



# **Server-Server Protocol XML Syntax**

Candidate Version 1.3 – 11 Oct 2005

---

**Open Mobile Alliance™**  
OMA-IMPS-WV-SSP\_XMLS-V1\_3-20051011-C

Use of this document is subject to all of the terms and conditions of the Use Agreement located at <http://www.openmobilealliance.org/UseAgreement.html>.

Unless this document is clearly designated as an approved specification, this document is a work in process, is not an approved Open Mobile Alliance™ specification, and is subject to revision or removal without notice.

You may use this document or any part of the document for internal or educational purposes only, provided you do not modify, edit or take out of context the information in this document in any manner. Information contained in this document may be used, at your sole risk, for any purposes. You may not use this document in any other manner without the prior written permission of the Open Mobile Alliance. The Open Mobile Alliance authorizes you to copy this document, provided that you retain all copyright and other proprietary notices contained in the original materials on any copies of the materials and that you comply strictly with these terms. This copyright permission does not constitute an endorsement of the products or services. The Open Mobile Alliance assumes no responsibility for errors or omissions in this document.

Each Open Mobile Alliance member has agreed to use reasonable endeavors to inform the Open Mobile Alliance in a timely manner of Essential IPR as it becomes aware that the Essential IPR is related to the prepared or published specification. However, the members do not have an obligation to conduct IPR searches. The declared Essential IPR is publicly available to members and non-members of the Open Mobile Alliance and may be found on the “OMA IPR Declarations” list at <http://www.openmobilealliance.org/ipr.html>. The Open Mobile Alliance has not conducted an independent IPR review of this document and the information contained herein, and makes no representations or warranties regarding third party IPR, including without limitation patents, copyrights or trade secret rights. This document may contain inventions for which you must obtain licenses from third parties before making, using or selling the inventions. Defined terms above are set forth in the schedule to the Open Mobile Alliance Application Form.

NO REPRESENTATIONS OR WARRANTIES (WHETHER EXPRESS OR IMPLIED) ARE MADE BY THE OPEN MOBILE ALLIANCE OR ANY OPEN MOBILE ALLIANCE MEMBER OR ITS AFFILIATES REGARDING ANY OF THE IPR'S REPRESENTED ON THE “OMA IPR DECLARATIONS” LIST, INCLUDING, BUT NOT LIMITED TO THE ACCURACY, COMPLETENESS, VALIDITY OR RELEVANCE OF THE INFORMATION OR WHETHER OR NOT SUCH RIGHTS ARE ESSENTIAL OR NON-ESSENTIAL.

THE OPEN MOBILE ALLIANCE IS NOT LIABLE FOR AND HEREBY DISCLAIMS ANY DIRECT, INDIRECT, PUNITIVE, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR EXEMPLARY DAMAGES ARISING OUT OF OR IN CONNECTION WITH THE USE OF DOCUMENTS AND THE INFORMATION CONTAINED IN THE DOCUMENTS.

© 2005 Open Mobile Alliance Ltd. All Rights Reserved.

Used with the permission of the Open Mobile Alliance Ltd. under the terms set forth above.

# Contents

1. SCOPE .....	5
2. REFERENCES .....	6
2.1 NORMATIVE REFERENCES .....	6
2.2 INFORMATIVE REFERENCES .....	6
3. TERMINOLOGY AND CONVENTIONS .....	7
3.1 CONVENTIONS .....	7
3.2 DEFINITIONS .....	7
3.3 ABBREVIATIONS .....	7
4. INTRODUCTION .....	8
5. DOCUMENT TYPE DEFINITION .....	9
6. DATA TYPE ASSIGNMENTS FOR SSP XML ELEMENTS .....	28
7. DATA TYPES FOR XML ELEMENTS AND ATTRIBUTES .....	34
7.1 REDEFINED AND NEW DATA TYPES .....	34
7.1.1 Boolean .....	34
7.1.2 MSISDN .....	34
7.1.3 URL .....	34
7.1.4 Integer .....	34
7.2 DATA TYPES FOR XML ATTRIBUTES .....	34
7.3 DATA TYPES FOR XML TERMINAL ELEMENTS .....	42
7.4 DESCRIPTION OF COMPOSED XML ELEMENTS .....	47
APPENDIX A. CHANGE HISTORY (INFORMATIVE) .....	49
A.1 APPROVED VERSION HISTORY .....	49
A.2 DRAFT/CANDIDATE VERSION 1.3 HISTORY .....	49
APPENDIX B. STATIC CONFORMANCE REQUIREMENTS .....	51
APPENDIX C. XML BINDING EXAMPLES (INFORMATIVE) .....	52
C.1 LOGIN TRANSACTION .....	52
C.2 STATUS PRIMITIVE WITH DETAILS .....	52
C.3 LOGOUT TRANSACTION .....	53
C.4 KEEPALIVE TRANSACTION .....	53
C.5 GETAVAILABLESERVICE TRANSACTION .....	53
C.6 SERVICEINDICATION TRANSACTION .....	54
C.7 SETSERVICEAGREEMENT TRANSACTION .....	54
C.8 GETUSERPROFILE TRANSACTION .....	55
C.9 UPDATEUSERPROFILE TRANSACTION .....	55

## Tables

Table 1: AnswerOptionID .....	42
Table 2: AnswerOptionText .....	42
Table 3: BlockedState .....	42
Table 4: CancelNote .....	43
Table 5: ChosenOptionID .....	43
Table 6: ContactListID .....	43
Table 7: ContactUser .....	43

Table 8: ContactUserSpec.....	43
Table 9: ContentID.....	43
Table 10: DateTime.....	43
Table 11: DeliveryTime.....	44
Table 12: Domain.....	44
Table 13: ExtendConversationID.....	44
Table 14: GroupID.....	44
Table 15: InText.....	44
Table 16: JoinedUser.....	44
Table 17: MessageSpec.....	45
Table 18: MSISDN.....	45
Table 19: PasswordDigest.....	45
Table 20: PresenceSublist.....	45
Table 21: Property.....	45
Table 22: ReasonText.....	45
Table 23: RequiresResponse.....	45
Table 24: SearchTerm.....	46
Table 25: ScreenName.....	46
Table 26: SecretToken.....	46
Table 27: StatusDescription.....	46
Table 28: SystemMessageID.....	46
Table 29.:SystemMessageText.....	46
Table 30: UPInfo.....	46
Table 31: UserID.....	47
Table 32: VerificationKey.....	47
Table 33: Version.....	47

# 1. Scope

The Instant Messaging and Presence Service (IMPS) includes four primary features:

- Presence
- Instant Messaging
- Groups
- Shared Content

Presence is the key enabling technology for IMPS. It includes client device availability (my phone is on/off, in a call), user status (available, unavailable, in a meeting), location, client device capabilities (voice, text, GPRS, multimedia) and searchable personal statuses such as mood (happy, angry) and hobbies (football, fishing, computing, dancing). Since presence information is personal, it is only made available according to the user's wishes - access control features put the control of the user presence information in the users' hands.

Instant Messaging (IM) is a familiar concept in both the mobile and desktop worlds. Desktop IM clients, two-way SMS and two-way paging are all forms of Instant Messaging. IMPS will enable interoperable mobile IM in concert with other innovative features to provide an enhanced user experience.

Groups or chat are a fun and familiar concept on the Internet. Both operators and end-users are able to create and manage groups. Users can invite their friends and family to chat in group discussions. Operators can build common interest groups where end-users can meet each other online.

Shared Content allows users and operators to setup their own storage area where they can post pictures, music and other multimedia content while enabling the sharing with other individuals and groups in an IM or chat session.

These features, taken in part or as a whole, provide the basis for innovative new services that build upon a common interoperable framework.

## 2. References

### 2.1 Normative References

- [IOPPROC] "OMA Interoperability Policy and Process", Version 1.1, Open Mobile Alliance™, OMA-IOP-Process-V1\_1, <http://www.openmobilealliance.org/>
- [CSP] "Client-Server Protocol Session and Transactions Version 1.3". Open Mobile Alliance™. <http://www.openmobilealliance.org/>
- [CSP DataType] "Client-Server Protocol Data Types Version 1.3". Open Mobile Alliance™. <http://www.openmobilealliance.org/>
- [RFC2119] "Keywords for using RFCs to Indicate Requirements levels", Bradner, S. URL:<http://www.ietf.org/rfc/rfc2119.txt?number=2119>
- [RFC2396] Uniform Resource Identifiers (URI): Generic Syntax. URL:<http://www.ietf.org/rfc/rfc2396.txt?number=2396>

### 2.2 Informative References

- [Arch] "IMPS Architecture Version 1.3", OMA-AD-IMPS-V1\_3, Open Mobile Alliance™. <http://www.openmobilealliance.org/>
- [CSP] "Client-Server Protocol Session and Transactions Version 1.3", OMA-TS-IMPS-CSP-V1\_3, Open Mobile Alliance™. <http://www.openmobilealliance.org/>
- [CSP DTD] "Client-Server Protocol XML Syntax Version 1.3", OMA-TS-IMPS-CSP-XMLS-V1\_3, Open Mobile Alliance™. <http://www.openmobilealliance.org/>
- [CSP Trans] "Client-Server Protocol Transport Bindings Version 1.3", OMA-TS-IMPS-CSP-Transport-V1\_3, Open Mobile Alliance™. <http://www.openmobilealliance.org/>
- [CSP DataType] "Client-Server Protocol Data Types Version 1.3", OMA-TS-IMPS-CSP-Data\_Types-V1\_3, Open Mobile Alliance™. <http://www.openmobilealliance.org/>
- [CSP SMS] "Client-Server Protocol Plain Text Syntax Version 1.3", OMA-TS-IMPS-CSP-PTS-V1\_3, Open Mobile Alliance™. <http://www.openmobilealliance.org/>
- [CSP WBXML] "Client-Server Protocol Binary XML Definition and Examples Version 1.3", OMA-TS-IMPS-CSP-\_WBXML-V1\_3, Open Mobile Alliance™. <http://www.openmobilealliance.org/>
- [PA] "Presence Attributes Version 1.3", OMA-TS-IMPS-PA-V1\_3, Open Mobile Alliance™. <http://www.openmobilealliance.org/>
- [PA DTD] "Presence Attribute XML Syntax Version 1.3", OMA-TS-IMPS-PA\_XMLS-V1\_3, Open Mobile Alliance™. <http://www.openmobilealliance.org/>
- [SSP] "Server-Server Protocol Semantics Version 1.3", OMA-TS-IMPS-SSP-V1\_3, Open Mobile Alliance™. <http://www.openmobilealliance.org/>
- [SSP Trans] "Server-Server Protocol Transport Binding Version 1.3", OMA-TS-IMPS-SSP\_Transport-V1\_3, Open Mobile Alliance™. <http://www.openmobilealliance.org/>

## 3. Terminology and Conventions

### 3.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope” and “Introduction”, are normative, unless they are explicitly indicated to be informative.

### 3.2 Definitions

None.

### 3.3 Abbreviations

<b>CSP</b>	Client-Server Protocol
<b>IMPS</b>	Instant Messaging and Presence Service
<b>OMA</b>	Open Mobile Alliance
<b>SSP</b>	Server-Server Protocol
<b>WV</b>	Wireless Village

## 4. Introduction

This document defines the valid XML documents representing a valid IMPS SSP protocol using Document Type Definition mechanism. IMPS Servers which support the SSP protocol [SSP], MUST conform to the DTD defined in this document. The base syntactical checking can be done by a validating XML parser. Further syntactical constraints are given in the data types assignment section. IMPS Servers MUST support protocol extensions to the SSP DTD.

For a request primitive, the TransactionMode element MUST be 'Request'. For a response primitive, the TransactionMode element MUST be 'Response'.



## 5. Document Type Definition

```

<!--
WV Server-Server Protocol (WV SSP) V1.3 Document Type Definition

Copyright Open Mobile Alliance Ltd., 2005
All rights reserved

WV-SSP is an XML language. Typical usage:
  <?xml version="1.0"?>
  <!DOCTYPE WV-SSP PUBLIC "-//OMA//DTD WV-SSP 1.3//EN"
    "http://www.openmobilealliance.org/DTD/WV-SSP.DTD"
    [<?oma-wv-ssp-ver supported-versions="1.3"?>]>
  <WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
    ...
  </WV-SSP-Message>

Terms and conditions of use are available from the
Open Mobile Alliance Ltd. web site at
http://www.openmobilealliance.org/useterms.html
-->

<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT WV-SSP-Message (SetupTransaction | Session)>
<!ATTLIST WV-SSP-Message xmlns CDATA #REQUIRED>

<!-- SetupTransaction is a transaction for session setup outside of an existing
session -->
<!ELEMENT SetupTransaction (SendSecretToken | LoginRequest | LoginResponse)>
<!ATTLIST SetupTransaction
  mode (Request | Response) #REQUIRED
  transactionID CDATA #REQUIRED>

<!-- SendSecretToken is a special request-like message without session and
MetaInfo -->
<!ELEMENT SendSecretToken (SecretToken)>
<!ATTLIST SendSecretToken
  serviceID CDATA #REQUIRED
  protocol CDATA #FIXED "WV-SSP"
  protocolVersion CDATA #FIXED "1.2">
<!ELEMENT SecretToken (#PCDATA)>
<!ATTLIST SecretToken
  encoding CDATA "base64">

<!-- LoginRequest is a special response-like message without Session and
MetaInfo -->
<!ELEMENT LoginRequest (PasswordDigest)>
<!ATTLIST LoginRequest
  serviceID CDATA #REQUIRED
  redirectHostId CDATA #IMPLIED
  timeToLive CDATA #IMPLIED>
<!ELEMENT PasswordDigest (#PCDATA)>
<!ATTLIST PasswordDigest
  encoding CDATA "base64">

<!-- LoginResponse -->
<!ELEMENT LoginResponse (Status, HostsList)>
<!ATTLIST LoginResponse

```

```
    sessionID    CDATA    #IMPLIED
    timeToLive   CDATA    #IMPLIED>
<!-- HostsList -->
<!ELEMENT HostsList (redirectHostId*)>
<!ELEMENT redirectHostId (#PCDATA)>

<!-- Status is contained in every normal response and it may also be a separate
message or can be sent in error cases of login phase messages -->
<!ELEMENT Status (StatusDescription?, DetailedResult?, SystemMessageList?,
TryAgainTimeout?)>
<!ATTLIST Status
    code CDATA    #REQUIRED>
<!ELEMENT DetailedResult (StatusDescription?, UserID*, GroupID*, ScreenName*,
messageID*, ContactListID*, Domain*)>
<!ATTLIST Status
    code CDATA    #REQUIRED>
<!ELEMENT StatusDescription (#PCDATA)>
<!ELEMENT TryAgainTimeout (#PCDATA)>
<!ELEMENT Domain (#PCDATA)>

<!-- Session -->
<!ELEMENT Session (Transaction+)>
<!ATTLIST Session
    sessionID    CDATA    #REQUIRED>

<!-- Transaction. It is a normal transaction inside an existing session -->
```

```

<!ELEMENT Transaction (Status | LogoutRequest | Disconnect | KeepAliveRequest |
  KeepAliveResponse | GetServiceRequest | ServiceList | ServiceNegotiation |
  ServiceAgreement | GetAcceptedContentTypesRequest |
  GetAcceptedContentTypesResponse | UserLoginRequest | UserLoginResponse |
  UserLogoutRequest | UserDisconnect | UserKeepAliveRequest |
  UserKeepAliveResponse | UserGetSPInfoRequest | UserGetSPInfoResponse |
  UserServiceRequest | UserServiceResponse | GetUserProfileRequest | UserProfile
  | UpdateUserProfileRequest | GetPublicProfileRequest |
  GetPublicProfileResponse | UpdatePublicProfile | SubscribeNotificationRequest
  | UnsubscribeNotificationRequest | NotificationRequest | SearchRequest |
  SearchResponse | StopSearchRequest | InviteRequest | InviteResponse |
  CancelInviteRequest | VerifyIDRequest | VerifyIDResponse | InviteUserRequest |
  InviteUserResponse | CancelInviteUserRequest | SystemMessageRequest |
  SystemMessageUser | CreateContactListRequest | CreateContactListResponse |
  DeleteContactListRequest | GetContactListRequest | GetContactListResponse |
  GetListMemberRequest | AddListMemberRequest | RemoveListMemberRequest |
  ContactListMemberResponse | GetListPropsRequest | SetListPropsRequest |
  ContactListPropsResponse | CreateAttrListRequest | DeleteAttrListRequest |
  GetAttrListRequest | GetAttrListResponse | SubscribeRequest |
  UnsubscribeRequest | SuspendPresenceNotifications | GetWatcherListRequest |
  GetWatcherListResponse | PresenceNotification | GetPresenceRequest |
  GetPresenceResponse | UpdatePresenceRequest | SendMessageRequest |
  SendMessageResponse | ForwardMessageRequest | ForwardMessageResponse |
  NewMessage | MessageDelivered | MessageNotification | GetMessageRequest |
  SetMessageDeliveryMethod | GetMessageListRequest | GetMessageListResponse |
  RejectMessageRequest | DeliveryStatusReport | BlockEntityRequest |
  GetBlockedRequest | GetBlockedResponse | IsUserBlocked | CreateGroupRequest |
  DeleteGroupRequest | JoinGroupRequest | JoinGroupResponse | LeaveGroupRequest
  | LeaveGroupIndication | GetJoinedMemberRequest | GetJoinedMemberResponse |
  GetGroupMemberRequest | GetGroupMemberResponse | AddGroupMemberRequest |
  RemoveGroupMemberRequest | MemberAccessRequest | GetGroupPropsRequest |
  GetGroupPropsResponse | SetGroupPropsRequest | RejectListRequest |
  RejectListResponse | SubscribeGroupChangeRequest |
  UnsubscribeGroupChangeRequest | GetGroupSubStatusRequest |
  GetGroupSubStatusResponse | GroupChangeNotice | ExtendConversationRequest |
  ExtendConversationResponse) >
<!ATTLIST Transaction
  mode (Request | Response) #REQUIRED
  transactionID CDATA #REQUIRED>

<!-- MetaInfo is contained in every normal request -->
<!ELEMENT MetaInfo ( Requestor ) >
<!ATTLIST MetaInfo
  clientOriginated ( Yes | No ) "Yes" >
<!ELEMENT Requestor ( User? )>
<!ATTLIST Requestor
  serviceID CDATA #REQUIRED>
<!ELEMENT User (ClientID?)>
<!ATTLIST User
  userID CDATA #REQUIRED>
<!ELEMENT ClientID ((URL | MSISDN), ApplicationID?)>
<!ELEMENT URL EMPTY >
<!ATTLIST URL
  url CDATA #REQUIRED>
<!ELEMENT MSISDN EMPTY >
<!ATTLIST MSISDN
  msisdn CDATA #REQUIRED>
<!ELEMENT ApplicationID EMPTY >

```

```

<!ATTLIST ApplicationID
  applicationID CDATA #REQUIRED>

<!-- Normal PROTOCOL MESSAGES detailed. Element types used first time in the
  protocol message is defined below the protocol message element type -->

<!-- LogoutRequest -->
<!ELEMENT LogoutRequest EMPTY>

<!-- Disconnect -->
<!ELEMENT Disconnect (Status?)>

<!-- KeepAliveRequest -->
<!ELEMENT KeepAliveRequest EMPTY >
<!ATTLIST KeepAliveRequest
  timeToLive CDATA #IMPLIED>

<!-- KeepAliveResponse -->
<!ELEMENT KeepAliveResponse (Status)>
<!ATTLIST KeepAliveResponse
  timeToLive CDATA #IMPLIED>

<!-- GetServiceRequest -->
<!ELEMENT GetServiceRequest EMPTY>

<!-- ServiceList -->
<!ELEMENT ServiceList (Status?, ServiceTree)>

<!-- ServiceTree -->
<!ELEMENT ServiceTree (SRV_SAP?, SRV_Common?, SRV_Presence?, SRV_IM?,
  SRV_Group?)>

<!ELEMENT SRV_SAP (SRV_ServiceNegotiation?, SRV_UserProfileMgmt?,
  SRV_ServiceRelay?, SRV_UserSessionMgmt?)>
<!ELEMENT SRV_ServiceNegotiation EMPTY>
<!ELEMENT SRV_UserProfileMgmt EMPTY>
<!ELEMENT SRV_ServiceRelay EMPTY>
<!ELEMENT SRV_UserSessionMgmt EMPTY>

<!ELEMENT SRV_Common (SRV_Invite?, SRV_ComplementaryInvite?, SRV_Search?,
  SRV_VerifyUser?, SRV_GetMap?, SRV_SGMNT?)>
<!ELEMENT SRV_Invite (SRV_InvitePresence?, SRV_InviteSharedContent?,
  SRV_InviteIM?, SRV_InviteGroup?)>
<!ELEMENT SRV_InvitePresence EMPTY>
<!ELEMENT SRV_InviteSharedContent EMPTY>
<!ELEMENT SRV_InviteIM EMPTY>
<!ELEMENT SRV_InviteGroup EMPTY>
<!ELEMENT SRV_InviteApp EMPTY>
<!ELEMENT SRV_ComplementaryInvite EMPTY>
<!ELEMENT SRV_Search (SRV_Group?, SRV_Other?, SVR_Advanced)>
<!ELEMENT SRV_Group EMPTY>
<!ELEMENT SRV_Other EMPTY><!ELEMENT SRV_Advanced EMPTY>
<!ELEMENT SRV_VerifyUser EMPTY>
<!ELEMENT SRV_GetMap EMPTY>
<!ELEMENT SRV_SGMNT EMPTY>

```

```

<!ELEMENT SRV_Presence (SRV_ContactListGet?, SRV_ContactListUpdate?,
  SRV_WatcherList?, SRV_ContactListAddress?)>
<!ELEMENT SRV_ContactListGet EMPTY>
<!ELEMENT SRV_ContactListUpdate EMPTY>
<!ELEMENT SRV_WatcherList EMPTY>
<!ELEMENT SRV_ContactListAddress EMPTY>
<!ELEMENT SRV_IM (SRV_SendMessage?, SRV_PushMessage?, SRV_MessageNotification?,
  SRV_GetMessage?, SRV_SetMessageDeliveryMethod?, SRV_GetMessageList?,
  SRV_RejectMessage?, SRV_DeliveryReport?, SRV_Blocking?, SRV_IsBlocked?,
  SRV_GroupHistory?)>
<!ELEMENT SRV_SendMessage (SRV_Group?, SRV_Contacts?)>
<!ELEMENT SRV_Contacts EMPTY>
<!ELEMENT SRV_PushMessage EMPTY>
<!ELEMENT SRV_MessageNotification EMPTY>
<!ELEMENT SRV_GetMessage EMPTY>
<!ELEMENT SRV_SetMessageDeliveryMethod EMPTY>
<!ELEMENT SRV_GetMessageList EMPTY>
<!ELEMENT SRV_RejectMessage EMPTY>
<!ELEMENT SRV_DeliveryReport EMPTY>
<!ELEMENT SRV_Blocking EMPTY>
<!ELEMENT SRV_IsBlocked EMPTY>
<!ELEMENT SRV_GroupHistory EMPTY>

<!ELEMENT SRV_Group (SRV_GroupMGMT?, SRV_GetMember?, SRV_MemberMGMT?,
  SRV_RejectList?, SRV_ExCon?)>
<!ELEMENT SRV_GroupMGMT EMPTY>
<!ELEMENT SRV_GetMember EMPTY>
<!ELEMENT SRV_MemberMGMT EMPTY>
<!ELEMENT SRV_RejectList EMPTY>
<!ELEMENT SRV_ExCon EMPTY>

<!-- ServiceNegotiation -->
<!ELEMENT ServiceNegotiation (ServiceTree) >
<!ATTLIST ServiceNegotiation
  subProtocol CDATA #IMPLIED
  timeToLive CDATA #IMPLIED>

<!-- ServiceAgreement -->
<!ELEMENT ServiceAgreement (Status, ServiceTree )>
<!ATTLIST ServiceAgreement
  subProtocol CDATA #IMPLIED
  timeToLive CDATA #IMPLIED>

<!-- GetAcceptedContentTypes -->
<!ELEMENT GetAcceptedContentTypesRequest (MetaInfo )>

<!ELEMENT GetAcceptedContentTypesResponse (AcceptedContentType* )>
<!ATTLIST GetAcceptedContentTypesResponse
  anyContent (Yes | No) #REQUIRED>

<!-- ***** -->
<!-- User Session Management -->
<!-- ***** -->

<!ELEMENT UserLoginRequest (MetaInfo, PasswordDigest, DigestSchema,
  SystemMessageResponseList?)>
<!ATTLIST UserLoginRequest

```

```

    applicationID CDATA #IMPLIED>
    password CDATA #IMPLIED
    timeToLive CDATA #IMPLIED>
<!ELEMENT DigestSchema (#PCDATA)>

<!ELEMENT UserLoginResponse (MetaInfo, Status)>
<!ATTLIST UserLoginResponse
    nonce CDATA #IMPLIED
    digestSchema CDATA #IMPLIED
    keepAliveTime CDATA #IMPLIED
    clientCapabilityRequest CDATA #IMPLIED>

<!ELEMENT UserLogoutRequest (MetaInfo)>

<!ELEMENT UserDisconnect (MetaInfo, Status)>
<!ELEMENT UserKeepAliveRequest (MetaInfo)>
<!ATTLIST UserKeepAliveRequest
    timeToLive CDATA #IMPLIED>

<!ELEMENT UserKeepAliveResponse (MetaInfo, Status)>
<!ATTLIST UserKeepAliveResponse
    keepAliveTime CDATA #IMPLIED>

<!ELEMENT UserGetSPInfoRequest (MetaInfo)>

<!ELEMENT UserGetSPInfoResponse (MetaInfo, Status, Logo?)>
<!ATTLIST UserGetSPInfoResponse
    name CDATA #IMPLIED
    description CDATA #IMPLIED
    url CDATA #IMPLIED>
<!ELEMENT Logo (ContentType, ContentSize?, ContentEncoding, ContentData)>
<!ELEMENT ContentType (#PCDATA)>
<!ELEMENT ContentData (#PCDATA)>
<!ELEMENT ContentSize (#PCDATA)>
<!ELEMENT ContentEncoding (#PCDATA)>

<!ELEMENT UserServiceRequest (MetaInfo, Functions, AllFunctionsRequest)>
<!ELEMENT UserServiceResponse (MetaInfo, Status, Functions?, AllFunctions?)>
<!ELEMENT AllFunctionsRequest (#PCDATA)>
<!ELEMENT AllFunctions (WVCSPFeat)>
<!ELEMENT Functions (WVCSPFeat)>
<!ELEMENT WVCSPFeat (FundamentalFeat?, PresenceFeat?, IMFeat?, GroupFeat?)>

<!ELEMENT FundamentalFeat ((MF | (ServiceFunc?, SearchFunc?, InviteFunc?,
    VerifyIDFunc?))?)>
<!ELEMENT PresenceFeat ((MP | (ContListFunc?, PresenceAuthFunc?,
    PresenceDeliverFunc?))?)>
<!ELEMENT IMFeat ((MM | (IMSendFunc?, IMReceiveFunc?, IMAuthFunc?))?)>
<!ELEMENT GroupFeat ((MG | (GroupMgmtFunc?, GroupUseFunc?, GroupAuthFunc?))?)>
<!ELEMENT MF EMPTY>
<!ELEMENT MP EMPTY>
<!ELEMENT MM EMPTY>
<!ELEMENT MG EMPTY>

<!ELEMENT ServiceFunc (GETSPI?, GETMAP?, SGMNT?)>
<!ELEMENT SearchFunc (SRCH?, ADVSR?, STSRC?)>
<!ELEMENT InviteFunc (INVIT?, CAINV?)>
<!ELEMENT VerifyIDFunc (VRID?)>

```

```

<!ELEMENT ContListFunc (GCLI?, CCLI?, DCLI?, MCLS?)>
<!ELEMENT PresenceAuthFunc (GETWL?)>
<!ELEMENT PresenceDeliverFunc (GETPR?, UPDPR?)>

<!ELEMENT IMSendFunc (MDELIV?, FWMSG?)>
<!ELEMENT IMReceiveFunc (SETD?, GETLM?, GETM?, REJCM?, NOTIF?, NEWM?)>
<!ELEMENT IMAuthFunc (GLBLU?, BLENT?)>

<!ELEMENT GroupMgmtFunc (CREAG?, DELGR?, GETGP?, SETGP?)>
<!ELEMENT GroupUseFunc (SUBGCN?, GRCHN?, EXCON?)>
<!ELEMENT GroupAuthFunc (GETGM?, ADDGM?, RMVGM?, MBRAC?, REJEC?, GETJU?)>

<!ELEMENT GETSPI EMPTY>
<!ELEMENT SGMNT EMPTY>
<!ELEMENT GETMAP EMPTY>
<!ELEMENT SRCH EMPTY>
<!ELEMENT ADVSR EMPTY>
<!ELEMENT STSRC EMPTY>
<!ELEMENT INVIT EMPTY>
<!ELEMENT CAINV EMPTY>
<!ELEMENT VRID EMPTY>

<!ELEMENT GCLI EMPTY>
<!ELEMENT CCLI EMPTY>
<!ELEMENT DCLI EMPTY>
<!ELEMENT MCLS EMPTY>
<!ELEMENT GETWL EMPTY>
<!ELEMENT GETPR EMPTY>
<!ELEMENT UPDPR EMPTY>

<!ELEMENT MDELIV EMPTY>
<!ELEMENT FWMSG EMPTY>
<!ELEMENT SETD EMPTY>
<!ELEMENT GETLM EMPTY>
<!ELEMENT GETM EMPTY>
<!ELEMENT REJCM EMPTY>
<!ELEMENT NOTIF EMPTY>
<!ELEMENT NEWM EMPTY>
<!ELEMENT GLBLU EMPTY>
<!ELEMENT BLENT EMPTY>

<!ELEMENT CREAG EMPTY>
<!ELEMENT DELGR EMPTY>
<!ELEMENT GETGP EMPTY>
<!ELEMENT SETGP EMPTY>
<!ELEMENT SUBGCN EMPTY>
<!ELEMENT GRCHN EMPTY>
<!ELEMENT EXCON EMPTY>
<!ELEMENT GETGM EMPTY>
<!ELEMENT ADDGM EMPTY>
<!ELEMENT RMVGM EMPTY>
<!ELEMENT MBRAC EMPTY>
<!ELEMENT REJEC EMPTY>
<!ELEMENT GETJU EMPTY>

<!-- ***** -->

```

```

<!-- User Profile -->
<!-- ***** -->

<!-- GetUserProfileRequest -->
<!ELEMENT GetUserProfileRequest (MetaInfo, UserID+)>
<!ELEMENT UserID EMPTY>
<!ATTLIST UserID
  userID CDATA #REQUIRED
  friendlyName CDATA #IMPLIED>

<!-- UserProfile -->
<!ELEMENT UserProfile (Status, UserProfileValue+)>
<!ELEMENT UserProfileValue (UPInfo+)>
<!ATTLIST UserProfileValue
  userID CDATA #REQUIRED>
<!ELEMENT UPInfo (ClientID, UPValue)>
<!ATTLIST UPInfo
  attr CDATA #REQUIRED>

<!ELEMENT UPValue (#PCDATA)>

<!-- UpdateUserProfileRequest -->
<!ELEMENT UpdateUserProfileRequest (MetaInfo, UserProfileValue+)>

<!-- ***** -->
<!-- GENERAL -->
<!-- ***** -->

<!-- GeneralNotification -->

<!ELEMENT SubscribeNotificationRequest (MetaInfo, NotificationTypeList? )>

<!ELEMENT NotificationTypeList(NotificationType+ )>
<!ELEMENT NotificationType (#PCDATA)>

<!ELEMENT UnsubscribeNotificationRequest (MetaInfo, NotificationTypeList? )>

<!ELEMENT Notification-Request (MetaInfo, NotificationType, (InUse | GroupID |
  ContactListID+ | InviteID | OnlineETEMHandling | (PresenceSubList, UserID*,
  ContactListID*, DefaultList) | SessionPriority | UserList)?)>

<!ELEMENT InUse EMPTY >
<!ATTLIST InUse
  listStatus (Active | Inactive) #REQUIRED>

<!ELEMENT InviteID EMPTY >
<!ATTLIST InviteID
  inviteID CDATA #REQUIRED>

<!ELEMENT DefaultList EMPTY >
<!ATTLIST DefaultList
  defaultList ( Yes | No ) #REQUIRED>

<!ELEMENT SessionPriority (#PCDATA)>

<!ELEMENT UserList (User*, ScreenName*)>

```



```

<!-- Public Profile -->
<!ELEMENT GetPublicProfileRequest (MetaInfo, UserID+)>

<!ELEMENT GetPublicProfileResponse (MetaInfo, Status, PublicProfile+)>

<!ELEMENT PublicProfile (UserID?, Property+)>

<!ELEMENT UpdatePublicProfileRequest (PublicProfile?)>
<!ATTLIST UpdatePublicProfileRequest
  clearPublicProfile (Yes | No) #REQUIRED>

<!-- SearchRequest -->
<!ELEMENT SearchRequest (MetaInfo, SearchTerm*)>
<!ATTLIST SearchRequest
  searchType      (G | U)          #IMPLIED
  advancedCriteria CDATA           #IMPLIED
  searchLimit     CDATA            #IMPLIED
  searchID        CDATA            #IMPLIED
  searchIndex     CDATA            #IMPLIED
  searchIndex     CDATA            #IMPLIED
  attr            CDATA            #REQUIRED
  value           CDATA            #REQUIRED
  id              CDATA            #IMPLIED>

<!-- SearchResponse -->
<!ELEMENT SearchResponse (Status, SearchResult?)>
<!ATTLIST SearchResponse
  searchID        CDATA            #IMPLIED
  searchFindings  CDATA            #REQUIRED
  searchIndex     CDATA            #REQUIRED
  completed       ( Yes | No ) #REQUIRED>
<!ELEMENT SearchResult ( (User | ScreenName)* | GroupID+ )>
<!ELEMENT GroupID EMPTY >
<!ATTLIST GroupID
  groupID CDATA #REQUIRED>
<!ELEMENT ScreenName (#PCDATA) >
<!ATTLIST ScreenName
  groupID CDATA #REQUIRED>

<!-- StopSearchRequest -->
<!ELEMENT StopSearchRequest (MetaInfo)>
<!ATTLIST StopSearchRequest
  searchID CDATA #REQUIRED>

<!-- InviteRequest -->
<!ELEMENT InviteRequest (MetaInfo, Inviting, Invited, GroupID?, ApplicationID?,
  AttributeList?, ContentIDList?, InviteNote?, ScreenName?)>
<!ATTLIST InviteRequest
  inviteID CDATA #REQUIRED
  inviteType (GR | IM | PR | SC | GM | AP | EC) #REQUIRED
  validity CDATA #IMPLIED>
<!ELEMENT Inviting (User | ScreenName)>
<!ELEMENT Invited (User | ScreenName | ContactListID | GroupID)>
<!ELEMENT ContactListID EMPTY>

```

```

<!ATTLIST ContactListID
  contactListID CDATA      #REQUIRED>
<!ELEMENT AttributeList (PresenceSubList)>

<!ELEMENT ContentIDList (ContentID+)>
<!ELEMENT ContentID EMPTY >
<!ATTLIST ContentID
  url CDATA      #REQUIRED>
<!ELEMENT InviteNote (#PCDATA)>

<!-- InviteResponse -->
<!ELEMENT InviteResponse (Status, Inviting, Responding, ResponseNote?,
  ScreenName?)>
<!ATTLIST InviteResponse
  inviteID      CDATA      #REQUIRED
  acceptance    ( Yes | No ) #REQUIRED>
<!ELEMENT Responding (User | ScreenName | GroupID)>
<!ELEMENT ResponseNote (#PCDATA)>

<!-- InviteUserRequest -->
<!ELEMENT InviteUserRequest (MetaInfo, Inviting, Invited, GroupID?,
  ApplicationID?, AttributeList?, ContentIDList?, InviteNote?, ScreenName?)>
<!ATTLIST InviteUserRequest
  inviteID      CDATA      #REQUIRED
  inviteType    (GR | IM | PR | SC | GM | AP | EC | EG)      #REQUIRED
  validity      CDATA      #IMPLIED>

<!-- InviteUserResponse -->
<!ELEMENT InviteUserResponse (Status, Inviting, Responding, ResponseNote?,
  ScreenName?)>
<!ATTLIST InviteUserResponse
  inviteID      CDATA      #REQUIRED
  acceptance    ( Yes | No ) #REQUIRED>

<!-- CancelInviteRequest -->
<!ELEMENT CancelInviteRequest (MetaInfo, Canceling, Canceled?, GroupID?,
  ApplicationID?, AttributeList?, ContentIDList?, CancelNote?, ScreenName?)>
<!ATTLIST CancelInviteRequest
  inviteID      CDATA      #REQUIRED>
<!ELEMENT Canceling (User | ScreenName)>
<!ELEMENT Canceled (User | GroupID | ContactListID)>
<!ELEMENT CancelNote (#PCDATA)>

<!-- CancelInviteUserRequest -->
<!ELEMENT CancelInviteUserRequest (MetaInfo, Canceling, Canceled, GroupID?,
  ApplicationID?, AttributeList?, ContentIDList?, CancelNote?, ScreenName?)>
<!ATTLIST CancelInviteUserRequest
  inviteID      CDATA      #REQUIRED>

<!-- GetMapRequest -->
<!ELEMENT GetMapRequest (MetaInfo, AcceptedContentType*, PresenceSubList?)>
<!ATTLIST GetMapRequest
  userID        CDATA      #IMPLIED>

<!ELEMENT AcceptedContentType EMPTY>
<!ATTLIST AcceptedContentType
  contentType   CDATA      #REQUIRED

```

```

    acceptedRichContentLength CDATA #REQUIRED
    contentPolicy CDATA #REQUIRED
    contentPolicyLimit CDATA #IMPLIED>

<!-- GetMapResponse -->
<!ELEMENT GetMapResponse (MetaInfo, Map | URL)>
<!ELEMENT Map (ContentType, ContentEncoding, ContentData)>

<!-- VerifyIDRequest -->
<!ELEMENT VerifyIDRequest (MetaInfo, WVIDList)>
<!ELEMENT WVIDList (VerifyUserID*, VerifyContactListID*, VerifyGroupID*,
    VerifyScreenName*, VerifyDomain*)>
<!ELEMENT VerifyUserID (DateTime?)>
<!ATTLIST VerifyUserID
    userID CDATA #REQUIRED>
<!ELEMENT VerifyContactListID (DateTime?)>
<!ATTLIST VerifyContactListID
    contactListID CDATA #REQUIRED>
<!ELEMENT VerifyGroupID (DateTime?)>
<!ATTLIST VerifyGroupID
    groupID CDATA #REQUIRED>
<!ELEMENT VerifyScreenName (DateTime?)>
<!ATTLIST VerifyScreenName
    screenName CDATA #REQUIRED>
<!ELEMENT VerifyDomain (DateTime?)>
<!ATTLIST VerifyDomain
    domain CDATA #REQUIRED>
<!ELEMENT DateTime (#PCDATA)>
<!ATTLIST DateTime
    format CDATA "iso8601">

<!-- VerifyIDResponse -->
<!ELEMENT VerifyIDResponse (Status, WVIDList)>

<!-- System Message -->
<!ELEMENT SystemMessageRequest (MetaInfo, SystemMessageList )>

<!ELEMENT SystemMessageUser (MetaInfo, SystemMessageResponseList )>

<!ELEMENT SystemMessageList (SystemMessage+)>
<!ELEMENT SystemMessageResponseList (SystemMessageResponse+)>
<!ELEMENT SystemMessage (SystemMessageID, RequiresResponse, SystemMessageText,
    AnswerOptions?, VerificationMechanism?)>
<!ELEMENT SystemMessageResponse (SystemMessageID, ChosenOptionID?,
    VerificationKey?)>
<!ELEMENT SystemMessageID (#PCDATA)>
<!ELEMENT RequiresResponse (#PCDATA)>
<!ELEMENT SystemMessageText (#PCDATA)>
<!ELEMENT AnswerOptions (AnswerOption+)>
<!ELEMENT AnswerOption (AnswerOptionID, AnswerOptionText)>
<!ELEMENT AnswerOptionID (#PCDATA)>
<!ELEMENT AnswerOptionText (#PCDATA)>
<!ELEMENT VerificationMechanism (InText | URL)>
<!ELEMENT InText (#PCDATA)>
<!ELEMENT ChosenOptionID (#PCDATA)>
<!ELEMENT VerificationKey (#PCDATA)>

```

```

<!-- ***** -->
<!-- CONTACT LIST -->
<!-- ***** -->

<!-- CreateContactListRequest -->
<!ELEMENT CreateContactListRequest (MetaInfo, ContactUser*)>
<!ATTLIST CreateContactListRequest
  contactListID CDATA #REQUIRED>
<!ELEMENT ContactUser EMPTY >
<!ATTLIST ContactUser
  userID CDATA #REQUIRED
  nick CDATA #IMPLIED>

<!-- CreateContactListResponse -->
<!ELEMENT CreateContactListResponse (Status, ContactListProperties)>
<!ATTLIST CreateContactListResponse
  contactListID CDATA #REQUIRED>

<!-- DeleteContactListRequest -->
<!ELEMENT DeleteContactListRequest (MetaInfo, ContactListID+)>

<!-- GetContactListRequest -->
<!ELEMENT GetContactListRequest (MetaInfo, Version)>
<!ELEMENT Version (#PCDATA)>
<!ATTLIST Version
  format CDATA "DateTime">

<!-- GetContactListResponse -->
<!ELEMENT GetContactListResponse (MetaInfo, Version, ContactListID*)>
<!ATTLIST GetListMemberRequest
  defaultContactListID CDATA #IMPLIED>

<!-- GetListMemberRequest -->
<!ELEMENT GetListMemberRequest (MetaInfo)>
<!ATTLIST GetListMemberRequest
  contactListID CDATA #REQUIRED>

<!-- AddListMemberRequest -->
<!ELEMENT AddListMemberRequest (MetaInfo, ContactUser+)>
<!ATTLIST AddListMemberRequest
  contactListID CDATA #REQUIRED
  authorizeAndGrant (Yes | No) #IMPLIED >

<!-- RemoveListMemberRequest -->
<!ELEMENT RemoveListMemberRequest (MetaInfo, ContactUserSpec+)>
<!ATTLIST RemoveListMemberRequest
  contactListID CDATA #REQUIRED>
<!ELEMENT ContactUserSpec EMPTY >
<!ATTLIST ContactUserSpec
  userID CDATA #IMPLIED
  nick CDATA #IMPLIED>

<!-- ContactListMemberResponse -->
<!ELEMENT ContactListMemberResponse (Status, ContactUser*)>
<!ATTLIST ContactListMemberResponse
  contactListID CDATA #REQUIRED>

```

```

<!-- GetListPropsRequest -->
<!ELEMENT GetListPropsRequest (MetaInfo)>
<!ATTLIST GetListPropsRequest
  contactListID CDATA #REQUIRED>

<!-- SetListPropsRequest -->
<!ELEMENT SetListPropsRequest (MetaInfo, ContactListProperties)>
<!ATTLIST SetListPropsRequest
  contactListID CDATA #REQUIRED>
<!ELEMENT ContactListProperties (Property*)>
<!ELEMENT Property (#PCDATA)>
<!ATTLIST Property
  prop CDATA #REQUIRED>

<!-- ContactListPropsResponse -->
<!ELEMENT ContactListPropsResponse (Status, ContactListProperties)>

<!-- CreateAttrListRequest -->
<!ELEMENT CreateAttrListRequest (MetaInfo, AttributeList, ContactListID*,
  UserID*)>
<!ATTLIST CreateAttrListRequest
  defaultList ( Yes | No ) #REQUIRED
  defaultNotify ( Yes | No ) #REQUIRED
  userNotify ( Yes | No ) #IMPLIED
  contactListNotify ( Yes | No ) #IMPLIED>

<!-- DeleteAttrListRequest -->
<!ELEMENT DeleteAttrListRequest (MetaInfo, ContactListID*, UserID*)>
<!ATTLIST DeleteAttrListRequest
  defaultList ( Yes | No ) #REQUIRED>

<!-- GetAttrListRequest -->
<!ELEMENT GetAttrListRequest (MetaInfo, AttributeList, ContactListID*, UserID*)>
<!ATTLIST GetAttrListRequest
  defaultList ( Yes | No ) #REQUIRED
  exportList ( Yes | No ) #REQUIRED>

<!-- GetAttrListResponse -->
<!ELEMENT GetAttrListResponse (Status, AttributeAssociation*,
  DefaultAttributeList?)>
<!ATTLIST GetAttrListResponse
  defaultList ( Yes | No ) #REQUIRED
  exportList ( Yes | No ) #REQUIRED>
<!ELEMENT AttributeAssociation ( (UserID | ContactListID), AttributeList)>
<!ATTLIST AttributeAssociation
  userNotify ( Yes | No ) #IMPLIED
  contactListNotify ( Yes | No ) #IMPLIED>

<!ELEMENT DefaultAttributeList (PresenceSubList )>
<!ATTLIST DefaultAttributeList
  defaultNotify ( Yes | No ) #REQUIRED>

<!-- ***** -->
<!-- PRESENCE -->
<!-- ***** -->

```

```

<!-- SubscribeRequest -->
<!ELEMENT SubscribeRequest (MetaInfo, UserID*, ContactListID*, AttributeList? ,
  AutoSubscribe)>
<!ELEMENT AutoSubscribe (#PCDATA)>

<!-- UnsubscribeRequest -->
<!ELEMENT UnsubscribeRequest (MetaInfo, UserID*, ContactListID*)>
<!-- AttributeList? -->

<!-- SuspendPresenceNotifications -->
<!ELEMENT SuspendPresenceNotifications (MetaInfo, UserID*, ContactListID*)>

<!-- PresenceNotification -->
<!ELEMENT PresenceNotification (MetaInfo, Subscribers, PresenceValue+,
  Version?)>
<!ELEMENT PresenceValue ( PresenceSubList, Version? )>
<!ATTLIST PresenceValue
  userID          CDATA          #REQUIRED>

<!ELEMENT PresenceSubList (#PCDATA)>
<!ATTLIST PresenceSubList
  xmlns          CDATA          #REQUIRED
  xmlns:Ext     CDATA          #IMPLIED>

<!-- GetWatcherListRequest -->
<!ELEMENT GetWatcherListRequest (MetaInfo)>

<!-- GetWatcherListResponse -->
<!ELEMENT GetWatcherListResponse (Status, UserID*)>

<!-- GetPresenceRequest -->
<!ELEMENT GetPresenceRequest (MetaInfo, (VerUserID | VerContactListID)+,
  AttributeList)>
<!ELEMENT VerUserID (Version?)>
<!ATTLIST VerUserID
  userID CDATA #REQUIRED>
<!ELEMENT VerContactListID (Version?)>
<!ATTLIST VerContactListID
  userID CDATA #REQUIRED>

<!-- GetPresenceResponse -->
<!ELEMENT GetPresenceResponse (Status, PresenceValue*, Version?)>

<!-- UpdatePresenceRequest -->
<!ELEMENT UpdatePresenceRequest (MetaInfo, PresenceValue+ )>

<!-- ***** -->
<!-- INSTANT MESSAGE -->
<!-- ***** -->

<!-- SendMessageRequest -->
<!ELEMENT SendMessageRequest (MetaInfo, MessageInfo, ContentData)>
<!ATTLIST SendMessageRequest
  deliveryReport ( Yes | No ) #REQUIRED >
<!ELEMENT MessageInfo (Recipient+, Sender, DateTime, Font?)>

```

```

<!ATTLIST MessageInfo
  messageID      CDATA      #IMPLIED
  messageURI     CDATA      #IMPLIED
  contentType    CDATA      #IMPLIED
  contentName    CDATA      #IMPLIED
  contentSize    CDATA      #IMPLIED
  validity       CDATA      #IMPLIED>
<!ELEMENT Recipient ((User | ScreenName | GroupID | ContactListID),
  RecipientDisplay?)>
<!ELEMENT Sender ((User | GroupID), SenderDisplay?)>
<!ELEMENT RecipientDisplay (User | ScreenName | GroupID | Name)>
<!ELEMENT Name (#PCDATA)>
<!ELEMENT SenderDisplay (UserID | ScreenName | GroupID | Name)>
<!ELEMENT ContentData (#PCDATA)>
<!ATTLIST ContentData
  contentType    CDATA      #REQUIRED
  encoding        CDATA      "base64">

<!ELEMENT Font (Size?, Style*, Color?)>
<!ELEMENT Size (#PCDATA)>
<!ELEMENT Style (#PCDATA)>
<!ELEMENT Color (#PCDATA)>

<!-- SendMessageResponse -->
<!ELEMENT SendMessageResponse (Status)>
<!ATTLIST SendMessageResponse
  messageID      CDATA      #REQUIRED>

<!-- ForwardMessageRequest -->
<!ELEMENT ForwardMessageRequest (MetaInfo, Recipient+)>
<!ATTLIST ForwardMessageRequest
  messageID      CDATA      #IMPLIED
  messageURI     CDATA      #IMPLIED>

<!-- ForwardMessageResponse -->
<!ELEMENT ForwardMessageResponse (Status)>
<!ATTLIST ForwardMessageResponse
  messageID      CDATA      #IMPLIED>

<!-- NewMessage -->
<!ELEMENT NewMessage ((MetaInfo | Status), RecipientIDs, MessageInfo,
  ContentData)>
<!ATTLIST NewMessage
  messageID      CDATA      #IMPLIED
  messageURI     CDATA      #IMPLIED>
<!ELEMENT RecipientIDs (UserID+)>

<!-- MessageDelivered -->
<!ELEMENT MessageDelivered ( MetaInfo | Status )>
<!ATTLIST MessageDelivered
  messageID      CDATA      #REQUIRED>

<!-- MessageNotification -->
<!ELEMENT MessageNotification ( MetaInfo, MessageInfo, RecipientIDs )>
<!ATTLIST MessageNotification
  messageID      CDATA      #REQUIRED>

<!-- GetMessageRequest -->

```

```

<!ELEMENT GetMessageRequest ( MetaInfo )>
<!ATTLIST GetMessageRequest
  messageID CDATA #REQUIRED>

<!-- SetMessageDeliveryMethod -->
<!ELEMENT SetMessageDeliveryMethod ( MetaInfo )>
<!ATTLIST SetMessageDeliveryMethod
  deliveryMethod (NotifyGet | Push) #REQUIRED
  groupID CDATA #IMPLIED
  groupContentLimit CDATA #IMPLIED>

<!-- GetMessageListRequest -->
<!ELEMENT GetMessageListRequest ( MetaInfo )>
<!ATTLIST GetMessageListRequest
  groupID CDATA #IMPLIED
  messageCount CDATA #IMPLIED>

<!-- GetMessageListResponse -->
<!ELEMENT GetMessageListResponse ( Status, MessageInfo )>
<!ATTLIST GetMessageListResponse
  messageTotalCount CDATA #IMPLIED>

<!-- RejectMessageRequest -->
<!ELEMENT RejectMessageRequest ( MetaInfo, MessageSpec* )>
<!ELEMENT MessageSpec EMPTY >
<!ATTLIST MessageSpec
  messageID CDATA #IMPLIED
  messageURI CDATA #IMPLIED>

<!-- DeliveryStatusReport -->
<!ELEMENT DeliveryStatusReport (MetaInfo, DeliveryResult, DeliveryTime?,
  MessageInfo)>
<!ELEMENT DeliveryResult (Status)>
<!ELEMENT DeliveryTime (#PCDATA)>

<!-- BlockEntityRequest -->
<!ELEMENT BlockEntityRequest ( MetaInfo, BlockList?, UnblockList?, GrantList?,
  UngrantList? )>
<!ATTLIST BlockEntityRequest
  blockListStatus (Active | Inactive) #REQUIRED
  grantListStatus (Active | Inactive) #REQUIRED>
<!ELEMENT BlockList (UserID*, ScreenName*, GroupID*, ContactListID*,
  ApplicationID*)>
<!ELEMENT UnblockList (UserID*, ScreenName*, GroupID*, ContactListID*,
  ApplicationID*)>
<!ELEMENT GrantList (UserID*, ScreenName*, GroupID*, ContactListID*,
  ApplicationID*)>
<!ELEMENT UngrantList (UserID*, ScreenName*, GroupID*, ContactListID*,
  ApplicationID*)>

<!-- GetBlockedRequest -->
<!ELEMENT GetBlockedRequest ( MetaInfo )>

<!-- GetBlockedResponse -->
<!ELEMENT GetBlockedResponse ( Status, BlockList, GrantList )>
<!ATTLIST GetBlockedResponse
  blockListStatus (Active | Inactive) #REQUIRED

```



```

    grantListStatus (Active | Inactive) #REQUIRED>

<!-- IsUserBlockedRequest -->
<!ELEMENT IsUserBlockedRequest ( MetaInfo, BlockingUser, BlockedUser)>
<!ELEMENT BlockingUser (UserID)>
<!ELEMENT BlockedUser (UserID)>

<!-- IsUserBlockedResponse -->
<!ELEMENT IsUserBlockedResponse (Status, BlockedState)>
<!ELEMENT BlockedState (#PCDATA)>

<!-- ***** -->
<!--          GROUP          -->
<!-- ***** -->

<!-- CreateGroupRequest -->
<!ELEMENT CreateGroupRequest (MetaInfo, GroupProperties, ScreenName?)>
<!ATTLIST CreateGroupRequest
  groupID CDATA #REQUIRED
  joinGroup (Yes | No) #REQUIRED
  subscribeNotif (Yes | No) #REQUIRED>
<!ELEMENT GroupProperties (Property+, WelcomeNote?)>
<!ELEMENT WelcomeNote (ContentData)>

<!-- DeleteGroupRequest -->
<!ELEMENT DeleteGroupRequest (MetaInfo)>
<!ATTLIST DeleteGroupRequest
  groupID CDATA #REQUIRED>

<!-- JoinGroupRequest -->
<!ELEMENT JoinGroupRequest (MetaInfo, ScreenName?, OwnProperties?)>
<!ATTLIST JoinGroupRequest
  groupID CDATA #REQUIRED
  joinedListRequest ( Yes | No ) #REQUIRED
  subscribeNotif (Yes | No) #REQUIRED
  joinedUsersAge CDATA #IMPLIED>

<!-- JoinGroupResponse -->
<!ELEMENT JoinGroupResponse (Status, JoinedList, WelcomeNote?)>
<!ELEMENT JoinedList (Name*)>

<!-- LeaveGroupRequest -->
<!ELEMENT LeaveGroupRequest (MetaInfo)>
<!ATTLIST LeaveGroupRequest
  groupID CDATA #REQUIRED>

<!-- LeaveGroupIndication -->
<!ELEMENT LeaveGroupIndication ((MetaInfo | Status), ReasonText)>
<!ELEMENT ReasonText (#PCDATA)>
<!ATTLIST LeaveGroupIndication
  groupID CDATA #IMPLIED>

<!-- GetJoinedMemberRequest -->
<!ELEMENT GetJoinedMemberRequest (MetaInfo)>
<!ATTLIST GetJoinedMemberRequest
  groupID CDATA #REQUIRED>

<!-- GetJoinedMemberResponse -->

```

```

<!ELEMENT GetJoinedMemberResponse (Status, JoinedUser*)>
<!ELEMENT JoinedUser (#PCDATA)>
<!ATTLIST JoinedUser
  userID    CDATA      #REQUIRED>

<!-- GetGroupMemberRequest -->
<!ELEMENT GetGroupMemberRequest (MetaInfo)>
<!ATTLIST GetGroupMemberRequest
  groupID   CDATA      #REQUIRED>

<!-- GetGroupMemberResponse -->
<!ELEMENT GetGroupMemberResponse (Status, Admins?, Moderators?, OrdinaryUsers?)>
<!ELEMENT Admins (UserID*)>
<!ELEMENT Moderators (UserID*)>
<!ELEMENT OrdinaryUsers (UserID*)>

<!-- AddGroupMemberRequest -->
<!ELEMENT AddGroupMemberRequest (MetaInfo, UserID*)>
<!ATTLIST AddGroupMemberRequest
  groupID   CDATA      #REQUIRED>

<!-- RemoveGroupMemberRequest -->
<!ELEMENT RemoveGroupMemberRequest (MetaInfo, UserID+)>
<!ATTLIST RemoveGroupMemberRequest
  groupID   CDATA      #REQUIRED>

<!-- MemberAccessRequest -->
<!ELEMENT MemberAccessRequest (MetaInfo, Admins?, Moderators?, OrdinaryUsers?)>
<!ATTLIST MemberAccessRequest
  groupID   CDATA      #REQUIRED>

<!-- GetGroupPropsRequest -->
<!ELEMENT GetGroupPropsRequest (MetaInfo)>
<!ATTLIST GetGroupPropsRequest
  groupID   CDATA      #REQUIRED>

<!-- GetGroupPropsResponse -->
<!ELEMENT GetGroupPropsResponse (Status, GroupProperties, OwnProperties)>
<!ELEMENT OwnProperties (Property+)>

<!-- SetGroupPropsRequest -->
<!ELEMENT SetGroupPropsRequest (MetaInfo, GroupProperties?, OwnProperties?)>
<!ATTLIST SetGroupPropsRequest
  groupID   CDATA      #REQUIRED>

<!-- RejectListRequest -->
<!ELEMENT RejectListRequest (MetaInfo, AddUsers, RemoveUsers)>
<!ATTLIST RejectListRequest
  groupID   CDATA      #REQUIRED>
<!ELEMENT AddUsers (UserID*)>
<!ELEMENT RemoveUsers (UserID*)>

<!-- RejectListResponse -->
<!ELEMENT RejectListResponse (Status, RejectList)>
<!ELEMENT RejectList (UserID*)>

<!-- SubscribeGroupChangeRequest -->
<!ELEMENT SubscribeGroupChangeRequest (MetaInfo)>

```

```

<!ATTLIST SubscribeGroupChangeRequest
  groupID CDATA #REQUIRED>

<!-- UnsubscribeGroupChangeRequest -->
<!ELEMENT UnsubscribeGroupChangeRequest (MetaInfo)>
<!ATTLIST UnsubscribeGroupChangeRequest
  groupID CDATA #REQUIRED>

<!-- GetGroupSubStatusRequest -->
<!ELEMENT GetGroupSubStatusRequest (MetaInfo)>
<!ATTLIST GetGroupSubStatusRequest
  groupID CDATA #REQUIRED>

<!-- GetGroupSubStatusResponse -->
<!ELEMENT GetGroupSubStatusResponse (Status)>
<!ATTLIST GetGroupSubStatusResponse
  groupID CDATA #REQUIRED
  subscribed (Yes | No) #REQUIRED>

<!-- GroupChangeNotice -->
<!ELEMENT GroupChangeNotice (MetaInfo, Subscribers, Joined?, Left?,
  JoinedBlocked?, LeftBlocked?, GroupProperties?, OwnProperties?)>
<!ATTLIST GroupChangeNotice
  groupID CDATA #REQUIRED>
<!ELEMENT Joined (Name*)>
<!ELEMENT Left (Name*)>
<!ELEMENT JoinedBlocked (UserMapList)>
<!ELEMENT LeftBlocked (UserList)>
<!ELEMENT UserMapList (UserMapping?)>
<!ELEMENT UserMapping (Mapping+)>
<!ELEMENT Mapping (SName, UserID?)>

<!-- ExtendConversationRequest -->
<!ELEMENT ExtendConversationRequest (ExtendConversationID, UserList,
  ExtendConversationUserID, WelcomeNote?, ScreenName)>
<!ATTLIST ExtendConversationRequest
  subscribeNotif (Yes | No) #REQUIRED>
<!ELEMENT ExtendConversationUserID (UserID)>

<!-- ExtendConversationRequest -->
<!ELEMENT ExtendConversationResponse (ExtendConversationID, GroupID, Status,
  ResponseNote?)>
<!ELEMENT> ExtendConversationID(#PCDATA)

```

## 6. Data Type Assignments for SSP XML ELEMENTS

This section provides the mapping of the informational elements in SSP primitives to XML element constructions or attributes. If the mapped value is an element construction, then the specified element construction inside the transaction element represents the SSP information element. If the mapped value is an attribute, then the information element is represented by the specified attribute of the container element (the transaction element).

Information Element	XML Element	XML Attribute
Accepted-Content-Type	AcceptedContentType	
Accepted-Content-Types-List	AcceptedContentType*	Accepted-Content-Types-List
Add-User-ID-List	AddUsers	
Advanced-Criteria		advancedCriteria
Agreed-Service-List	ServiceTree	
Agreed-Sub-Protocol		subProtocol
Agreed-Time-To-Live		timeToLive
Any-Content		anyContent
Application-ID		applicationID
Attribute-Association-List	AttributeAssociation+	
All-Functions	AllFunctions	
All-Functions-Request	AllFunctionsRequest	
Auto-Subscribe	AutoSubscribe	
Authorize-And-Grant		authorizeAndGrant
Blocked-List-Inuse	InUse	
Blocked-User	BlockedUser	
Blocked-State	BlockedState	
Blocking-User	BlockingUser	
Block-List-Status		blockListStatus
Block-Entity-List	BlockList	
Canceled-Content-ID-List	ContentIDList	
Canceled-User	Canceled	
Canceling-User	Canceling	
Cancel-Reason	CancelNote	
Clear-Public-Profile		clearPublicProfile

Client-ID	ClientID	
Client-Capability-Request		clientCapabilityRequest
Client-Originated		clientOriginated
Completed		completed
Contact-List-ID		contactListID
Contact-List-ID-List	ContactListID+	
Contact-List-ID-List of GetPresenceRequest	VerContactListID+	
Contact-List-Notify		contactListNotify
Contact-List-Props	ContactListProperties	
Contact-List-Version	Version	
Content	ContentData	
Default-Association-List	DefaultAttributeList	
Default-C-List-ID		defaultContactListID
Default-List	DefaultList	defaultList
Default-Notify		defaultNotify
Delivery-Result	DeliveryReport	
Delivery-Report-Request		deliveryReport
Delivery-Time	DeliveryTime	
Description		description
Desired-Service-List	ServiceTree	
Desired-Sub-Protocol		subProtocol
Detailed-Result	DetailedResult	
Digest-Schema		digestSchema
Export-List		exportList
ExtendConversation-ID	ExtendConversationID	
ExtendConversation-User-ID	ExtendConversationUserID	
Grant-List-Status		grantListStatus
Grant-Entity-List	GrantList	
Granted-List-Inuse	InUse	
Group-Content-Limit		groupContentLimit

Group-ID		groupID
Group-Prop-List	GroupProperties	
Invite-Acceptance		acceptance
Invite-Application-ID	ApplicationID	Invite-Application-ID
Invite-Content-ID-List	ContentIDList	
Invite-Group-ID	GroupID	
Invite-ID		inviteID
Invite-Presence-Attribute-List	AttributeList	
Invite-Reason	InviteNote	
Invite-Response	ResponseNote	
Invite-Type		inviteType
Invited-User	Invited	
Inviting-User	Inviting	
Join-Group		joinGroup
Joined-Blocked-Users-List	JoinedBlocked	Joined-Blocked-Users-List
Joined-Request		joinedListRequest
Joined-User-Screen-Name-List	JoinedUser+	
Joined-User-Screen-Name-List of GroupChangeNotice	Joined	
Joined-Users-Age		joinedUsersAge
Keep-Alive-Time		keepAliveTime
Left-Blocked-Users-List	LeftBlocked	
Left-User-Screen-Name-List	Left	
List-of-Hosts	HostsList	
Logo	Logo	Logo
Message-Count		messageCount
Message-Delivery-Method		deliveryMethod
Message-ID		messageID
Message-ID-List		messageID of MessageSpec+
Message-Info	MessageInfo	
Message-Total-Count		messageTotalCount

Message-URI		messageURI
Message-URI-List		messageID of MessageSpec+
Meta-Information	MetaInfo	
Name		name
Nonce		nonce
Not-Available-Functions	Functions	
Notification-Type	NotificationType	
Notification-Type-List	NotificationTypeList	
Own-Prop-List	OwnProperties	
Own-Screen-Name	ScreenName	
Password-Digest	PasswordDigest	
Password-String		password
Presence-Attribute-List	AttributeList	
Presence-Value-List	PresenceValue+	
Presence-Value-Version	Version	
Presence-Value-Version-List of GetPresenceRequest	Version of (VerUserID and VerContactListID)	
Presence-Value-Version-List	Version of (PresenceValue+)	
Protocol		protocol
Protocol-Version		protocolVersion
Public-Profile-List	PublicProfile+	Public-Profile-List
Reason-Text	ReasonText	
Recipients	Recipient+	
Recipient-UserID-List	RecipientIDs	
Redirect-HostID		redirectHostId
Reject-User-ID-List	RejectList	
Remove-User-ID-List	RemoveUsers	
Requested-Functions	Functions	
Responding-User	Responding	
Response-Note	ResponseNote	
Screen-Name	ScreenName	

Search-Criteria	SearchCriteria	
Search-Findings		searchFindings
Search-ID		searchID
Search-Index		searchIndex
Search-Limit		searchLimit
Search-Pair-List	SearchTerm+	
Search-Results	SearchResult	
Search-Type		searchType
SecretToken	SecretToken	
Service-ID		serviceID
Service-List	ServiceTree	
Session-ID		sessionID
Session-Priority	SessionPriority	
Status code		code
Status description		statusDescription
Status-Info	Status	
Subscribe-Notif		subscribeNotif
Subscribing-User-ID-List	Subscribers	
Subscribing-User-ID-List in AuthorizationResponse	SubscriberResult+	
Subscription-Status		subscribed
SupportedDigestSchema	DigestSchema	
System-Message-List	SystemMessageList	
System-Message-Response-List	SystemMessageResponseList	
Time-To-Live		timeToLive
Transaction-ID		transactionID
Unblock-Entity-List	UnblockList	
Ungrant-Entity-List	UngrantList	
Updated-User-Profile-List	UserProfileValue+	
URL		url
User-ID		userID



User-ID-List	UserID+	
User-ID-List of GetPresenceRequest	VerUserID+	
User-ID-List of (GetGroupMemberResponse or MemberAccessRequest)	OrdinaryUsers	
User-ID-List-Adm	Admins	
User-ID-List-Mod	Moderators	
User-List	ContactUser+	
User-List in RemoveListMemberRequest	ContactUserSpec+	
User-Notify		userNotify
User-Profile-List	UserProfileValue+	
Validity		validity
Welcome-Text	WelcomeNote	
WV-ID-List	WVIDList	

## 7. Data Types for XML Elements and Attributes

### 7.1 Redefined and new data types

#### 7.1.1 Boolean

A boolean value indicates either true or false.

It is encoded to XML as an enumerated type with possible values of:

- "Yes" : means the specified value is true.
- "No" : means the specified value is false.

These values are case sensitive. Other values are invalid.

#### 7.1.2 MSISDN

MSISDN is a string containing a valid MSISDN number.

#### 7.1.3 URL

The value of the type is a string of a Uniform Resource Locator. See [rfc2396].

#### 7.1.4 Integer

An integer is a number from 0-4294967295 expressed in decimal format.

### 7.2 Data types for XML Attributes

The XML attributes of the elements used are defined in this section. Description of the basic types can be found in [SSP].

XML Attribute	acceptance
Data type	Boolean
Description	Specifies if the invited party accepted the invitation.

XML Attribute	acceptedRichContentLength
Data type	Integer
Description	Maximum size of rich content that can be delivered to a client See [CSP].

XML Attribute	advancedCriteria
Data type	String
Description	See in [SSP].

XML Attribute	anyContent
Data type	Boolean
Description	Specifies that the provider server accepts all content types.

XML Attribute	applicationID
Data type	String
Description	Identifies an application . See CSP Data Types [CSP Data Type]

XML Attribute	authID
Data type	String
Description	Gives an identifier for an authentication request tuple. It is specified by the requestor server. Unique for a session. Case sensitive.

XML Attribute	authorizeAndGrant
Data type	Boolean
Description	See in [SSP].

XML Attribute	blockListStatus
Data type	Enum
Range	"Active"   "Inactive"
Description	Indicates if the block list is in use (“Active”) or not (“Inactive”). See [SSP].

XML Attribute	clientCapabilityRequest
Data type	Boolean
Description	Informs the Client that it needs to perform a Client Capability Request transaction.

XML Attribute	clientOriginated
Data type	Boolean
Description	Attribute of the Transaction method. It specifies whether the transaction is originated from a client (Yes) or from a Wireless Village server (No).

XML Attribute	code
Data type	Integer
Description	Defined only in the Status element. It specifies the status code. Possible status codes are defined in the [SSP].

XML Attribute	completed
Data type	Boolean
Description	Indicates the completion of the search on the server. 'No' if still searching and new results may come, 'Yes' if server has finished searching, and new results will not come anymore.

XML Attribute	contactListID
Data type	String
Description	Corresponds to the Contact-List-ID defined in [SSP]. Case insensitive.

XML Attribute	contentPolicy
Data type	Enum
Range	"N"   "R"   "C"
Description	An enumerated value that indicates whether if there is a special handling policy active regarding a media type. N – No policy R – Reject policy. C – Cost policy.

XML Attribute	contentPolicyLimit
Data type	String
Description	An integer value that indicates the special handling policy limitation regarding a particular media type. See [CSP]

XML Attribute	contactListNotify
Data type	Boolean
Description	Corresponds to the Contact-List-Notify defined in [SSP]

XML Attribute	contentType
Data type	String
Description	Specifies the content type of an instant message. Must be in MIME type format.

XML Attribute	defaultList
Data type	Boolean
Description	Specifies if the container attribute list operation element is for the default list of the user ("Yes") or for an ordinary list ("No").

XML Attribute	defaultListNotify
Data type	Boolean
Description	Corresponds to the Default-List-Notify defined in [SSP]

XML Attribute	deliveryMethod
Data type	Enum
Range	"NotifyGet"   "Push"
Description	Specifies the method used for message delivery. See [SSP] and [CSP].

XML Attribute	deliveryReport
Data type	Boolean
Description	Specifies if a delivery report is requested for the sent instant message.

XML Attribute	description
Data type	String
Description	Descriptive text for service provider.

XML Attribute	digestSchema
Data type	String
Description	Type of digest schema to use in user login transaction.

XML Attribute	encoding
Data type	String
Description	Defines the used encoding method for the content of the element. Encoding scheme is base64 by default (i.e. when this attribute is not specified).

XML Attribute	exportList
Data type	Boolean
Description	See [SSP].

XML Attribute	format
Data type	String
Description	Defines the format of the container element. In this version this value is only specified for Version, for what the default format is DateTime. This format means a value in DateTime format specified in [SSP].

XML Attribute	friendlyName
Data type	String
Description	The Friendly Name an IMPS user creates in his/her public profile.

XML Attribute	granted
Data type	Boolean
Description	Specifies for a user if his authentication request was allowed ("Yes") or not ("No").

XML Attribute	grantListStatus
Data type	Enum
Range	"Active"   "Inactive"
Description	Indicates if the grant list is in use ("Active") or not ("Inactive"). See [SSP].

XML Attribute	groupContentLimit
Data type	Integer
Description	Maximum size of message that can be sent from the specified group. It MUST NOT be indicated when Group-ID is missing. See [SSP].

XML Attribute	groupID
Data type	String
Description	Corresponds to the Group-ID defined in [SSP]. Case insensitive.

XML Attribute	inviteID
Data type	String
Description	See in [SSP]. Case sensitive.

XML Attribute	inviteType
Data type	Enum
Range	"GR"   "IM"   "PR"   "SC"   "GM"   "AP"   "EC"
Description	Defines the type of the invitation. In case of value "GR", the invitation is for a group, in case of value "IM", the invitation is for instant messaging, in case of "PR", the invitation is for a presence attribute, in case of "SC" the invitation is for a shared content. In case of "GM" the invitation is for group membership. . In case of "AP" the invitation is for End-to-End Application. EC is used for the Extend Private Group Conversation.

XML Attribute	joinGroup
Data type	Boolean
Description	Indicates that the newly created group is joined (or not) at creation time.

XML Attribute	joinedListRequest
Data type	Boolean
Description	See in [SSP].

XML Attribute	joinedUsersAge
Data type	Integer
Description	See in [SSP].

XML Attribute	keepAliveTime
Data type	Integer
Description	Validity time of the user session.

XML Attribute	messageCount
Data type	Integer
Description	The maximum number of message-info structures to be returned.

XML Attribute	messageID
Data type	String
Description	Defines a message identifier that uniquely identifies the message. The generator of the message must guarantee that the message identifier is globally unique. Must be in format msg-id of rfc2822. For the right-hand side of the msg-id, the domain name of the server should be used. Case sensitive.

XML Attribute	messageTotalCount
Data type	Integer
Description	When the Message-Count was in the request but there are more instant messages waiting than the amount requested in Message-Count, this value MUST indicate the total number of instant messages. See [SSP].

XML Attribute	messageURI
Data type	URI
Description	Specifies a message by a Uniform Resource Locator.

XML Attribute	mode
Data type	Integer
Range	"Request"   "Response"
Description	Attribute of the Transaction method. It specifies whether the transaction is a request of a response without parsing of the content of the transaction.

XML Attribute	msisdn
Data type	MSISDN
Description	Specifies a valid phone number, by which for example a client can be identified.

XML Attribute	name
Data type	String
Description	Name of the service provider.

XML Attribute	nick
Data type	String
Description	Defines a nickname for a user on the contact list.

XML Attribute	nonce
Data type	String
Description	Random string generated by server for user password digest.

XML Attribute	password
Data type	String
Description	The user password corresponding to the User-ID

XML Attribute	prop
Data type	String
Description	Defines a property name.

XML Attribute	protocol
Data type	String
Range	"WV-SSP"
Description	Defines the protocol used. Must be "WV-SSP".

XML Attribute	protocolVersion
Data type	String
Range	"1.3"
Description	Defines the version of the protocol used. Must be "1.3".

XML Attribute	redirectHostId
Data type	String
Description	Redirect host DNS name

XML Attribute	searchFindings
Data type	Integer
Description	See in [SSP].

XML Attribute	searchID
Data type	String
Description	See in [SSP]. Case sensitive.

XML Attribute	searchIndex
Data type	Integer
Description	See in [SSP].

XML Attribute	searchLimit
Data type	Integer
Description	See in [SSP].

XML Attribute	searchType
Data type	Enum
Range	"G"   "U"
Description	Defines the type of the search. In case of value "G", the search is for a group, in case of value "U", the search is for users.

XML Attribute	serviceID
Data type	String
Format	Defined in [SSP].
Description	See [SSP]. Case sensitive.



XML Attribute	sessionID
Data type	String
Description	Specifies the used session identifier. Case sensitive.

XML Attribute	subscribed
Data type	Boolean
Description	Used in a subscription status response. Specifies if the asked entity is subscribed ("Yes") to the group or not ("No").

XML Attribute	subscribeNotif
Data type	Boolean
Description	A flag indicating that the user wants to activate the group change notifications while joining the group.

XML Attribute	subProtocol
Data type	String
Description	During session negotiation it specifies the subprotocol name and version for proprietary extensions.

XML Attribute	timeToLive
Data type	Integer
Description	Defines the time in second how long the containing action is valid.

XML Attribute	transactionID
Data type	String
Description	Identifies a SSP transaction. The initiating party assigns this ID. Case sensitive.

XML Attribute	url
Data type	URL
Description	Identifies a content or a client by a Uniform Resource Locator.

XML Attribute	userID
Data type	String
Description	Corresponds to the User-ID defined in [SSP]. Case insensitive.

XML Attribute	userListNotify
Data type	Boolean
Description	Corresponds to the User-List-Notify defined in [SSP].

XML Attribute	validity
Data type	Integer
Description	Specifies the time in seconds how long the specified entity is regarded valid.

XML Attribute	value
Data type	String
Description	Specifies the value for the attribute and the operand of the search to be executed.

XML Attribute	xmlns
Data type	URL
Description	Specifies the namespace the message is using.

## 7.3 Data Types for XML Terminal Elements

This section will define the SSP specific or redefined XML elements. Description of the basic types and description of terminal elements common with CSP can be found in [CSP DataType]. If an XML terminal element definition is found in both this document and in [CSP DataType] then the given XML terminal elements is redefined by this document and the definition below must be used. Empty terminal elements representing protocol primitives are not listed here.

For description of the XML terminal elements inside the ServiceTree element see [SSP] document.

XML PCDATA	AnswerOptionID
Data type	Integer.
Format	Defined in section 7.1.4
Description	Identifies an answer option within a system message.
Range	Defined in section 7.1.4

**Table 1: AnswerOptionID**

XML PCDATA	AnswerOptionText
Data type	String
Format	Text string.
Description	The answer option to be shown to the end-user.
Range	Max 30 characters

**Table 2: AnswerOptionText**

XML PCDATA	BlockedState
Data type	Boolean
Format	Defined in section 7.1.1
Description	'Yes' means that the blocking user has blocked the requested user. No means that the blocking user has not blocked the requested user.
Range	Defined in section 7.1.1

**Table 3: BlockedState**

XML PCDATA	CancelNote
Data type	String
Format	
Description	Short descriptive text for canceling of invitation.
Range	

**Table 4: CancelNote**

XML PCDATA	ChosenOptionID
Data type	See AnswerOptionID.
Format	See AnswerOptionID.
Description	Identifies the answer option that the end-user chose from the options available in a system message.
Range	See AnswerOptionID.

**Table 5: ChosenOptionID**

XML PCDATA	ContactListID
Data type	EMPTY
Format	
Description	Container element for contactListID attribute.
Range	

**Table 6: ContactListID**

XML PCDATA	ContactUser
Data type	EMPTY
Format	
Description	Defines a user in a contact list by user id and nickname.
Range	

**Table 7: ContactUser**

XML PCDATA	ContactUserSpec
Data type	EMPTY
Format	
Description	Uniquely identifies a user in the contact list either by user id or nickname.
Range	

**Table 8: ContactUserSpec**

XML PCDATA	ContentID
Data type	EMPTY
Format	Defined in [CSP].
Description	Identifies a shared content by url attribute.
Range	

**Table 9: ContentID**

XML PCDATA	DateTime
Data type	Date and Time
Format	Specified in the forat attribute.
Description	Date and time.
Range	

**Table 10: DateTime**

XML PCDATA	DeliveryTime
Data type	DateTime
Format	Specified in the format attribute
Description	Date and time of message delivery.
Range	

**Table 11: DeliveryTime**

XML PCDATA	Domain
Data type	String
Format	
Description	Specifies the domain
Range	

**Table 12: Domain**

XML PCDATA	ExtendConversationID
Data type	String
Format	
Description	Identifies the Extended conversation
Range	

**Table 13: ExtendConversationID**

XML PCDATA	GroupID
Data type	EMPTY
Format	
Description	Container element for groupID attribute.
Range	

**Table 14: GroupID**

XML PCDATA	InText
Data type	String
Format	Text string.
Description	Text containing instructions for the end-user that describes what needs to be entered as verification key.
Range	Max 128 characters

**Table 15: InText**

XML PCDATA	JoinedUser
Data type	String
Format	
Description	Specifies a joined user of a group with screen name and UserID. The element contains the actual screen name of the user in the specified group, the userID attribute contains the UserID.
Range	

**Table 16: JoinedUser**

XML PCDATA	MessageSpec
Data type	EMPTY
Format	
Description	Uniquely identifies a message by message id or url.
Range	

**Table 17: MessageSpec**

XML PCDATA	MSISDN
Data type	EMPTY
Format	
Description	Container element for msisdn attribute.
Range	

**Table 18: MSISDN**

XML ELEMENT	PasswordDigest
Data type	Binary
Format	Defined in encoding attribute. BASE64 by default.
Description	The password digest used in login phase. See [SSP]
Range	

**Table 19: PasswordDigest**

XML ELEMENT	PresenceSubList
Data type	String
Format	As defined in the referred namespace.
Description	Presence attribute list with or without values. This element is used to give reference to the namespace (DTD) to be used under this specific tag.
Range	The namespace attribute points to a valid Wireless Village presence namespace.

**Table 20: PresenceSublist**

XML ELEMENT	Property
Data type	String
Format	
Description	Text containing a property information identified by the prop attribute and its value is contained in the element value.
Range	

**Table 21: Property**

XML ELEMENT	ReasonText
Data type	String
Format	
Description	Text describing a reason of an operation.
Range	

**Table 22: ReasonText**

XML ELEMENT	RequiresResponse
Data type	Boolean
Format	Defined in section 7.1.1
Description	'Yes' means that the server requires response to a System Message. 'No' means that the server does not require response to a System Message.
Range	Defined in section 7.1.1

**Table 23: RequiresResponse**

XML ELEMENT	SearchTerm
Data type	EMPTY
Format	
Description	Container element for search terms. Contains an attribute type (attr), a value, and an optional id (for advanced searches) with which the search operation must be carried out.
Range	

**Table 24: SearchTerm**

XML ELEMENT	ScreenName
Data type	String
Format	
Description	Defines the screen name of a user in a group. Group is specified in the groupID attribute. Element content is case insensitive.
Range	

**Table 25: ScreenName**

XML ELEMENT	SecretToken
Data type	Binary
Format	Defined in encoding attribute. By default BASE64 encoded
Description	Secret token used in login phase.
Range	

**Table 26: SecretToken**

XML ELEMENT	StatusDescription
Data type	String
Format	
Description	Text containing a status description in reponse of a request.
Range	

**Table 27: StatusDescription**

XML ELEMENT	SystemMessageID
Data type	String
Format	Text string. The SystemMessage-ID is case sensitive.
Description	Identifies a System Message
Range	Max 50 characters

**Table 28: SystemMessageID**

XML PCDATA	SystemMessageText
Data type	String
Format	Text string.
Description	The content of the system message to be shown to the end-user.
Range	Max 512 characters

**Table 29.:SystemMessageText**

XML ELEMENT	UPIInfo
Data type	String
Format	
Description	Text containing a user profile information indetified by the attr attribute and its value is contained in the element value.
Range	

**Table 30: UPIInfo**

XML ELEMENT	UserID
Data type	EMPTY
Format	
Description	Container element for userID attribute.
Range	

Table 31: UserID

XML PCDATA	VerificationKey
Data type	String
Format	Text string.
Description	Text containing the verification key entered by the end-user.
Range	Max 128 characters

Table 32: VerificationKey

XML ELEMENT	Version
Data type	EMPTY
Format	
Description	Specifies a version in the element vale. Format specified in the format attribute. By default format is DateTime defined in [CSP].
Range	

Table 33: Version

## 7.4 Description of Composed XML Elements

In this section will provide the descriptions of the composed XML elements that are neither assigned to an informational element nor are terminal elements.

XML Element	ClientID
Description	This element contains the Client-ID of the client (either URL or MSISDN) and an optional Application-ID.

XML Element	PresenceValue
Description	Contains a list of presence attributes along with its values (Presence-Value-List informational element) and an optional version information (Presence-Value-Version informational element)

XML Element	Recipient
Description	Defines the recipient of an instant message. The instant message must be delivered to the address defined in this field. Can contain an optinal RecipientDisplay element descibing what destination should be displayed to the recipient user (e.g. the group name where the message was originally sent).

XML Element	Requestor
Description	Represents the requestor of a request primitive.

XML Element	Sender
Description	Describes the sender of an instant message. Can contain the optional SenderDisplay element describing what source should be displayed to the recipient user.

XML Element	Session
Description	Session element represents transactions sended in the same sessions. Its sessionID attribute identifies the session between the requestor and the provider server.

XML Element	SetupTransaction
Description	Contains a transaction outside of n existing session. Possible child elements are the transactions for session setup.

XML Element	Transaction
Description	Contains an SSP transaction primitive.

XML Element	User
Description	Represents a specific user using a specific terminal.

XML Element	VerUserID
Description	Represents a versioned user identifier (User-ID-List informational element). See description of Presence-Value-Version-List in [SSP].

XML Element	VerContactListID
Description	Represents a versioned contact list identifier (Contact-List-ID-List informational element). See description of Presence-Value-Version-List in [SSP].

XML Element	WV-SSP-Message
Description	Represents the SSP protocol message



## Appendix A. Change History

(Informative)

### A.1 Approved Version History

Reference	Date	Description
n/a	n/a	No prior version

### A.2 Draft/Candidate Version 1.3 History

Document Identifier	Date	Sections	Description
Draft Versions OMA-IMPS-WV-SSP_XMLS-V1_3	9 Jan 2005	2, 5, 8, 9	OMA-IMPS-WV-SSP_Syntax-V1_3-20041217-D used as baseline Name changed from SSP_Syntax to SSP_XMLS Removed references to SCR documents Updated references to 1.3 specifications Updated version reference from 1.2 to 1.3 in examples and DTD Added SCR table (empty content)
	31 Jan 2005		SCR Restructuring
	31 Mar 2005		Updated with following approved CRs: - OMA-IM-2005-0089-Editorial-changes-SSP-Trans-and-XML-IMPS-1.3 - OMA-TS-IMPS-SSP_XMLS-V1_3-20051011-C.doc - OMA-TS-IMPS-SSP_XMLS-V1_3-20051011-C.doc - OMA-IM-2005-0067R04-IMPS-1.3-SSP-XML-Syntax-Changes-for-Content-Font-Formatting - OMA-IM-2005-0104-IMPS13-ExMove_SSP_XMLS - OMA-IM-2005-0113-IMPS13_MMCName-SSP_XMLS
	25 Apr 2005		Updated with following approved CRs: - OMA-IM-2005-0187-IMPS13_AdvSrc_SSP_XMLS - OMA-IM-2005-0258-ShowMap-SSP-XMLS - OMA-IM-2005-0283-IMPS1_3-CR-Presence-Authorization-Model-SSP-XMLS - OMA-IM-2005-0290-SSP-CRs-XMLS - OMA-IM-2005-0307-MultiSessionApp-SSP-XMLS
Draft Versions OMA-TS-IMPS-SSP_XMLS-V1_3	28 Jul 2005		Fixed copyright date on cover page. Changed Open Mobile Alliance to Open Mobile Alliance™ Following agreed CRs have been included: - OMA-IM-2005-0367-IMPS13-ArchRef-SSP_XMLS - OMA-IM-2005-0375R01-IMPS-13-CONRR-Correction-SSP-XMLS - OMA-IM-2005-0422-IMPS-13-SSP-XML-Syntax - OMA-IM-2005-0423R01-IMPS-13-SSP-XML-CONRR-Fixes - OMA-IM-2005-0431R01-IMPS-13-SSP-XMLS-DetailedResult-fix - OMA-IM-2005-0432-IMPS-13-SSP-XMLS-EC-fix - OMA-IM-2005-0441-IMPS-13-SSP-XMLS-GroupChangeNotice - OMA-IM-2005-0494-SSP-XMLS-Extend-IM - OMA-IM-2005-0496-SSP-XMLS-User-Profile - OMA-IM-2005-0527-IMPS13_DocUpdateAI-SSP_XMLS
	8 Aug 2005		Following agreed CRs have been included: - OMA-IM-2005-0560-MPS13_AIs_SSP-XMLS - OMA-IM-2005-0564-SSP-XMLS-MAPURL - OMA-IM-2005-0565-IMPS-1_3-SSP-XMLS-Remove-CLP - OMA-IM-2005-0577R01-IMPS13_SysMsgReqResp_SSP-XMLS - OMA-IM-2005-0590-IMPS-1_3-SSP-XMLS-scr update - OMA-IM-2005-0595-IMPS-1_3-SSP-XMLS-Missing-Elements - OMA-IM-2005-0599-IMPS-1_3-SSP-XMLS-IsUserBlocked
	26 Aug 2005		Editorial update: change the references to OMA documents. See mail Peter Arnby White space deleted

Document Identifier	Date	Sections	Description
	16 Sep 2005		Following agreed CRs have been included: - OMA-IM-2005-0663-IMPS-13-SSP_XMLS-PostRNAFixes.doc
Candidate Versions OMA-TS-IMPS-SSP_XMLS-V1_3	11 Oct 2005	n/a	Status changed to Candidate by TP TP ref # OMA-TP-2005-0279R01-IMPS-V1_3-for-Candidate-approval

## Appendix B. Static Conformance Requirements

The notation used in this appendix is specified in [IOPPROC].

Item	Function	Reference	Status	Requirement
IMPS-SSP-XMLS-S-1	XML encoding is well-formed	All	M	
IMPS-SSP-XMLS-S-2	XML encoding follows the DTD	Section 5	M	
IMPS-SSP-XMLS-S-3	XML encoding tolerates protocol extension	section 5	M	
IMPS-SSP-XMLS-S-4	The namespace identifier is "http://www.openmobilealliance.org /DTD/WV-SSP1.3"	section 5	M	
IMPS-SSP-XMLS-S-5	For a request primitive, the TransactionMode element must be 'Request'.	Section 5	M	
IMPS-SSP-XMLS-S-6	For a response primitive, the TransactionMode element must be 'Response'.	Section 5	M	
IMPS-SSP-XMLS-S-7	XML encoding is constraint by the data type assignment	Section 6 and Section 7	M	

## Appendix C. XML Binding Examples

(Informative)

Examples of selected SSP transactions are presented in this section.

### C.1 Login Transaction

```
<!-- Message from ServerA to ServerB -->
<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
  <SetupTransaction mode="Request" transactionID="0">
    <SendSecretToken serviceID="wv:@operator.hu" protocol="WV-SSP"
      procotolVersion="1.3">
      <SecretToken encoding="base64">
        R5R5FHJF47RY838289290050W0R989E0ER0R9E0392848GFJF8484JF84839294U745723934U7
      </SecretToken>
    </SendSecretToken>
  </SetupTransaction>
</WV-SSP-Message>
```

```
<!-- Message from ServerB to ServerA -->
<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
  <SetupTransaction mode="Response" transactionID="0">
    <LoginRequest serviceID="wv:@operator.hu" timeToLive="1314000">
      <PasswordDigest
        encoding="base64">FJS823HFUGF73HFBNE73BCJED34E60F5504W560001929394
      E52</PasswordDigest>
    </LoginRequest>
  </SetupTransaction>
</WV-SSP-Message>
```

```
<!-- Message from ServerA to ServerB -->
<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
  <SetupTransaction mode="Response" transactionID="0">
    <LoginResponse sessionID="47563@someone.za" timeToLive="1314000">
      <Status code="200"/>
    </LoginResponse>
  </SetupTransaction>
</WV-SSP-Message>
```

### C.2 Status primitive with details

```
<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
  <Session sessionID="123456@pgsm.hu">
    <Transaction mode="Response" transactionID="420042">
      <Status code="400">
        <StatusDescription>Bad request</StatusDescription>
      </Status>
    </Transaction>
    <Transaction mode="Response" transactionID="420043">
      <Status code="200"/>
    </Transaction>
  </Session>
</WV-SSP-Message>
```

### C.3 Logout Transaction

```
<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@someone.za">
  <Transaction mode="Request" transactionID="5006">
    <LogoutRequest/>
  </Transaction>
</Session>
</WV-SSP-Message>
```

```
<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@operator.hu">
  <Transaction mode="Response" transactionID="5006">
    <Disconnect>
      <Status code="200"/>
    </Disconnect>
  </Transaction>
</Session>
</WV-SSP-Message>
```

### C.4 KeepAlive transaction

```
<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@someone.za">
  <Transaction mode="Request" transactionID="5003">
    <KeepAliveRequest timeToLive="3600"/>
  </Transaction>
</Session>
</WV-SSP-Message>
```

```
<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@operator.hu">
  <Transaction mode="Response" transactionID="5003">
    <KeepAliveResponse timeToLive="3600">
      <Status code="200"/>
    </KeepAliveResponse>
  </Transaction>
</Session>
</WV-SSP-Message>
```

### C.5 GetAvailableService transaction

```
<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@someone.za">
  <Transaction mode="Request" transactionID="5002">
    <GetServiceRequest/>
  </Transaction>
</Session>
</WV-SSP-Message>
```

```
<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@operator.hu">
  <Transaction mode="Response" transactionID="5002">
    <ServiceList>
      <Status code="200"/>
    </ServiceList>
  </Transaction>
</Session>
</WV-SSP-Message>
```

```

    <ServiceTree>
      <SRV_SAP/>
    <SRV_Common/>
  <SRV_Presence/>
  <SRV_IM/>
  <SRV_Group/>
  <ServiceTree>
</ServiceList>
</Transaction>
</Session>
</WV-SSP-Message>

```

## C.6 ServiceIndication transaction

```

<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@operator.hu">
  <Transaction mode="Response" transactionID="5002">
    <ServiceList>
      <Status code="200"/>
    <ServiceTree>
      <SRV_SAP/>
    <SRV_Common/>
  <SRV_Presence/>
  <SRV_IM/>
  <SRV_Group/>
  <ServiceTree>
</ServiceList>
</Transaction>
</Session>
</WV-SSP-Message>

```

## C.7 SetServiceAgreement transaction

```

<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@someone.za">
  <Transaction mode="Request" transactionID="113">
    <ServiceNegotiation timeToLive="3600">
      <ServiceTree>
        <SRV_SAP/>
      <SRV_Common/>
    <SRV_Presence/>
    <SRV_IM/>
    <SRV_Group/>
    <ServiceTree>
  </ServiceNegotiation>
</Transaction>
</Session>
</WV-SSP-Message>

<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@operator.hu">
  <Transaction mode="Response" transactionID="113">
    <ServiceAgreement timeToLive="3600">
      <Status code="200"/>
    <ServiceTree>

```

```

    <SRV_SAP/>
  <SRV_Common/>
  <SRV_Presence/>
  <SRV_IM/>
  <SRV_Group/>
  <ServiceTree>
</ServiceAgreement>
</Transaction>
</Session>
</WV-SSP-Message>

```

## C.8 GetUserProfile transaction

```

<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@someone.za">
  <Transaction mode="Request" transactionID="113">
    <GetUserProfileRequest userID="wv:doboz.bela@operator.za">
      <MetaInfo clientOriginated="Yes">
        <Requestor>
          <User userID="wv:quador@operator.hu">
            <ClientID url="http://www.wireless-village.org/applications/MudServer" />
          </User>
        </Requestor>
      </MetaInfo>
    </GetUserProfileRequest>
  </Transaction>
</Session>
</WV-SSP-Message>

```

```

<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@operator.hu">
  <Transaction mode="Response" transactionID="113">
    <UserProfile>
      <Status code="200"/>
      <UserProfileValue userID="wv:doboz.bela@operator.za">
        <UPInfo attr="Services.Presence.Status">ON</UPInfo>
        <UPInfo attr="Services.Common.VHSE">pgsm.hu</UPInfo>
      </UserProfileValue>
    </UserProfile>
  </Transaction>
</Session>
</WV-SSP-Message>

```

## C.9 UpdateUserProfile transaction

```

<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@someone.za">
  <Transaction mode="Request" transactionID="113">
    <UpdateUserProfileRequest userID="wv:doboz.bela@operator.za">
      <MetaInfo clientOriginated="Yes">
        <Requestor>
          <User userID="wv:quador@operator.hu">
            <ClientID url="http://www.wireless-village.org/applications/MudServer" />
          </User>
        </Requestor>
      </MetaInfo>
    </UpdateUserProfileRequest>
  </Transaction>
</Session>
</WV-SSP-Message>

```

```
<UserProfileValue>
  <UPInfo attr="Services.Presence.Status">ON</UPInfo>
  <UPInfo attr="Services.Common.VHSE">pgsm.hu</UPInfo>
</UserProfileValue>
</GetUserProfileRequest>
</Transaction>
</Session>
</WV-SSP-Message>
```

```
<WV-SSP-Message xmlns="http://www.openmobilealliance.org/DTD/WV-SSP1.3">
<Session sessionID="123456@operator.hu">
  <Transaction mode="Response" transactionID="113">
    <Status code="200"/>
  </Transaction>
</Session>
</WV-SSP-Message>
```