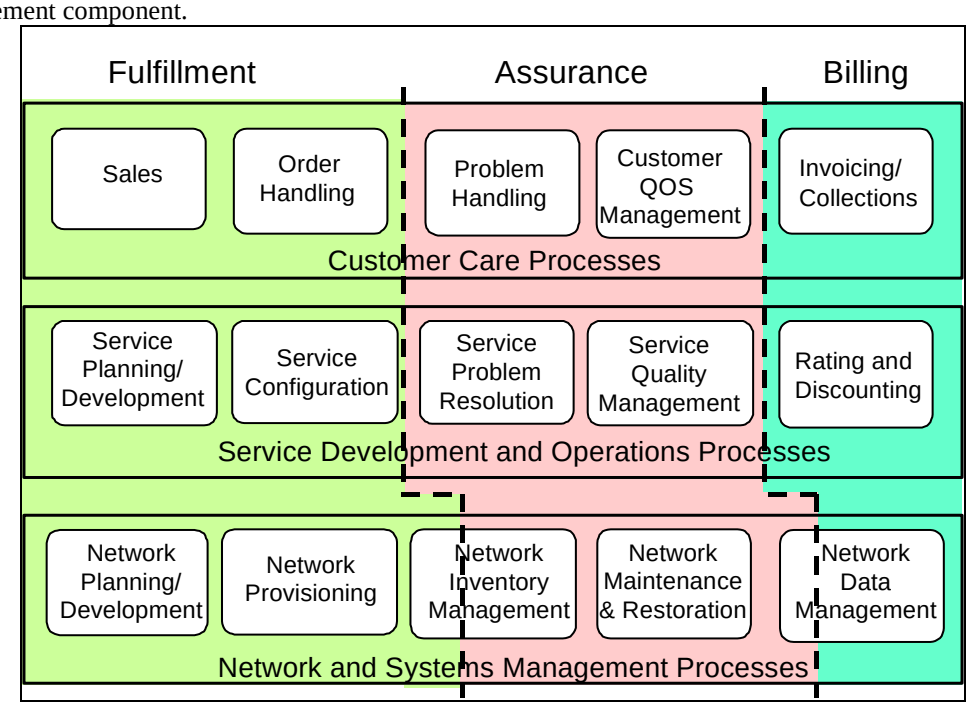

Figure 1 - Telecommunications Operations Map (TOM) of the TeleManagement Forum (TMF).

2.2. IPDR NDM-U High-level Model

Within the scope of the NDM-U module, the TOM shows interfaces to billing (i.e., rating and discounting) and customer care systems (i.e., service quality management and problem resolution). Likewise, the figure shows that the NDM-U component must interface directly with the network and service element manager to accomplish their various services.

The IPDR NDM-U reference model, shown in Figure 2[2], expands on the NDM-U definition by dividing the module into layers; namely: (1) the network and service element layer, (2) the mediation layer, and (3) the business support systems layer. Each layer is discussed below:

- Network and service element layer (NSE): The NSE layer consists of all the network and service elements required to provide an IP-based service to a given customer. For example, routers, access devices and transmission facilities together provide basic connectivity; firewalls might provide a security service; email, file and print servers provide application services; gateways provide a translation service from circuit to packet voice; and more. In addition to physical devices, the systems that configure and manage such devices are considered part of the NSE layer (note, that this functionality is identified as adjacent component within the "Network and Systems Management" layer in the TOM model). Examples here include a bandwidth management system, H.323 gatekeeper, RADIUS, authentication server or network management platform.

- Mediation layer: As shown in Figure 2, mediation systems sit between the network elements/infrastructure and the business support systems. Typically, a mediation system provides a single interface to BSS systems that provides all network usage data as well as a single service elements provisioning. In terms of usage collection, the goal of the mediation system is to capture all usage information required by the BSS systems, and export it within the temporal requirements. Thus, the mediation system must, in some way, determine the devices at the service element layer and interface with that infrastructure to extract the relevant usage information. The second mediation goal is to pass provisioning information from the BSS, to the network elements – again, within the temporal constraints.

- Business support systems (BSS) layer: The BSS layer consists of the systems deployed by a Service Provider or provider to support IP business operations. This layer corresponds to the "Systems Development and Operation Processes" in the TOM model. Some examples include billing (i.e., rating and discounting), customer care/relationship management, decision support, and market analysis and fraud detection. The BSS layer is the highest layer in the model. Thus, the BSS usage collection and provisioning requirements drive the mediation system and ultimately the services provided at the service element layer.

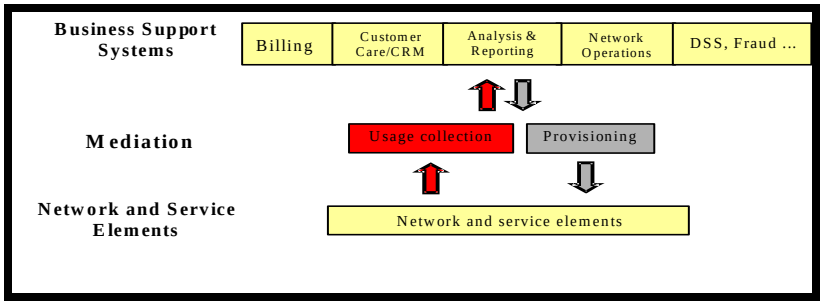

Figure 2 - IPDR NDM-U High-Level Model

[2] Note that both usage and provisioning flows are depicted in this figure. Only the usage component is discussed further in this issue.

The IPDR NDM-U model shown in Figure 2 gives a layered perspective of the components and interfaces designed to meet the NDM-U specification. The mission statement given in the introduction limits the organization's scope to the usage collection path (shown flowing upward in Figure 2). Thus, provisioning or the internal design of any of the identified components is not considered in this document.

The usage collection process represents a flow of usage data from the network and service elements to the BSS processes. Figure 3 illustrates the usage data path from network elements (e.g., gateways, remote access servers (RAS), routers, and bandwidth managers) to a mediation device. Note that this example assumes the interface between the mediation device and network elements is based on a proprietary access protocol, record format and API. The mediation system aggregates, normalizes and correlates the usage data as required, and exports the data to the billing, decision support, or other business support systems.

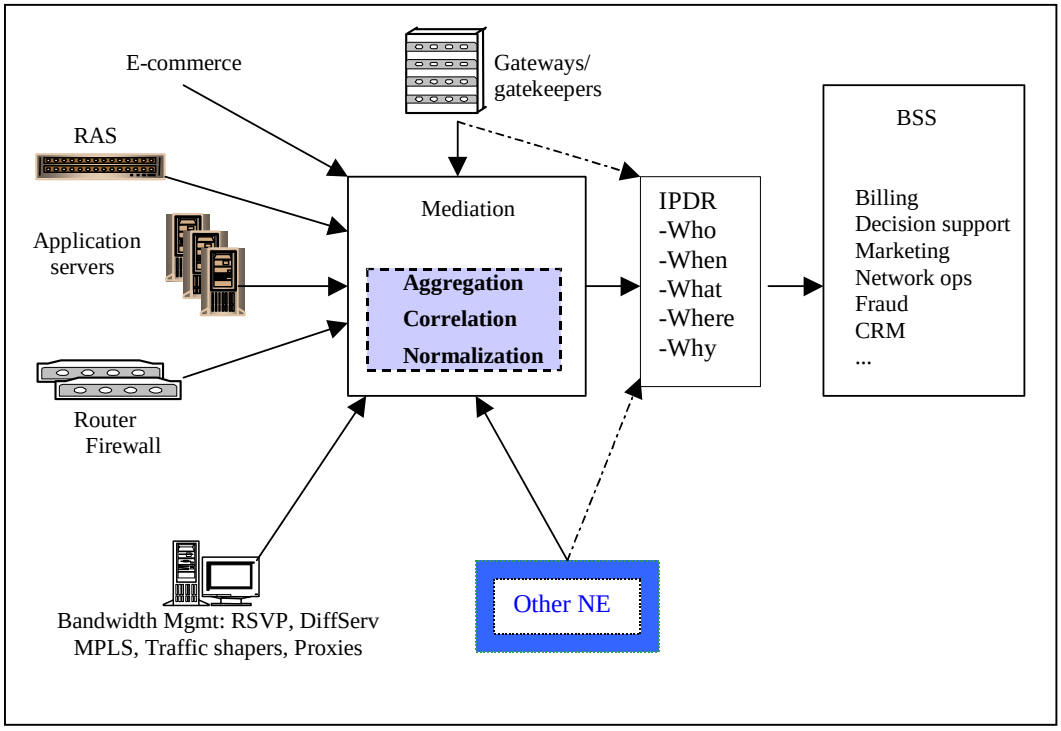

Figure 3 - IPDR Record Flow

The IPDR NDM-U plays several roles in this data transfer. First, the IPDR record provides flexible structure that is sufficiently powerful to describe the usage attributes collected by the mediation system, and required by the BSS system. Second, the IPDR NDM-U provides a set of interfaces that facilitate the exchange of IPDR records between mediation systems and BSS systems, or between IP network elements and BSS systems. Finally, the IPDR specification provides a set of interfaces that facilitate the intermediate storage of IPDR records between IPDR-enabled components.

2.3. IPDR Record Structure

The IPDR record structure must be capable of characterizing any type of usage that might be collected from an IP-based network or application service. As figure 4 shows, there are 5 components common to all IPDR records. Broadly, these components are the "who, what, where, when and why" values that describe a particular usage event. Each are discussed below:

- {Who} (Responsible for the usage)
  - User ID (in some form, if available)

- {When}
  - End Time or Event Time (This is the time that triggered the creation of this record.
    Note: It is arguable whether Start Time belongs here or is really more closely associated with either usage measures (duration) or state information as Start Time.)

- {What}
  - Service
  - Usage measures / quantities
  - Ex: Bytes, packets, flows, hits, transactions, time duration...
  - QoS measures
  - State information
  - Event code (logon, logoff, threshold exceeded)
  - Other information about state transition or current state (Start Time)[3]

- {Where}
  - Traceability / Context
  - Source Identifier
  - Destination Identifier
  - Service Element identifier (originator)

- {Why}
  - Event trigger type – (i.e., why is the network and service element reporting this data?)

In addition to the "5Ws" defined above, each record may include reference pointers to other IPDR records that either capture related usage information, or contain usage information that was used to create the given record.

[3] Note that "always on" services may be measured via periodic emission of IPDRs, recording usage since the last interval boundary.

2.4. IPDR Interfaces

In addition to the IPDR record structure, the IPDR specification defines a set of interfaces for exchanging IPDRs between IPDR-enabled devices or systems. Figure 5 shows the key interfaces and elements found within the NDM-U module, represented in an abstract form. Note that this model does not constrain implementations to be physically packaged as portrayed, nor to present all of the interfaces to other systems.

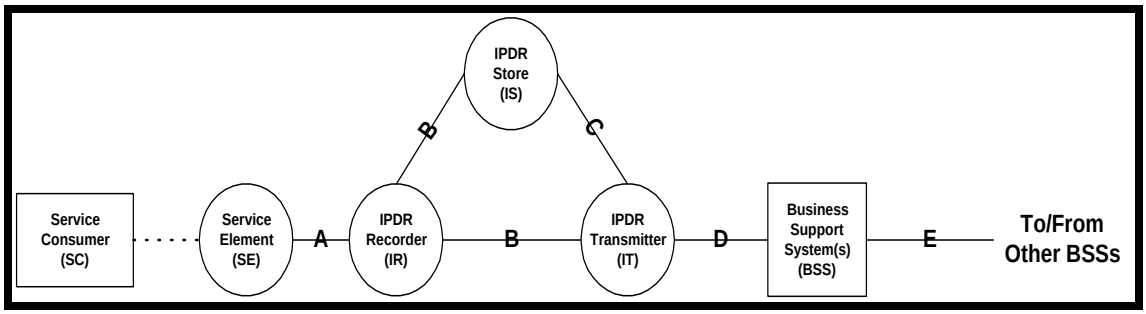

Figure 4 - Basic Network Model

Each element and the roles played by the various nodes in the model are given below:

- Service Consumer (SC) - requests and receives services (typically end user on end system)
- Service Element (SE) - provides access to services and resources requested, authenticates SC, authorizes access, performs accounting measurement for resources provided, provides services requested by SC, and performs sccounting measurement for services provided
- IPDR Recorder (IR) - packages usage information into IPDRs, optionally packages IPDRs into IPDR documents and presents the documents to an IS or an IT
- IPDR Transmitter (IT) - delivers IPDR documents to a BSS
- IPDR Store (IS) - retains IPDR documents from an IR in a non-volatile medium, presents such documents to an IT upon request
- Business Support System (BSS) - receives information contained in IPDR documents from an IT, processes the information contained in IPDRs for use in the commercial activities of a Service Provider, presents information for transmittal to other BSSs

The following describes the various interfaces in the model:

- A - for delivery of usage information from SEs to IRs[4]
- B - for delivery of IPDR documents from IRs to ISs and ITs
- C - for delivery of IPDR documents from ISs to ITs
- D - for delivery of IPDR documents from ITs to BSSs
- E - for delivery of IDPR documents from BSS to BSS

[4] Note that the A interface is not specified in detail in this document, but rather minimally constrained as to the basic behavior and content that is necessary to allow the recording of an IPDR compliant with BSS requirements.

3. Business Requirements

3.1. Introduction

For each IP service, this chapter provides high-level requirements, general and specific usage attributes, and use cases for BSS applications needs. It also provides the framework for specifying new IP services not yet covered in this chapter to meet the extensibility needs of the future.
Section 3.4 provides the general overview of the network model from previous chapters and it's applicability to the BSS applications needs. The main focus for specifying requirements is given to D interface though there may be implied requirements to other interfaces. Section 3.5 provides general requirements and general usage attributes applicable to any IP service. Note that any new IP service added in the future may impact the general requirements and usage attributes. Section 3.7 provides IP services covered and yet to be covered in this chapter. Section 3.9 provides the use cases for the services covered in this chapter. Section 3.9.1 provides details of ASP service, section 3.9.2 provides details of VoIP/FoIP service and section 3.9.3 provides details of basic E-mail service.

3.2. Notation

Use cases are used in this chapter to describe the environment of business requirements. To standardize the extension of those use cases a use case template is developed and included in appendix A.

3.3. Assumptions

1. Home Service Provider handles all business needs (via BSS applications) of the Service Consumer.
2. Service Consumer may or may not be within the Home Service Provider's service area.
3. BSS to BSS interfaces (E Interface) is outside the scope of this version of the NDM-U.
4. Some applications of NDM-U will result in large numbers of IPDRs being generated, requiring economical storage, transport, and processing implementations. Several requirements stated below are intended to address this assumption. However, no quantitative requirements regarding performance (end-to-end delay, transfer rate, etc.) or efficiency (message size, compression ratio, etc.) will be stated in this document. The mechanisms designed in later chapters of this document, which satisfy the general requirements in this area, should give implementers adequate tools to make cost versus technology tradeoffs, justified in light of the business problem being solved. Product vendors designing implementations of this specification are

assumed to be aware of the overall marketplace requirements for such systems and service providers selecting one or more of these implementations will be expected to require those vendors to demonstrate competitive features in the area of performance.

3.4. Network View

The interfaces described in the abstract network model for an IPDR system have been projected into a generic Network View as shown in Figure 5. This is done to allow various operations scenarios, some of which will involve complex relationships between SPs. The use cases will be expressed in terms of this Operations Model, thus allowing the specific service usage to be tied to the behavior at the specified interfaces in the Protocol chapter.

The Operations Model is segmented into five domains:

- Service Consumer - there are no IPDR NDM-U interfaces specified relating to this domain. The interface with SP1 is with an unspecified access interface, possibly over one or more categories of media (PSTN, cable network, PLMN, xDSL, ISDN, fixed wireless, mobile data, or a combination of one or more). The Service Consumer is not necessarily associated with SP1 from a BSS perspective in this model (i.e., a roaming scenario is possible).
- Service Provider 1 (SP1: Access/Transport/Application Services Provider) - this SP plays the unique role of providing access services, at a minimum. This SP may also provide transport and/or application services in given scenarios.
- Service Provider 2 (SP2: Transport/Application Services Provider) - this SP plays the role of transport services between SP1 and SP3, at a minimum. They may also provide application services in given scenarios.
- Service Provider 3 (SP3: Application Services Provider) - this SP provides only application services, relying on one or more access/network SPs for establishment of sessions with an SC.
- Service Provider 4 (SP4: BSS Services Provider) - this SP acts as an intermediate BSS on behalf of one or more other SPs. Such applications as service bureaus, clearing houses, rating bureaus, fraud bureaus, pre-paid authorization centers and other intermediate IPDR processing applications are examples of the role of this SP.

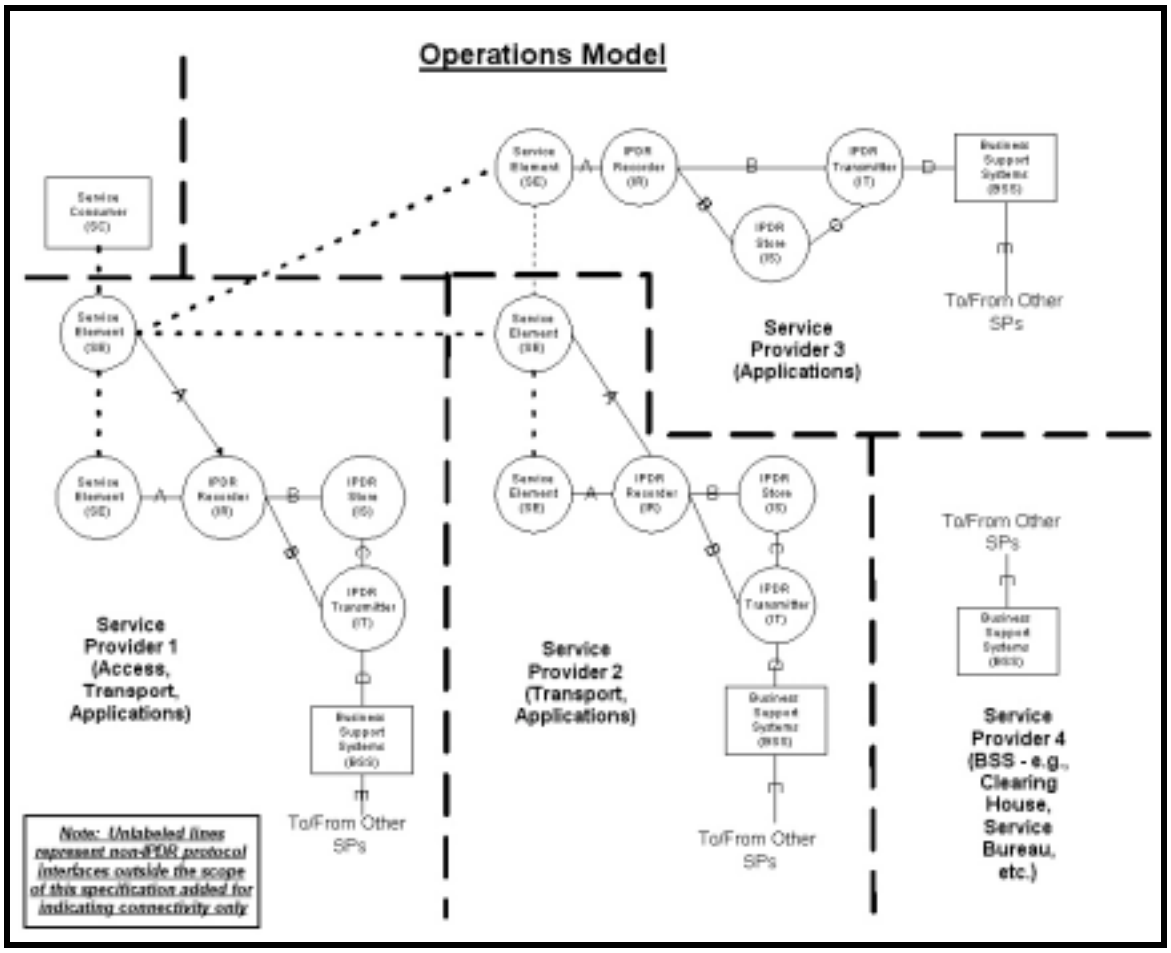

Figure 5 - Network view

3.5. General Requirements for the Operational Model

This section lists requirements that are service independent. That is, requirements that are not captured in the specific service use cases covered in section 3.9.

3.5.1. Mediation Requirements

The general requirements for mediation are, in almost all cases, service independent. Depending on the business model mediation tasks could span a wide variety of actions. However, in general terms mediation tasks include the collection, generation, aggregation and reconciliation of IPDRs across Service Elements, geographical areas and time.

1. Mediation shall support both polling and pushing for data transfer, so that the data transfer can be initiated either by the collector or spontaneously by the Mediation.

2. Mediation shall support data transfer for both individual events and batches of events.

3. Mediation shall support retrieval of IPDR documents.

4. Service elements shall be uniquely identified within the scope of each terminating IPDR recorder.

5. IPDR shall have a unique event identifier within service elements. If IPDRs are related and the relation is visible to the IPDR recorder (aggregator) then, a reference to the related record (base IPDR) shall contain this unique identifier.

6. IPDR shall enable the interim recording across multiple service elements and time. That is, enabling event information to exist in multiple records, over several IPDR documents.

7. Mediation shall support uniquely identifying IPDR documents for the purpose of gap and duplicate detection.

3.5.2. Format

This set of requirements pertains to the IPDR format.

1. The IPDR format shall be extensible permitting the addition of any set of services and service specific usage attributes.

2. The IPDR format shall be able to self-describe its usage attributes.

3. The IPDR format shall capture sufficient information to identify an IPDR service consumer.

4. The IPDR format shall provide specified data types, so that various systems can interpret the data properly.

3.5.3. Application Protocol

1. The IPDR protocol shall support encryption of IPDR documents.

2. The IPDR shall support efficient encoding.

3. The IPDR shall use open protocols and description languages.

4. IPDR protocol/format shall separate the record format and exchange protocol.

5. IPDR protocol shall support transfer capabilities negotiation.

6. IPDR protocol shall support both individual and batch transfers of data

7. IPDR protocol shall support resynchronization to a particular point in the order of delivery of IPDR documents.

3.5.4. Usage Attributes

1. The IPDR format specification shall indicate, for all usage attributes, if the information is required, optional or conditional.

2. The IPDR format specification shall indicate usage attributes data type.

3.5.5. Miscellaneous (Settlement)

1. The IPDR protocol and format shall support roaming.

2. The IPDR protocol and format shall support mobile service consumer.

3.6. Required Usage Attributes

This section includes a required set of usage attributes that is general across services. That is, those usage attributes apply to all the services and must exist for the correct operations of BSS systems. Additional IPDR document attributes may be required to support the protocol implementation. In that case, those document attributes will be specified in the Structures chapter.

Table 1

Category	Name	Remarks
What	Service_type	Type of service recorded
What	Number_of_Units	Number of units
What	Units_type	Type of units on the Number_of_Units – field; this could be CPU time, memory usage, disk usage.
When	Start_Time	Start time for service element usage
Where	Location	A unique identifier for the recording entity.
Where	Originating_point	Originating IP address
Where	Raw_Data_Reference	Identifying the original data source. Especially, in the case of aggregated records
Where	Terminating_point	Terminating IP address
Who	Service_Element_ID	A unique value identifying the service element for the event content.
Who	Session_ID	A unique reference number identifying the session
Who	User_ID	A value identifying uniquely the end-user
Why	Cause_for_termination	Reason for termination, if abnormal, possible causes (server reboot…)

3.7. Listing of Services

3.7.1. Services Covered

For describing the context environment of the business requirements listed in this chapter a set of services are analyzed then, for each service a multiple of use cases are depicted. The list of services considered in this chapter is a representative and not a comprehensive list. This list will be augmented through contributions by other relevant standard bodies and through the progress of IPDR organization work.

Services considered by the BSS working group in this version of the draft are:

1. Application Services (ASP)
2. Voice over IP (VoIP)
3. Fax over IP

4. E-mail services

3.7.2. Services for Future Consideration

Since the list of services considered in this version is not a comprehensive list, and recognizing the importance of other services we are including a list of services to be considered in future released of this specifications.

1. Video on Demand (VoD)
2. Access
3. Virtual Private Networks (VPN)
4. Conference Calling

3.8. Services Considered by other Organizations (References)

It is recognized that the specification of services requires expertise and experience in the providing or equipping such services. The IPDR encourages domain experts and service providers to submit specifications of services whose usage would be recorded by an IPDR recorder. The form of such submissions should conform to the templates and guidelines described in this chapter.

3.9. Use Cases

3.9.1. Application Service Provider

3.9.1.1. Service Definition

Application Service is the use of applications supplied by providers outside the service consumers business.

3.9.1.2. Service Requirements

1. IPDR must provide unique and clear identification of parties participating in the activity.
2. IPDR must provide information about the amount and type of resources used.

3.9.1.3. Service Usage Attribute List

Table 2

Category	Usage Attribute Name	Data Type	Presence	Possible Values	Remarks
What	Feature	String	Conditional	"Sort" as a feature of a "spreadsheet" app, etc.	Specific feature of service
What	Type	String	Conditional	"Front office service", "front office/word processor service", "front office/spreadsheet service", etc.	Type of application that is invoked.
When	AppRequestTime	Datetime	Required	ISO 8601 time	May be different from AppStartTime. This will allow measuring response time.
When	AppStartTime	Datetime	Required	ISO 8601 time	Time when application starts
Where	LoginLocation	String	Required		
Where	ProviderLocation	String	Required		Will support providers that host applications at different locations
Who	ProviderName	String	Required		Actual provider of the service
Who	UserLoginName	String	Required		Identifies a unique user in the system. Real time mapping of dynamically allocated IP addresses might be necessary

3.9.1.4. Front Office Service Use Case

Front office service provides ASP user with access to common office applications.

3.9.1.4.1. Basic Flow

1. User logs into an Application Service Provider (ASP).
2. After authentication, the user invokes a word processor application (front office service).
3. In the middle of using the word processor, the user invokes a spreadsheet application.
4. While still using the spreadsheet, application the user closes the word processor application.
5. The user closes the spreadsheet application.
6. The user logs out.

This could be considered as a single session with multiple events. Events are:

Start of session (at login)
Start of word processor
Start of Spreadsheet
End of word processor
End of spreadsheet
End of session

Another alternative is considering the word processor and the spreadsheet to be different services. As such, two separate records tracking identifiers will be produced one for each service.

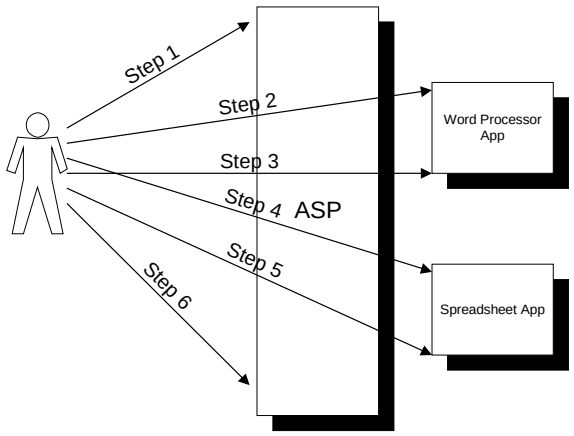

Figure 6. Front office use case: basic flow

3.9.1.4.2. Basic Flow Requirements

These are the basic flow requirements for the front office use case:

1. IPDR must provide information about the time that the event occurred.
2. IPDR must provide a correlation between the start and end events for the application.

3.9.1.4.3. Basic Flow Usage Attribute List

3.9.1.4.3.1. Service Definitions

Table 3

Category	**Usage Attribute Name**	**Data Type**	**Presence**	**Possible Values**	**Remarks**
What	AppActiveTime	Integer	Conditional	Milliseconds	Total elapsed active time for each process in the session. Active time is a measure of time when the CPU usage exceeds a certain percentage. At least one of the conditional usage attributes must be present.
What	AppLoadedTime	Integer	Conditional	Milliseconds	Total elapsed loaded time for each process in the session
What	NumberOfApps	Integer	Conditional		Number of apps invoked during the session time.
What	SessionDuration	Integer	Conditional	Seconds	Duration of the session.

3.9.1.5. Back Office Service Use Case

Back office service provides ASP user with access to applications specific to back-office operations.

3.9.1.5.1. Basic Flow

1. User logs into an Application Service Provider (ASP).
2. After authentication, the user invokes an employee management application.
3. The user checks his benefits info.
4. The user cancels one of the benefits.
5. The user modifies another benefit.
6. The user logs off.

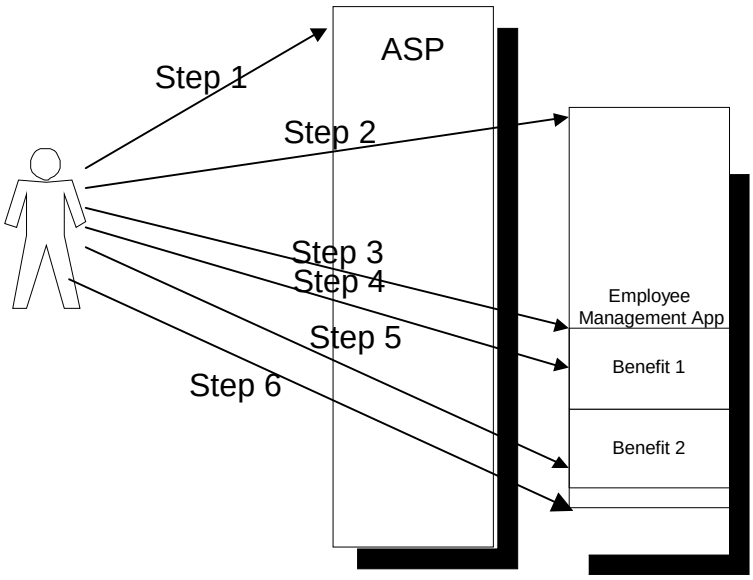

Figure 7. Back office use case: basic flow

3.9.1.5.2. Basic Flow Requirements

1. IPDR must provide unique and clear identification of parties participating in the activity.
2. IPDR must provide information about the type of service used and operations performed.
3. IPDR must provide service consumer activities to including number of requests and amount obtained for each service feature.

3.9.1.5.3. Basic Flow Usage Attribute List

3.9.1.5.4. Service Definitions

Table 4

Category	Usage Attribute Name	Data Type	Presence	Possible Values	Remarks
What	BytesTransferred	Integer	Conditional		Number of bytes transferred on request basis. Should not include any inline images or ad view.
What	NumberOfTransactionsRequested	Integer	Conditional		Number of transactions requested by the user during the session.
What	NumberOfTransactionsCompleted	Integer	Conditional		
What	RequestDuration	Integer	Conditional	Seconds	Time between two consecutive requests
What	VisitTime	Integer	Required	Seconds	The duration that covers a series of consecutive requests to the ASP site, bounded by the first and last requests made by user

3.9.1.6. Online Trading Service Use Case

Online trading service provider provides ASP customer with an opportunity to view the current stock quotes and trade stocks online. It includes access to latest company news, comprehensive account information services, and e-mail assistance from a brokerage firm. It is assumed that the brokerage firm owns the brokerage application.

3.9.1.6.1. Basic Flow

1. User logs into an Application Service Provider (ASP).
2. After authentication user invokes online trading application using his account number.
3. The user checks current stock quotes
4. The user browses latest company news.
5. The user checks his account status.
6. The user attempts to place a stock trading request, encounters a problem with his account.
7. The user sends e-mail to his broker through the application, waits for response.
8. Broker modifies a parameter on user's account.
9. The user receives a response from the broker.
10. The user checks stock quotes again.
11. The user places a stock-trading request, this time successfully.
12. The user logs out.

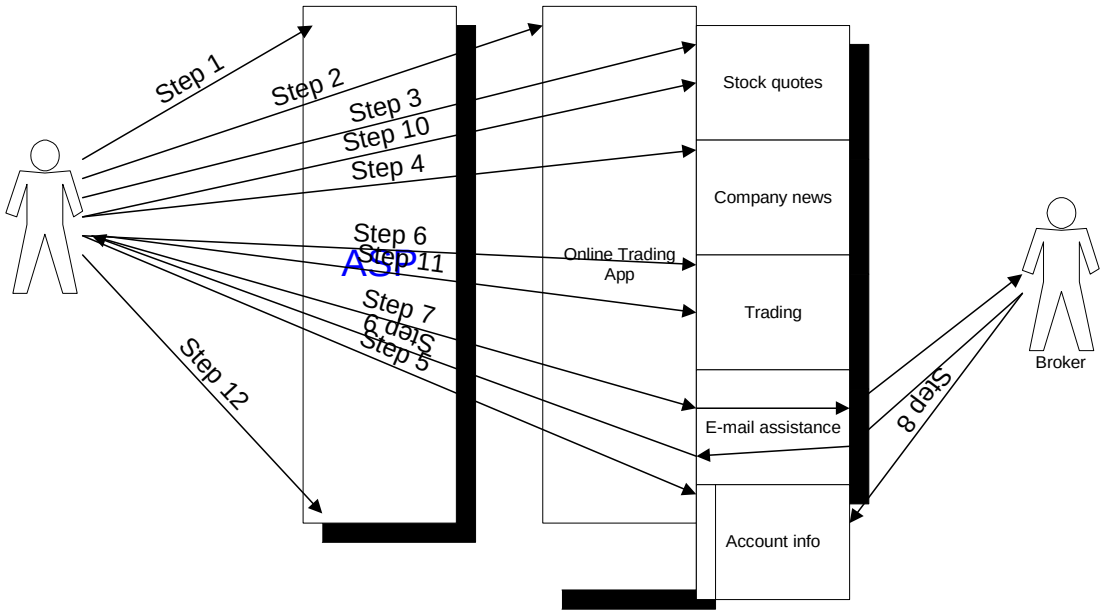

Figure 8 - Online trading use case: basic flow

3.9.1.6.2. Basic Flow Requirements

1. IPDR must provide unique and clear identification of parties participating in the activity, and their locations.
2. IPDR must provide information about the amount and type of resources used.
3. IPDR must provide a list of trading requests executed during the session.

3.9.1.6.3. Basic Flow Usage Attribute List

3.9.1.6.4. Service Consumer

Table 5

Category	**Usage Attribute Name**	**Data Type**	**Presence**	**Possible Values**	**Remarks**
Who	UserAccountNumber	String	Required		May be derived from the user login

3.9.1.6.5. Service Definitions

Table 6

Category	**Usage Attribute Name**	**Data Type**	**Presence**	**Possible Values**	**Remarks**
What	NumberOfEmailAssistanceRequests	Integer	Conditional		Total number of e-mail assistance requests per session
What	NumberOfNewsRequests	Integer	Conditional		Total number of company news requests per session
What	NumberOfQuotes	Integer	Conditional		Total number of stock quotes obtained during the session
What	SessionDuration	Integer	Conditional	Seconds	Duration of the session, including both browsing and trading
What	TradeRequests	String	Conditional	"MSFT, buy, 100", etc.	List of trade requests

3.9.2. Voice over IP (VoIP)

3.9.2.1. Service Definition

VoIP is voice communications between two or more parties over a partial / complete Internet-based connection. The "call" is initiated by a calling party and received by recipient(s). The "call" participants include service elements, gatekeepers, and endpoints (end-users).

The transmission path of the call is realized at VoIP switch by a VoIP gatekeeper (GK) and at each customer location by an Endpoint (VoIP/PSTN). At customer locations, the user speaks and listens into a device that carries the voice data.

The intent is for SEs to generate and transfer to a BSS IPDR records which represent each voice call transparently between all SEs involved in the VoIP call.

3.9.2.2. Service Requirements

1. An IPDR must contain the identifiers of all call participants (call initiator & call recipients).
2. An IPDR must contain the time that the call was initiated and completed.
3. An IPDR must contain call progress codes for each call.
4. The state (as defined in the use case) that describes the phases that each SE goes through in a call must be contained by an IPDR.
5. All times contains in IPDR must be within one 1ms granularity and must have an accuracy of +- 100ms.

3.9.2.3. IP to IP

This use case covers a VoIP scenario where the participating parties use completely Internet-based connections.

3.9.2.3.1. Basic Flow

1. A VoIP caller (IP based EP1) signals a GK1 for a call activation and passes in a callee (IP based call recipient – EP2) id / phone number.
2. The GK1 who owns the call may contact (via dir lookups) another GK to complete the call if the callee is not a subscriber in the GK1's domain.
3. The GK with the callee as a subscriber acknowledges the request for service activation and proceeds to ring/signal the callee.

4. The callee answers the call and enters into the CallConnectedState. The EP's deliver their voice content for a finite amount of time (call duration), and then disconnect the call (call complete).
5. Alternatively, an error may occur, and the call is disconnected. In duration of the call, various QoS changes may occur in real-time

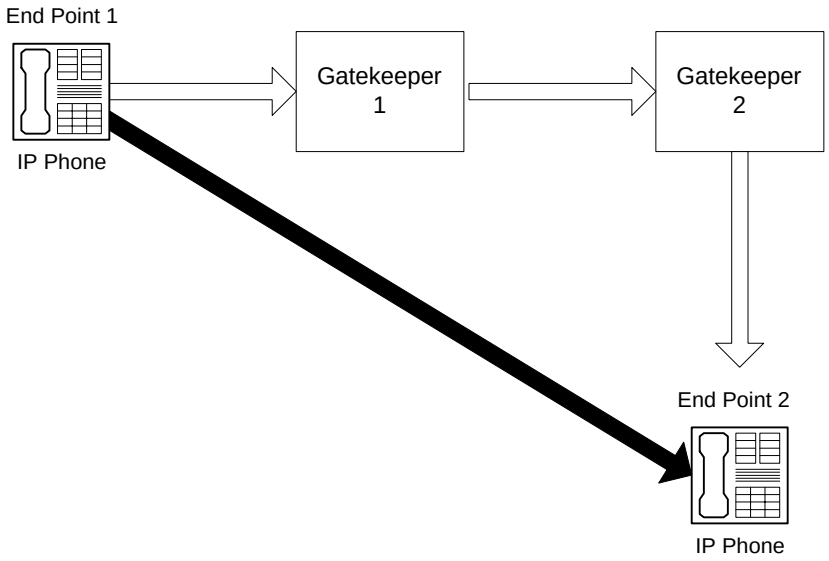

Figure 9 - VoIP Basic Flow

3.9.2.3.2. Basic Flow Requirements

1. Requires at least two EP's for a call to be complete.
2. All EP's must be "On-Net" and all GKs must be "On-Net"
3. An EP needs to be identifiable via either an ID or a phone number.
4. GKs (SP's) must maintain a directory of subscribers, and each subscriber is assigned a unique ID within his domain Each GK must maintain a directory of otherVoIP GKs/GWs.
5. Each GK has a universally unique identifier

3.9.2.3.3. Basic Flow Usage Attribute List

Table 7

Category	Usage Attribute Name	Data Type	Presence	Possible Values	Remarks
Who	Subscriber_ID	String	Required		Unique within a service provider network Tied to a Service Consumer or a Service Element requesting a service
Who	IMSI_ingress	Integer	Conditional		Optional International Mobile Subscriber Identity Optional Required if calling party is using a cellular phone
Who	ESN_ingress	Integer	Conditional		Optional Electronic Serial Number which uniquely identifies each cellular phone Required if calling party is using a cellular phone
Who	Service_Consumer_Type		Optional	EU for end user, NE for network element, NK for partner network	
Who	PIN	String	Conditional		Optional Unique within a service provider network Tied to a Service Consumer or a Service Element requesting a service
When	Start_Access_Time	Integer	Optional		ISO 8601 time when a Service Consumer starts using a Network Element
When	Start_Time	Integer	Required		ISO 8601 time when a Service Consumer starts using a Service Element
When	End_Time	Integer	Required		
What	Call_Duration	Integer	Required	Seconds	This is exclusive of all set-up procedures
What	Total_Time	Integer	Required	Seconds	
What	Type		Conditional	A is for administrative (e.g. authentication and authorization), I is for IVR, N for no answer, V for voice, F for fax, D for data, VF for voice and fax combination, VD for voice and data combination	
What	Feature	String	Conditional	R for roaming, H for home	
What	Codec	String	Optional	G711Alaw, G711Mulaw, G723Low, G723High, G726, G727, G728, G729A, P for proprietary	
What	Modem	String	Optional		Optional Required if a modem is involved
What	Supplementary_Service	String	Optional		This field needs to be extensible to accommodate any number and any type of new service that could be used in conjunction with point-to-point IP telephony (e.g. call waiting, three-way calling, call forwarding, etc.)

Category	Usage Attribute Name	Data Type	Presence	Possible Values	Remarks
What	Disconnect_Reason	String	Required		
What	Extended_Reason_Code	String	Optional		
What	Proprietary_Error_Code	Integer	Optional		Vendor-specific error code
What	Units_Consumed	Integer	Optional		
What	Average_latency	Integer	Required	Milliseconds	Incoming from the IP network. Measured from the preceding node in the call path
What	Inbound_Byte_Count	Integer	Optional		
What	Outbound_Byte_Count	Integer	Optional		
What	Fax_Page_Count	Integer	Conditional		
What	Packet_Loss_Percentage	Integer	Conditional		
What	Out_of_Sequence_Packets	Integer	Optional		
What	Correct_Sequence_Packets	Integer	Optional		
Where	ANI	String	Optional		
Where	ii_Digits	String	Optional		
Where	DNIS	String	Optional		
Where	Destination_Phone_Number	String	Conditional		Digit string entered by the calling party Different from DNIS if two-stage dialing
Where	Outpulsed_Digits	Integer	Optional		Digit string given to the switch on the egress side
Where	IP_Address_Ingress_Device	String	Optional		Required if using a gateway Null if using DHCP.
Where	IP_Address_Egress_Device	String	Optional		Required if using a gateway Null if using DHCP.
Where	Port_Number	String	Optional		
Where	IMSI_egress	String	Conditional		International Mobile Subscriber Identity Required if called party is using a cellular phone
Where	ESN_egress	String	Conditional		Electronic Serial Number which uniquely identifies each cellular phone Required if calling party is using a cellular phone
Where	Home_Location_ID_ingress	String	Conditional	An MSCID or IP address of an HLR	Required if calling party is using a cellular phone
Where	Home_location_ID_egress	String	Conditional	An MSCID or IP address of an HLR	Required if called party is using a cellular phone

3.9.3. E-mail Service

3.9.3.1. Service Definition

E-mail service is a service provided by Internet service provider that includes receiving and sending of messages. It also includes storage of incoming messages.

3.9.3.2. Service Requirements

Not applicable

3.9.3.3. Service Usage Attribute List

Not applicable

3.9.3.4. Basic E-mail

User subscribes to an e-mail service with a service provider. User is advised during the subscription process to configure her/his mail client to delete messages from the server on download. This process will eliminate storing of the e-mail messages on the service provider machines. In the case where download of messages is not practical, an example of which is using portable devices, the service provider would store a limited number of messages or use a limited amount of storage space. When the number of new messages or the amount of storage space exceeds a certain limit three alternative actions can be taken (1) the message will be rejected and a notification will be sent back, (2) the oldest message will be purged and replaced with the new one (FIFO), or (3) a combination of both 1 and 2 where, if read messages exist they are purged and replaced other wise the new message is rejected.

3.9.3.4.1. Basic Flow

1. Message arrives at a service provider's mail server.
2. After determining the recipient of the message, the mail server will check the number of messages, or storage space, in the respective inbox to determine availability.
3. If the number of messages, or storage space, is below a certain limit, the message is deposited into the inbox.
4. Otherwise, the mail server will reject the message sending a notification to the sender and keeping a note to the owner.
5. The owner logs into the system using her/his mail client. Client is configured to delete messages on download.
6. As messages are downloaded or read, they may be deleted from the server.

3.9.3.4.2. Basic Flow Requirements

These are the basic flow requirements for the e-mail use case:

1. Mediation shall support recording IPDRs that are not service consumer initiated.

3.9.3.4.3. Basic Flow Usage Attribute List

3.9.3.4.3.1. Service Consumer

Table 8

Category	Usage Attribute Name	Data Type	Presence	Possible Values	Remarks
Who	UserLoginName	String	Required		This should identify a unique user in the system. Real time mapping of dynamically allocated IP addresses might be necessary.
Where	UserLoginLocation	String	Required		This could also be included under the where section. However, it is noted here to recognize the difference between the location of the service provider and the service consumer.

3.9.3.4.3.2. Service Provider

Table 9

Category	Usage Attribute Name	Data Type	Presence	Possible Values	Remarks
Where	ProviderName	String	Required		This is the actual provider of the service.
Where	ProviderLocation	String	Required		This is the location of the service provider.

3.9.3.4.3.3. Service Definitions

Table 10

Category	Usage Attribute Name	Data Type	Presence	Possible Values	Remarks
What	Service	String	Required		This use case supports only one service type, e-mail, however, a generic type field is need to cover multiple services offered by the same provider.
What	Feature	String	Required		Again, this is included to keep the general notion of having multiple features for each service type. An example could be monitoring e-mail based on messages as a different feature than monitoring based on storage space.
What	ProcessingTime	Integer	Conditional		This is the total time used by the server to process an e-mail.
What	Storage	Integer	Conditional		Is based on the total storage at the time of generating a usage record.
What	StorageDuration	Date time	Conditional		The duration of time messages were stored on the providers servers.
What	BytesTransferred	Integer	Conditional		This is the total bytes transferred either during e-mail arrival or during e-mail download.
What	EventTime	Integer	Conditional		This is the time at which the event took place. Events are generated when a message arrives and when a message is read.

4. Protocol

4.1. Introduction

The complete discussion of protocol, namely notation, syntax, encoding, message format, procedure, and semantics, is the subject of this section.

4.2. Notation

The initial notation for the IPDR is based on the eXtensible Markup Language specification, XML 1.0.

4.3. Structure Diagrams

The following diagrams illustrate an IPDR documents structure. The XML DTD for the IPDR document (IPDRDoc) element defines this hierarchy of elements.
The hierarchy allows an IPDRDoc to contain many usage records (IPDRs). A usage record is further divided into groupings defining a Service Session (SS) between a Service Consumer (SC) and a Service Element (SE) and a record of the Usage Event (UE) itself. Details about the consumer, service element and usage event are contained in usage attribute value (v) elements. These elements are identified by well-defined names. Section 5 contains a set of tables with names defined by IPDR.org.

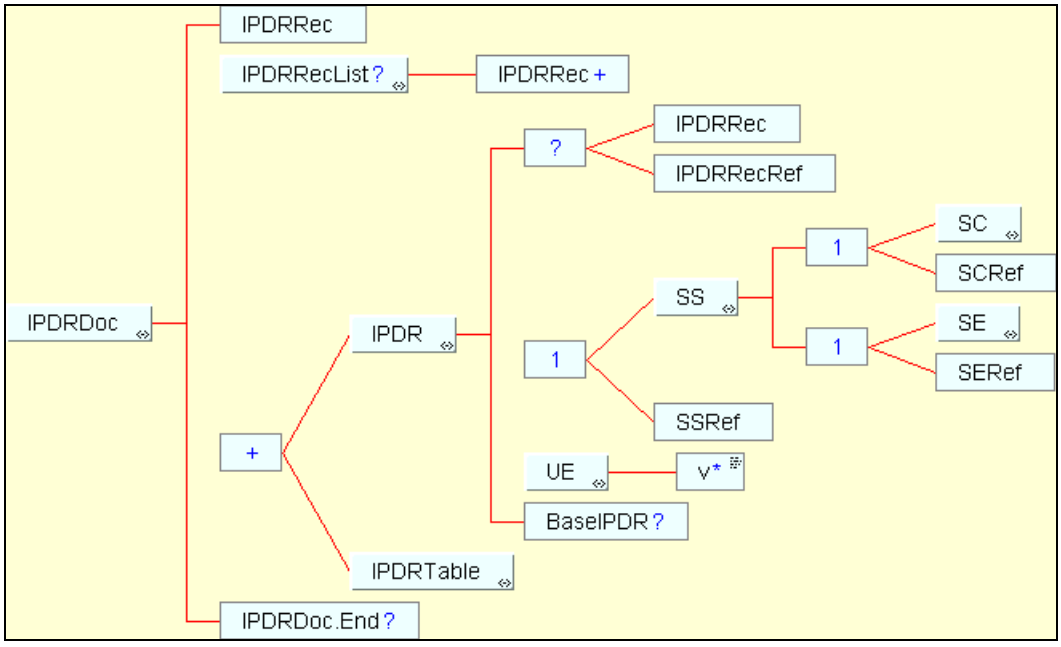

Figure 10- IPDRDoc (with IPDR expanded)

Figure 10 shows the elements under the top-level element, the IPDR document (IPDRDoc). The elements directly below consist of an identifier for the entity recording the usage (IPDRRec) followed by an optional list of additional recording entities (IPDRRecList).

The document's main body is made up of one or more IPDRs that represent a single usage event.

IPDRs may appear immediately within the body of an IPDRDoc or may be grouped under an IPDRTable.

The document has an optional ending block of information represented by IPDRDoc.End.

Figure 10expands the IPDR element to show that it begins with an optional IPDRRec or a reference to another IPDRRec in the document. An IPDR then contains a pairing of Service Consumer (SC) and Usage Element (UE) under the Service Session (SS).

The Service Session (SS) element is merely a structural convenience. It allows repeated pairs of the same SC and SE to be associated via reference. The usage represented may be measured as a discrete event, or part of an ongoing session.

The Usage Event (UE) element contains data describing metrics or parameters of a given Usage Event. The (v) element represents a Usage Attribute Value.

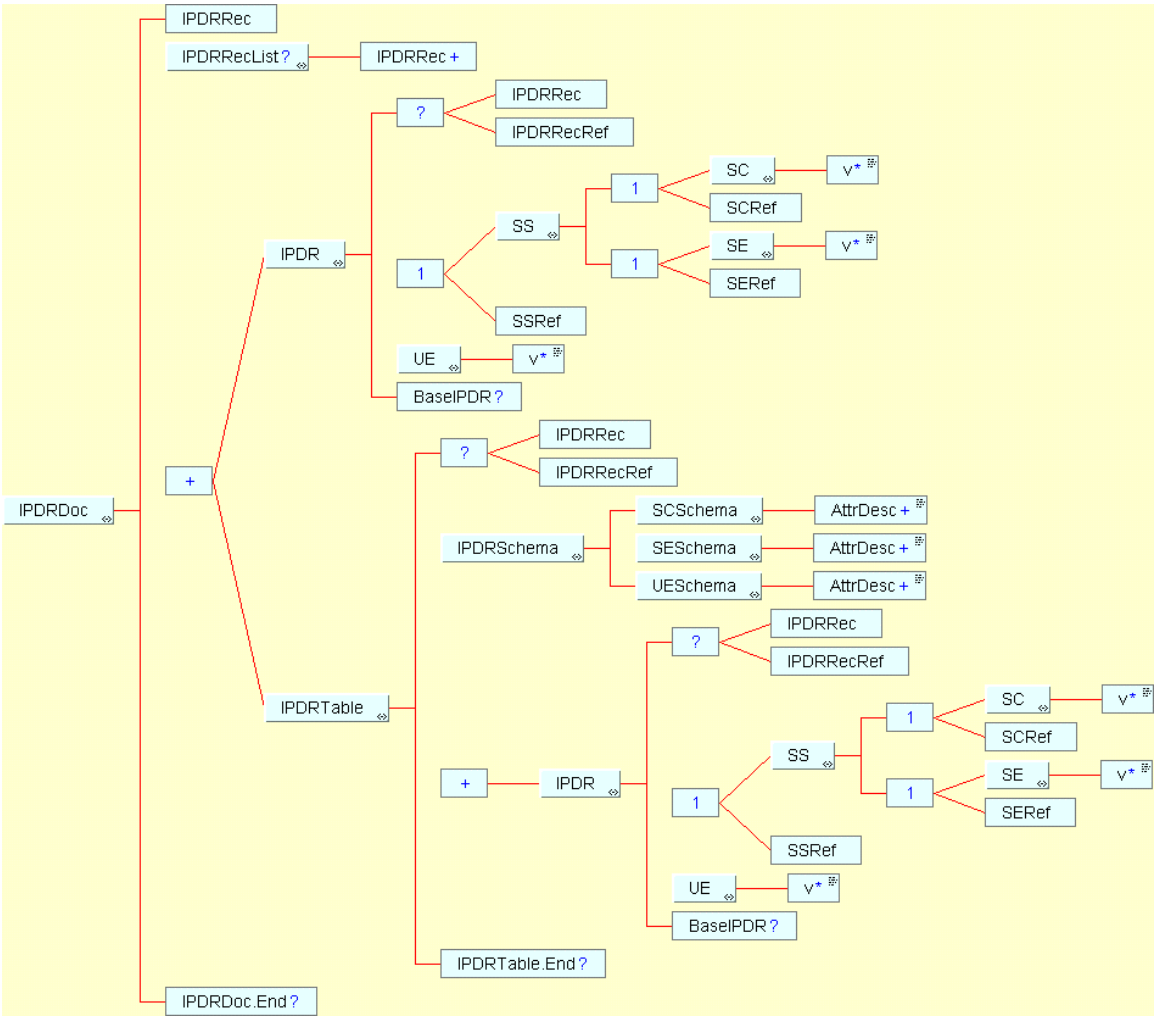

Figure 11 - IPDRDoc (with IPDRTable element expanded)

Figure 11 expands on the IPDRTable construct. The IPDRTable element is used to group a homogenous set of IPDRs.

Grouping allows for more compact representation of the IPDRs by only requiring that Usage Attribute Value names be made explicit once.

The three schema elements provide a description of the usage attribute values that will appear in the contained set of IPDRs. It requires that all IPDRs in the set have common sets of usage attribute values in a consistent order.

The optional IPDRTable.End element may be used for additional ending information (like a check count).

4.4. DTD

The DTD for IPDR documents is presented first, because it formally describes how IPDR documents are constructed. Comments are interspersed throughout the DTD, but they do not form part of the specification. This document contains additional details on how the elements of the DTD may be used. These additional details are considered part of the overall IPDR Document specification.

The most recent version of the DTD for the IPDR Document format is available at:

http://www.ipdr.org/public/ipdr_1.0.dtd.

The DTD in its entirety is presented below. This section is followed by the Annotated DTD section that contains further description and restrictions on the various elements. The annotations are considered part of the specification.

```
<!--  The IPDRDoc element is the top-level container of a set of
      IPDRs.  The document will also define the entity which
      recorded these IPDRs via the IPDRRec element.
  -->
<!ELEMENT IPDRDoc  (IPDRRec , IPDRRecList? ,  (IPDR | IPDRTable )+ ,
IPDRDoc.End? )>
<!ATTLIST IPDRDoc  seqNum    CDATA     #IMPLIED
                   version   CDATA     #IMPLIED
                   startTime CDATA     #IMPLIED
                   info      CDATA     #IMPLIED
                   a-dtype   NMTOKENS  'seqNum    int
                                        startTime dateTime.tz' >
<!--   The IPDRDoc.End element optionally marks the end of the IPDR
block.
      It may contain some check information like a count of IPDRs.
  -->
<!ELEMENT IPDRDoc.End EMPTY>
<!ATTLIST IPDRDoc.End  count   CDATA     #IMPLIED
                       endTime CDATA     #IMPLIED
                       a-dtype NMTOKENS  'count   int
                                          endTime dateTime.tz' >
<!-- The IPDRRec element describes the entity that is responsible for
     creating (recording) the IPDRDoc.
  -->
<!ELEMENT IPDRRec EMPTY>
<!ATTLIST IPDRRec  id        ID        #IMPLIED
                   startTime CDATA     #IMPLIED
                   info      CDATA     #IMPLIED
                   a-dtype   NMTOKENS  'startTime dateTime.tz' >
<!-- The IPDRRecRef element may be used to associate common references
     to the same IPDRRec element without repeating its other usage
attributes.
  -->
<!ELEMENT IPDRRecRef EMPTY>
<!ATTLIST IPDRRecRef  ref IDREF  #REQUIRED >
<!-- The IPDRRecList identifies contributing IPDR recording entities
```

```
     which were used in the construction of the current IPDR Document.
     A typical example use would be for an aggregator of IPDR documents
     to identify the set of initial recorders presenting IPDRs
  -->
<!ELEMENT IPDRRecList  (IPDRRec+ )>

<!-- An IPDR describes an event between a Service Consumer (SC) and
     a Service Element (SE).  The SC and SE elements are contained
     beneath an entity called the Service Session (SS).  Details of
     the event is contained in the Usage Event (UE) element.  All IPDRs
     have a time indicating when the event occurred.
  -->
<!ELEMENT IPDR   ( (IPDRRec | IPDRRecRef )? ,   (SS | SSRef ) , UE ,
BaseIPDR? )>
<!ATTLIST IPDR  id      ID        #IMPLIED
                time    CDATA     #REQUIRED
                seqNum  CDATA     #IMPLIED
                a-dtype NMTOKENS  'time   dateTime.tz
                                   seqNum int' >
<!-- The Service Session (SS) element groups the Service Consumer
     and Service Element information.  This grouping allows
     an SC/SE pair to be associated with other IPDRs via
     a single reference (the SSRef).
  -->
<!ELEMENT SS  ( (SC | SCRef ) ,  (SE | SERef ) )>
<!ATTLIST SS  id      ID     #IMPLIED
              service CDATA  #IMPLIED >
<!-- The SSRef element may be used to associate common references
     to the same pairing of a Service Consumer and a Service Element.
  -->
<!ELEMENT SSRef EMPTY>
<!ATTLIST SSRef  ref IDREF  #REQUIRED >
<!-- An IPDRs Service Consumer, Service Element and Usage Event
     sections  are  used  to  partition  a  set  of  usage  attributes  <v>
elements
     into their appropriate categories.  The SC and SE components
     also, have Reference analogs to associate common groups of
     usage attributes, and reduce the number of elements in a document.

     All Service events indicate what type of metrics they carry.
     By default a 'Start-Stop' type indicates that it is a
     complete measurement and does not rely on other
     records to form a complete picture of an activity.
  -->
<!ELEMENT SC  (v* )>
<!ATTLIST SC  id ID  #IMPLIED >
<!ELEMENT SE  (v* )>
<!ATTLIST SE  id ID  #IMPLIED >
<!ELEMENT UE  (v* )>
<!ATTLIST UE   type   (Start | Stop | Start-Stop | Interim )   'Start-
Stop'
              name CDATA  #IMPLIED >
<!--  The  SERef  and  SCRef  elements  may  be  used  to  associate  common
references to the Session Element or Session Consumer.
  -->
<!ELEMENT SERef EMPTY>
<!ATTLIST SERef  ref IDREF  #REQUIRED >
```

```
<!ELEMENT SCRef EMPTY>
<!ATTLIST SCRef  ref IDREF  #REQUIRED >
<!-- The BaseIPDR element allows reference to be made to IPDRs which
     contributed to the construction of the current IPDR element.
  -->
<!ELEMENT BaseIPDR EMPTY>
<!ATTLIST BaseIPDR  refs IDREFS  #REQUIRED >
<!-- The usage attribute value <v> element is the main extensibility
mechanism
     of the IPDR record.  The IPDR group will define a set of named
     usage attributes that follow a hierarchical naming convention.
     All IPDR derived usage attribute names will reside under the
     org.ipdr.* hierarchy.  Based on their position in an element,
     the leading org.ipdr name of the usage attribute may be omitted
     for compactness.  Third party usage attributes must be fully
     qualified.
  -->
<!ELEMENT v  (#PCDATA )>
<!ATTLIST v  name CDATA  #IMPLIED >
<!-- An IPDRTable element is used to more compactly represent a set
     of homogenous IPDRs.  Homogenous in the sense that each IPDR
     in a table contains the same set of describing usage attribute
     elements <v> elements.  Use of IPDRTable is optional.  IPDR
     elements may directly live under the IPDRDoc if they
     do not share a common structure or verbosity is desired
     over compactness.
  -->
<!ELEMENT IPDRTable  ( (IPDRRec | IPDRRecRef )? , IPDRSchema ,  (IPDR
)+ , IPDRTable.End? )>
<!ATTLIST IPDRTable  startTime CDATA     #IMPLIED
                     a-dtype   NMTOKENS  'startTime dateTime.tz' >
<!--  The IPDRTable.End element optionally marks the end of a set of
IPDRs
      in an IPDRTable element.  It may contain some check information
like a count of IPDRs.
  -->
<!ELEMENT IPDRTable.End EMPTY>
<!ATTLIST IPDRTable.End  count   CDATA     #REQUIRED
                         endTime CDATA     #IMPLIED
                         a-dtype NMTOKENS  'count   int
                                            endTime dateTime.tz' >
<!-- The IPDRSchema element within an IPDRTable defines the usage
attribute
     elements common to all the IPDRs in the given table
  -->
<!ELEMENT IPDRSchema  (SCSchema , SESchema , UESchema )>

<!-- The following schema components define the sets of usage
attributes
     for the Service Consumer (SC), Service Element (SE) and
     Usage Event (UE) components of the IPDR respectively.
  -->
<!ELEMENT SCSchema  (AttrDesc+ )>

<!ELEMENT SESchema  (AttrDesc+ )>

<!ELEMENT UESchema  (AttrDesc+ )>
```

```
<!-- An AttrDesc element describes a usage attribute that will appear
in
     in a particular position in a subsequent set of IPDRs contained
     in a table.  One can think of this as being analogous to the
     table header <TH> elements in HTML.
  -->
<!ELEMENT AttrDesc  (#PCDATA )>
```

4.5. Annotated DTD

A description of each element in the IPDR DTD is presented below.

IPDRDoc

```
<!--  The IPDRDoc element is the top-level container of a set of
      IPDRs.     The   document   will   also   define   the   entity   that
recorded these IPDRs via the IPDRRecorder element.
  -->
<!ELEMENT IPDRDoc   (IPDRRec, IPDRRecList? ,   (IPDR | IPDRTable )+ ,
IPDRDoc.End? )>
<!ATTLIST IPDRDoc  seqNum    CDATA     #IMPLIED
                   version   CDATA     #IMPLIED
                   startTime CDATA     #IMPLIED
                   info      CDATA     #IMPLIED
                   a-dtype   NMTOKENS  'seqNum    int
                                        startTime dateTime.tz' >
```

The attributes of the IPDRDoc element are described below.

- seqNum - an optional integer value for auditing sets of IPDRs. ***[Ed. note: additional work on numbering policy is required. Subsequent revisions may provide more guidelines on how sequencing is done. It is also possible the type may change (e.g. to basic strings)]***
- version - identifies the version of the IPDRDoc DTD being used. This version shall be '1.0'.
- startTime - indicates the time the recorder began producing this document. (See the "Additional Element Details" subsection for more information about timestamps")
- info - optional describing string for this document.

IPDRDoc.End

```
<!--   The IPDRDoc.End element optionally marks the end of the IPDR
block.
      It may contain some check information like a count of IPDRs.
  -->
<!ELEMENT IPDRDoc.End EMPTY>
<!ATTLIST IPDRDoc.End  count   CDATA     #IMPLIED
                       endTime CDATA     #IMPLIED
                       a-dtype NMTOKENS  'count   int
                    endTime dateTime.tz' >
```

The attributes of the IPDRDoc.End element are described below.

- count - the number of IPDRs contained in this document (used as a check). This includes IPDRs contained in tables.
- endTime - the time the IPDR recorder finished creating this document. (See the "Additional Element Details" subsection for more information about timestamps")

IPDRRec

```
<!-- The IPDRRec element describes the entity that is responsible for
creating (recording) the IPDRDocument.
  -->
<!ELEMENT IPDRRec EMPTY>
<!ATTLIST IPDRRec  id        ID        #IMPLIED
                   startTime CDATA     #IMPLIED
                   info      CDATA     #IMPLIED
                                       a-dtype   NMTOKENS  'startTime dateTime.tz' >
```

The attributes of the IPDRRec element are described below.

- id - a unique identifier within this document. It allows subsequent IPDRs to repeat a common IPDR recorder by reference.
- startTime - indicates the time this IPDR recording entity began running. The same entity may have produced multiple documents during its run. (See the "Additional Element Details" subsection for more information about timestamps")
- info - optional describing string for this document.

IPDRRecRef

```
<!-- The IPDRRecRef element may be used to associate common references
     to the same IPDRRec element without repeating its other usage
attributes.
  -->
<!ELEMENT IPDRRecRef EMPTY>
<!ATTLIST IPDRRecRef  ref IDREF  #REQUIRED >
```

The attributes of the IPDRRecRef element are described below.

- ref - a reference to an IPDRRec element defined earlier in the document. Note that implicitly all IPDRs were recorded by the IPDR recorder described at the top of the IPDRDoc. Individual IPDRs or tables may optionally identify a different recorder.

IPDRRecList

```
<!-- The IPDRRecList identifies contributing IPDR recording entities
     which were used in the construction of the current IPDR Document.
     A typical example use would be for an aggregator of IPDR documents
     to identify the set of initial recorders presenting IPDRs
  -->
<!ELEMENT IPDRRecList  (IPDRRec+ )>
```

IPDR

```
<!-- An IPDR describes an event between a Service Consumer (SC) and
     a Service Element (SE).  The SC and SE elements are contained
     beneath an element called the ServiceSession (SS).  Details of
     the event is contained in the Usage Event (UE) element.  All IPDRs
     have a time indicating when the event occurred.
  -->
<!ELEMENT  IPDR   (  (IPDRRec  |  IPDRRecRef)?  ,   (SS  |  SSRef)  ,  UE,
BaseIPDR? )>
<!ATTLIST IPDR  id      ID        #IMPLIED
                time    CDATA     #REQUIRED
                seqNum  CDATA     #IMPLIED
                a-dtype NMTOKENS  'time   dateTime.tz
                                   seqNum int'>
```

The attributes of the IPDR element are described below.

- id - a unique identifier for this IPDR in order to support the BaseIPDR element.
- time - the time the recorded usage event occurred. (See the "Additional Element Details" subsection for more information about timestamps")
- seqNum - an optional integer value for auditing sets of IPDRs. *(ed note: additional work on numbering policy req'd)*

SS

```
<!-- The Service Session (SS) element groups the Service Consumer
     and Service Element information.  This grouping allows
     an SC/SE pair to be associated with other IPDRs via
     a single reference (the SSRef).
  -->
<!ELEMENT SS  ( (SC | SCRef ) ,  (SE | SERef ) )>
<!ATTLIST SS  id              ID     #IMPLIED
              service         CDATA  #IMPLIED>
```

The attributes of the SS element are described below.

- id - a unique identifier within this document. It allows subsequent IPDRs to repeat a common Service Session by reference.
- service – a name indicating the service being provided for this session. This may aid in classification of IPDR records in a document (e.g. HTTP vs. IMAP services).

SSRef

```
<!-- The SSRef element may be used to associate common references
to the same pairing of a Service Consumer and a Service Element.
  -->
<!ELEMENT SSRef EMPTY>
<!ATTLIST SSRef  ref IDREF  #REQUIRED >
```

The attributes of the SSRef element are described below.

- ref - a reference to an SS element that contains the same set of Consumer and Element identifiers.

SC, SE, UE

```
<!-- An IPDRs ServiceConsumer, ServiceElement and UsageEvent
     sections are used to partition a set of usage attributes <v>
elements
     into their appropriate categories.  The SC and SE components
     also, have Reference analogs to associate common groups of usage
     attributes, and reduce the number of elements in a document.

     All Service events indicate what type of metrics they carry.
     By default a 'Start-Stop' type indicates that it is a
     complete measurement and does not rely on other records to form a
complete picture of an activity.
  -->
<!ELEMENT SC  (v* )>
<!ATTLIST SC  id ID  #IMPLIED >
<!ELEMENT SE  (v* )>
<!ATTLIST SE  id ID  #IMPLIED >
<!ELEMENT UE  (v* )>
<!ATTLIST UE  type  (Start | Stop | Start-Stop | Interim )
                                                  "Start-Stop"
              name CDATA #implied>
```

The attributes of the SC and SE element are described below.
id - a unique identifier within this document. It allows subsequent IPDRs to repeat a common Service Consumer or Element by reference.

The UE element has the following attributes:

- type – indicates the type of event recorded. These may be Start, Stop or Interim events, or a Start-Stop. Start and Stop indicate the beginning or end of a service being delivered. Interim events provide updated metrics on an ongoing activity. Start-Stop events describe an entire service delivery.
- name – provides additional qualification of the type field.

SCRef, SERef

```
<!-- The SERef and SCRef elements may be used to associate common
     references to the Session Element or Session Consumer.
  -->
<!ELEMENT SERef EMPTY>
<!ATTLIST SERef  ref IDREF  #REQUIRED >
<!ELEMENT SCRef EMPTY>
<!ATTLIST SCRef  ref IDREF  #REQUIRED >
```

The attributes of the SCRef and SERef element are described below.

- ref - a reference to an SC or SE element that contains the same set of identifiers.

BaseIPDR

```
<!-- The BaseIPDR element allows reference to be made to IPDRs that
contributed to the construction of the current IPDR element.
  -->
<!ELEMENT BaseIPDR EMPTY>
<!ATTLIST BaseIPDR  refs IDREFS  #REQUIRED >
```

The attributes of the BaseIPDR element are described below.

- refs - a list of references to IPDRs that contributed to the formation of this IPDR. ***[Ed note: additional work on how references are constructed is required.]***

Usage Attribute Value <v>

```
<!-- The usage attribute value <v> element is the main extensibility
mechanism
     of the IPDR record.  The IPDR group will define a set of named
usage attributes that follow a hierarchical naming convention.
     All IPDR derived usage attribute names will reside under the
     org.ipdr.* hierarchy.  Based on their position in an element,
     the leading org.ipdr name of the usage attribute may be omitted
     for compactness.   Third party usage attributes must be fully
qualified.
  -->
<!ELEMENT v (#PCDATA )>
<!ATTLIST v name CDATA  #IMPLIED >
```

The attributes of the Usage Attribute Value <v> element are described below.

- name - gives the name of this identifier, metric or parameter. Names are taken from the namespace described in the Usage Attribute Values section.

IPDRTable

```
<!-- An IPDRTable element is used to more compactly represent a set
```

```
     of homogenous IPDRs.  Homogenous in the sense that each IPDR
     in a table contains the same set of describing usage attribute
     elements <v> elements.  Use of IPDRTable is optional.  IPDR
     elements may directly live under the IPDRDoc if they
     do not share a common structure or verbosity is desired
     over compactness.
  -->
<!ELEMENT IPDRTable  ( (IPDRRec | IPDRRecRef)?, IPDRSchema,  (IPDR )+,
IPDRTable.End?)>
<!ATTLIST IPDRTable  startTime CDATA     #IMPLIED
a-dtype  NMTOKENS  'startTime dateTime.tz' >
```

See additional discussion of the IPDRTable element
Further discussion of the IPDRTable construct is present in the "Additional Element Details" subsection. Also, see the discussion on "Elements vs. Usage Attributes".

The attributes of the IPDRTable element are described below.

- startTime - the time the IPDR recorder began creating this table of IPDRs. (See the "Additional Element Details" subsection for more information about timestamps")

IPDRTable.End

```
<!--  The IPDRTable.End element optionally marks the end of a set of
IPDRs
     in an IPDRTable element.  It may contain some check information
like a count of IPDRs.
  -->
<!ELEMENT IPDRTable.End EMPTY>
<!ATTLIST IPDRTable.End  count   CDATA     #REQUIRED
                         endTime CDATA     #IMPLIED
                         a-dtype NMTOKENS  'count   int
                                           endTime dateTime.tz' >
```

The attributes of the IPDRTable element are described below.

- count - the number of IPDRs contained in this table (used as a check).
- endTime - the time the IPDR recorder finished creating this table of IPDRs. (See the "Additional Element Details" subsection for more information about timestamps")

IPDRSchema

```
<!-- The IPDRSchema element within an IPDRTable defines the usage
attribute
     elements common to all the IPDRs in the given table
  -->
<!ELEMENT IPDRSchema ( SCSchema, SESchema, UESchema)>
```

SCSchema, SESchema, UESchema

```
<!--  The following schema components define the sets of usage
attributes
     for the ServiceConsumer (SC), ServiceElement (SE) and
     Usage Event (UE) components of the IPDR respectively.
  -->
```

```
<!ELEMENT SCSchema (AttrDesc+)>
<!ELEMENT SESchema (AttrDesc+)>
<!ELEMENT UESchema (AttrDesc+)>
```

AttrDesc

```
<!-- An AttrDesc element describes a usage attribute that will appear
in
     in a particular position in a subsequent set of IPDRs contained
     in a table.  One can think of this as being analogous to the
     table header <TH> elements in HTML.
  -->
<!ELEMENT AttrDesc (#PCDATA)>
```

4.6. Additional Element Details

4.6.1. Element Naming

Some element names have been shortened. Human readability is a nice feature of ASCII XML, but for the most part these documents will be processed by applications. Transfer, storage and parsing costs should not be entirely ignored when developing the DTD. The most frequently used elements gain the most from shortened names.

4.6.2. Timestamps

Human readability is a desirable property of an XML document. When humans are intended or even somewhat likely consumers of an IPDR document, the format are described (along with many more) in ISO 8601.
Date representations which are human readable are not a particularly efficient mechanism for some applications. Using the direct integer value of UTC time can be a simpler format and parse exercise . Both the 32-bit (UNIX, NT) and 64-bit (Java) representations should be allowed. To aid parsing distinction from the human format, the UTC time representations will be written beginning the letter 'U'. Whether the value is a 32-bit or 64-bit quantity is determined by the second character position. For 64-bits the second position is the letter 'L', for 32-bit it will be a digit. Note that 32-bit values record seconds since Epoch, while the 64-bit records milliseconds since Epoch. Epoch is defined as Jan 1, 1970 00:00 GMT.

Examples

1999-09-22T11:20:01-08:00
1999-09-22T19:20:01Z
U938292388
UL938292388011

4.6.3. IPDRTable Construct

Applications may often produce IPDR documents consisting of homogenous sets of IPDRs. If all IPDRs have the same structure, some efficiency can be gained by laying out the structure of the IPDRs once in an IPDRSchema, and then interpreting all IPDRs contents according to the defined schema.
The IPDRTable is similarly structured to the familiar HTML <Table> entity. When defining an HTML table, the columns are defined with <TH> entries. The table itself is populated with values organized into rows, using the <TR> and <TD> entries.

For an IPDRTable, there are logically three sets of tables, one for the ServiceConsumer, ServiceElement and UsageEvent entries.

4.6.4. Usage Attribute Value Element (<v>)

The usage attribute value element is the most common element used within the IPDR structure. It is also the point of greatest extension. The element is extended by the definition of new names.

The ipdr.org group will manage a set of standardized names. Ideally, the majority of IPDR documents produced will exclusively use names from this common namespace.

However, it is expected that some vendors will also want to create their own names and may not need or desire the approval of an outside party (such as ipdr.org). To address this a namespace should be used which can be administered in a hierarchical (and preferably already established) manner. The DNS naming hierarchy has these properties and is reasonably compact and readable.

The proposed naming mechanism would use a simple dotted notation, for example *org.ipdr..ipAddress* ***or*** *com.foo.ipdr.specialAttr.*

Since names from the org.ipdr.* namespace will be used most frequently, they may alternatively be written omitting the "org.ipdr." prefix which will be implied by context.

Section 5 of this document defines a set of usage attribute value elements for use with IPDR v1.0.

4.6.5. XML Element vs. XML Attribute

An alternative to the use of structures that create an element for each piece of data related to a usage event, the data could also be brought directly into the containing element as XML attributes. For example:

```
<IPDR>
  <SS>
    <SC><v name="ipAddress">12.12.3.1</v></SC>
    <SE><v name="hostName">www.foo.com</v></SE>
  </SS>
  <UE><v name="objSize">12333</v>
      <v name="status">200</v>
      <v name="URL">http://xyz.com/banner.gif</v>
  </UE>
</IPDR>
```

Could also be represented as follows:

```
<IPDR>
  <SS>
    <SC ipAddress="12.12.3.1"/>
    <SE hostName="www.foo.com"/>
  </SS>
  <UE URL="http://xyz.com/banner.gif" objSize="12333" status="200"/>
</IPDR>
```

The latter form is more compact than the explicit IPDR form above, but would not benefit from the IPDRTable construct. A benefit of the latter form is that it is subject to stronger type checking by automatic parsers. This same benefit however, makes extensibility of the XML syntax a bit more difficult. For the result to be considered a well-formed expression, all attributes that appear in an element must have been specified in a DTD.

Ongoing work in the W3C organization related to XML schemas and namespaces may make the attribute based approach a better-suited longer-term alternative. These mechanisms were not used in v1.0 of IPDR because they were still in draft form.

If future versions of IPDR move in this direction, the tables currently presented in Section 5, would likely be formal XML documents themselves.

4.7. *Examples*

4.7.1. Example 1. Video on Demand

This example shows a minimal IPDR document containing a record describing a summary of a Video on Demand session.

```
<?xml version="1.0"?>
<!DOCTYPE IPDRDoc SYSTEM "ipdrdoc.dtd">
<IPDRDoc seqNum="124" version="1.0" >
  <IPDRRec info="aggregator.ipdr.org"/>
  <IPDR seqNum="1" time="2000-02-01T07:00:00Z">
    <SS id="ses10" service="RTSP">
      <SC>
         <v name="subscriberId">Joe Blow</v>
         <v name="ipAddress">192.168.1.10</v>
      </SC>
      <SE>
         <v name="hostName">rtsp.ipdr.org</v>
      </SE>
    </SS>
    <UE>
      <v name="movieName">Rocky CIX</v>
      <v name="startTime">1999-12-31T23:59:00Z</v>
      <v name="endTime">2000-01-01T04:15:00Z</v>
      <v name="numAudioStreams">1</v>
      <v name="numVideoStreams">1</v>
       <v name="terminationStatus">Normal</v>
    </UE>
  </IPDR>
</IPDRDoc>
```

4.7.2. Example 2. HTTP

This example shows an IPDR document containing a descriptions of several HTTP accesses against a web server. It also demonstrates the use of the IPDRTable mechanism.

```
<?xml version ="1.0"?>
<!DOCTYPE IPDRDoc SYSTEM "ipdrdoc.dtd">
<IPDRDoc version="1.0" startTime="1999-09-21T21:33:17Z">
   <IPDRRec info="foo.xyz.com"/>
   <IPDRTable>
      <IPDRSchema>
         <SCSchema><AttrDesc>ipAddress</AttrDesc></SCSchema>
         <SESchema><AttrDesc>hostName</AttrDesc>
         </SESchema>
         <UESchema><AttrDesc>URL</AttrDesc>
                   <AttrDesc>objSize</AttrDesc>
                   <AttrDesc>status</AttrDesc>
         </UESchema>
      </IPDRSchema>

      <IPDR time="1999-09-21T21:33:17Z">
         <SS>
           <SC><v>192.2.1.1</v></SC>
           <SE><v>www12.foo.com</v</SE>
```

```
         </SS>
         <UE><v>http://www.foo.com/index.html</v>
             <v>8712</v>
             <v>200</v>
         </UE>
      </IPDR>
      <IPDR time="1999-09-21T21:33:19Z">
         <SS>
           <SC><v>17.123.1.29</v></SC>
           <SE><v>www12.foo.com</v></SE>
         </SS>
         <UE><v>http://www.foo.com/banner.gif</v>
             <v>0</v>
             <v>404</v>
         </UE>
      </IPDR>
      <IPDR time="1999-09-21T21:33:19Z">
         <SS>
           <SC><v>17.123.1.29</v></SC>
           <SE><v>www12.foo.com</v><v>http</v></SE>
         </SS>
         <UE><v>http://www.foo.com/ad.gif</v>
             <v>11344</v>
             <v>200</v>
         </UE>
      </IPDR>
   </IPDRTable>
</IPDRDoc>
```

5. Structures

This chapter presently contains only the protocol-independent document attributes related to the Protocol chapter. When the notation system is upgraded to support XML Schema (pending formal approval by W3C), IPDRs will be specified here corresponding to the usage attributes specified in the Business Requirements chapter use cases.

5.1. Protocol- independent document attributes

5.1.1. Service metrics

Table 11

Number	**Document Attribute Name**	**Data Type**	**Possible Values**	**Notes**
1.	sessionId	String ?		Unique within a service provider network
2.	subscriberId	String ?		Unique within a service provider network Can be tied to a Service Consumer or a Service Element
3.	serviceProviderId	String ?		Should be globally unique and assigned by a central authority
4.	ipAddress	ipAddress		
5.	port	Integer		
6.	startTimeISO 8601	Datetime		ISO 8601 time when a Service Consumer starts using a Service Element
7.	startTime	Datetime		Local time for the Service Element when a Service Consumer starts using it
8.	endTimeISO 8601	Datetime		ISO 8601 time when a Service Consumer end using a Service Element
9.	endTime	Datetime		Local time for the Service Element when a Service Consumer end using it
10.	duration	Integer	Seconds	service duration excluding set-up procedure
11.	totalTime	Integer	Seconds	service duration including set-up procedure
12.	setupDuration	Integer	Milliseconds	set-up procedure duration
13.	gmtOffset	int	Minutes	Offset from GMT necessary to get local time.

5.1.2. Traffic metrics

Table 12

Number	**Document Attribute Name**	**Data Type**	**Possible Values**	**Notes**
14.	sendOctets	Integer		The total number of octets send by Service Consumer
15.	recvOctets	Integer		The total number of octets received by Service Consumer
16.	sendPkt	Integer		The total number of packets send by Service Consumer
17.	recvPkt	Integer		The total number of packets received by Service Consumer

6. Appendix - Use Case Template

There is no need for an introduction to this section since it is part of a bigger document where introduction is included.

Service Definition

Include in this section a brief description of the service covered in this use case. Also, list service features where it is important to know the feature for a true identification of the service. This is a one paragraph high-level description of the service.

Service Requirements

Include in this section the general requirements that apply to all the basic and the alternative flows of the use case. Itemize the requirements and include a few sentences description of each one of them. Those requirements should be derived from the use cases described below.

Service Usage Attribute List

Include in this section a listing of the usage attributes that are common to all the basic and alternative flows of the use cases. A table format is preferable. Include information such as usage attribute category, usage attribute name, data type, possible values, and remarks. Reference the use case step and the requirement number that lead to this usage attribute.

Use Case

Basic Flow

Describe the basic flow steps of the use case. Itemize the steps and include a few sentences description of each step. Also, include a high-level state diagram that shows the general flow. Concentrate on the interfaces between the different sub-systems. Try not to go into great detail about each sub-system internals. This approach will be closer to the business level use case than that of the technical flow of information.

Basic Flow Requirements

Include in this section the requirements that are specific to this basic flow. You do not need to repeat the general requirements mentioned earlier. However, to indicate the continuity of those requirements numbering could be consecutive from the previous section.

Basic Flow Usage Attribute List

Use the same table format as for the general usage attribute list.

Alternative/Specific Flow (repeat as necessary)

Describe the alternative flow steps using the same format as the basic flow. Include any other interfaces that are presented in this alternative. Repeat the state diagram highlighting the changes of this alternative flow.

Alternative/Specific Flow Requirements

Same as for Basic Flow

Alternative/Specific Flow Usage Attribute List

Same as basic flow