

Web Service: wfApi

Web Service: wfApi

Target Namespace: <http://faxcoreserver/services/>

Port *wfApiSoap* [Port type](#) [Source code](#)

Location: <http://co.faxcore.com/webservices/wfapi.asmx>

Protocol: SOAP

Transport protocol: SOAP over HTTP

Operations:

1. [ActivateUser](#) [Detail](#) [Source code](#)
2. [ActivateUsers](#) [Detail](#) [Source code](#)
3. [AssignMessage](#) [Detail](#) [Source code](#)
4. [AuthenticateUser](#) [Detail](#) [Source code](#)
5. [CancelMessage](#) [Detail](#) [Source code](#)
6. [CreateInboundRoute](#) [Detail](#) [Source code](#)
7. [CreateUser](#) [Detail](#) [Source code](#)
8. [CreateUser](#) [Detail](#) [Source code](#)
9. [CreateUser](#) [Detail](#) [Source code](#)
10. [DeactivateUser](#) [Detail](#) [Source code](#)
11. [DeactivateUsers](#) [Detail](#) [Source code](#)
12. [DelegateMessage](#) [Detail](#) [Source code](#)
13. [DeleteMessage](#) [Detail](#) [Source code](#)
14. [DownloadMessageByte](#) [Detail](#) [Source code](#)
15. [DownloadMessageByte](#) [Detail](#) [Source code](#)
16. [DownloadMessageByte](#) [Detail](#) [Source code](#)
17. [DownloadMessageByte](#) [Detail](#) [Source code](#)
18. [ForwardToUser](#) [Detail](#) [Source code](#)
19. [GetDomainList](#) [Detail](#) [Source code](#)
20. [GetMessageDetails](#) [Detail](#) [Source code](#)
21. [GetMessageList](#) [Detail](#) [Source code](#)
22. [GetMessageList](#) [Detail](#) [Source code](#)
23. [GetMessageList](#) [Detail](#) [Source code](#)
24. [GetMessageList](#) [Detail](#) [Source code](#)
25. [GetMessageList](#) [Detail](#) [Source code](#)
26. [GetMessageList](#) [Detail](#) [Source code](#)
27. [GetMessageList](#) [Detail](#) [Source code](#)
28. [GetMessageList](#) [Detail](#) [Source code](#)
29. [GetMessageStatus](#) [Detail](#) [Source code](#)
30. [GetMsgFolders](#) [Detail](#) [Source code](#)
31. [GetUserDetails](#) [Detail](#) [Source code](#)
32. [GetUserDetails](#) [Detail](#) [Source code](#)

33. [GetUserFaxSetting](#) [Detail](#) [Source code](#)
34. [GetUserFaxSetting](#) [Detail](#) [Source code](#)
35. [GetUserList](#) [Detail](#) [Source code](#)
36. [GetUserList](#) [Detail](#) [Source code](#)
37. [IsMessageDeleted](#) [Detail](#) [Source code](#)
38. [IsMessageRead](#) [Detail](#) [Source code](#)
39. [MarkMsgAsDownloaded](#) [Detail](#) [Source code](#)
40. [MarkMsgAsRead](#) [Detail](#) [Source code](#)
41. [MarkMsgAsUnread](#) [Detail](#) [Source code](#)
42. [RemoveUser](#) [Detail](#) [Source code](#)
43. [RetryFailed](#) [Detail](#) [Source code](#)
44. [SearchIBRouteRules](#) [Detail](#) [Source code](#)
45. [SendMessage](#) [Detail](#) [Source code](#)
46. [SendMessage](#) [Detail](#) [Source code](#)
47. [SendMessage](#) [Detail](#) [Source code](#)
48. [SendMessage](#) [Detail](#) [Source code](#)
49. [SendMixedMode](#) [Detail](#) [Source code](#)
50. [ServerStatus](#) [Detail](#) [Source code](#)
51. [UpdateUser](#) [Detail](#) [Source code](#)
52. [UpdateUser](#) [Detail](#) [Source code](#)
53. [UpdateUser](#) [Detail](#) [Source code](#)

Port *wfApiSoap12* [Port type](#) [Source code](#)

Location: <http://co.faxcore.com/webservices/wfapi.asmx>

Protocol: SOAP

Transport protocol: SOAP over HTTP

Operations:

1. [ActivateUser](#) [Detail](#) [Source code](#)
2. [ActivateUsers](#) [Detail](#) [Source code](#)
3. [AssignMessage](#) [Detail](#) [Source code](#)
4. [AuthenticateUser](#) [Detail](#) [Source code](#)
5. [CancelMessage](#) [Detail](#) [Source code](#)
6. [CreateInboundRoute](#) [Detail](#) [Source code](#)
7. [CreateUser](#) [Detail](#) [Source code](#)
8. [CreateUser](#) [Detail](#) [Source code](#)
9. [CreateUser](#) [Detail](#) [Source code](#)
10. [DeactivateUser](#) [Detail](#) [Source code](#)
11. [DeactivateUsers](#) [Detail](#) [Source code](#)
12. [DelegateMessage](#) [Detail](#) [Source code](#)
13. [DeleteMessage](#) [Detail](#) [Source code](#)
14. [DownloadMessageByte](#) [Detail](#) [Source code](#)
15. [DownloadMessageByte](#) [Detail](#) [Source code](#)
16. [DownloadMessageByte](#) [Detail](#) [Source code](#)

17. [DownloadMessageByte](#) [Detail](#) [Source code](#)
18. [ForwardToUser](#) [Detail](#) [Source code](#)
19. [GetDomainList](#) [Detail](#) [Source code](#)
20. [GetMessageDetails](#) [Detail](#) [Source code](#)
21. [GetMessageList](#) [Detail](#) [Source code](#)
22. [GetMessageList](#) [Detail](#) [Source code](#)
23. [GetMessageList](#) [Detail](#) [Source code](#)
24. [GetMessageList](#) [Detail](#) [Source code](#)
25. [GetMessageList](#) [Detail](#) [Source code](#)
26. [GetMessageList](#) [Detail](#) [Source code](#)
27. [GetMessageList](#) [Detail](#) [Source code](#)
28. [GetMessageList](#) [Detail](#) [Source code](#)
29. [GetMessageStatus](#) [Detail](#) [Source code](#)
30. [GetMsgFolders](#) [Detail](#) [Source code](#)
31. [GetUserDetails](#) [Detail](#) [Source code](#)
32. [GetUserDetails](#) [Detail](#) [Source code](#)
33. [GetUserFaxSetting](#) [Detail](#) [Source code](#)
34. [GetUserFaxSetting](#) [Detail](#) [Source code](#)
35. [GetUserList](#) [Detail](#) [Source code](#)
36. [GetUserList](#) [Detail](#) [Source code](#)
37. [IsMessageDeleted](#) [Detail](#) [Source code](#)
38. [IsMessageRead](#) [Detail](#) [Source code](#)
39. [MarkMsgAsDownloaded](#) [Detail](#) [Source code](#)
40. [MarkMsgAsRead](#) [Detail](#) [Source code](#)
41. [MarkMsgAsUnread](#) [Detail](#) [Source code](#)
42. [RemoveUser](#) [Detail](#) [Source code](#)
43. [RetryFailed](#) [Detail](#) [Source code](#)
44. [SearchIBRouteRules](#) [Detail](#) [Source code](#)
45. [SendMessage](#) [Detail](#) [Source code](#)
46. [SendMessage](#) [Detail](#) [Source code](#)
47. [SendMessage](#) [Detail](#) [Source code](#)
48. [SendMessage](#) [Detail](#) [Source code](#)
49. [SendMixedMode](#) [Detail](#) [Source code](#)
50. [ServerStatus](#) [Detail](#) [Source code](#)
51. [UpdateUser](#) [Detail](#) [Source code](#)
52. [UpdateUser](#) [Detail](#) [Source code](#)
53. [UpdateUser](#) [Detail](#) [Source code](#)

Operations

Port type ***wfApiSoap*** [Source code](#)

1. ActivateUser

[Source code](#)

Description: Set user to be active

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/ActivateUser

Input: ActivateUserSoapIn (soap:body, use = literal) [Source code](#)

parameters type *ActivateUser*

- adminID type *long*
- userID type *long*

Output: ActivateUserSoapOut (soap:body, use = literal) [Source code](#)

parameters type *ActivateUserResponse*

- ActivateUserResult - optional; type *string*

2. ActivateUsers

[Source code](#)

Description: Set a list of users to be active

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/ActivateUsers

Input: ActivateUsersSoapIn (soap:body, use = literal) [Source code](#)

parameters type *ActivateUsers*

- adminID type *long*
- userIDList - optional; type *ArrayOfLong*
 - long - optional, unbounded; type *long*

Output: ActivateUsersSoapOut (soap:body, use = literal) [Source code](#)

parameters type *ActivateUsersResponse*

- ActivateUsersResult - optional; type *string*

3. AssignMessage

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/AssignMessage

Input: AssignMessageSoapIn (soap:body, use = literal) [Source code](#)

parameters type *AssignMessage*

- userID - optional; type *string*
- msgID - optional; type *string*
- usernames - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*

Output: AssignMessageSoapOut (soap:body, use = literal) [Source code](#)

parameters type *AssignMessageResponse*

- AssignMessageResult - optional; type *string*

4. AuthenticateUser

[Source code](#)

Description: Authenticate user via SDK

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/AuthenticateUser>

Input: AuthenticateUserSoapIn (soap:body, use = literal) [Source code](#)

parameters type *AuthenticateUser*

- *userName* - optional; type *string*
- *password* - optional; type *string*

Output: AuthenticateUserSoapOut (soap:body, use = literal) [Source code](#)

parameters type *AuthenticateUserResponse*

- *AuthenticateUserResult* type *long*

5. CancelMessage

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/CancelMessage>

Input: CancelMessageSoapIn (soap:body, use = literal) [Source code](#)

parameters type *CancelMessage*

- *msgID* - optional; type *string*

Output: CancelMessageSoapOut (soap:body, use = literal) [Source code](#)

parameters type *CancelMessageResponse*

- *CancelMessageResult* - optional; type *string*

6. CreateInboundRoute

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/CreateInboundRoute>

Input: CreateInboundRouteSoapIn (soap:body, use = literal) [Source code](#)

parameters type *CreateInboundRoute*

- *userID* type *long*
- *forwardedUserName* - optional; type *string*
- *condition1* type *ConditionType* - type *string* with restriction - enum { 'None', 'BarCode', 'FaxPortNo', 'RoutingDigits', 'RemoteCallerID', 'RemoteCSID', 'ServerName' }
- *conditionExp1* - optional; type *string*
- *isAndOperator* type *boolean*
- *condition2* type *ConditionType* - type *string* with restriction - enum { 'None', 'BarCode', 'FaxPortNo', 'RoutingDigits', 'RemoteCallerID', 'RemoteCSID', 'ServerName' }
- *conditionExp2* - optional; type *string*
- *isActive* type *boolean*
- *priority* type *int*

Output: CreateInboundRouteSoapOut (soap:body, use = literal) [Source code](#)

parameters type *CreateInboundRouteResponse*

- *CreateInboundRouteResult* - optional; type *string*

7. CreateUser

[Source code](#)

Description: Create new user with default NOR/NOS=false

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/CreateNewUser1>

CreateNewUser1SoapIn (soap:body, use = literal) [Source code](#)

Input:

parameters type *CreateNewUser1*

- userID type *long*
- domainName - optional; type *string*
- userName - optional; type *string*
- password - optional; type *string*
- role type *Role* - type *string* with restriction - enum { 'DomainAdmin', 'SuperUser', 'StandardUser' }
- isExternalAuth type *boolean*
- isActive type *boolean*
- displayName - optional; type *string*
- firstName - optional; type *string*
- middleName - optional; type *string*
- lastName - optional; type *string*
- companyName - optional; type *string*
- preferAddrType type *AddressType* - type *string* with restriction - enum { 'Fax', 'Email', 'FaxRaw' }
- email - optional; type *string*
- fax - optional; type *string*
- rawFax - optional; type *string*
- desc - optional; type *string*

Output: CreateNewUser1SoapOut (soap:body, use = literal) [Source code](#)

parameters type *CreateNewUser1Response*

- CreateNewUser1Result - optional; type *string*

8. CreateUser

[Source code](#)

Description: Create new user with NOR/NOS values

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/CreateNewUser1

Input: CreateNewUser2SoapIn (soap:body, use = literal) [Source code](#)

parameters type *CreateNewUser2*

- userID type *long*
- domainName - optional; type *string*
- userName - optional; type *string*
- password - optional; type *string*
- role type *Role* - type *string* with restriction - enum { 'DomainAdmin', 'SuperUser', 'StandardUser' }
- isExternalAuth type *boolean*
- isActive type *boolean*
- displayName - optional; type *string*
- firstName - optional; type *string*
- middleName - optional; type *string*
- lastName - optional; type *string*
- companyName - optional; type *string*
- preferAddrType type *AddressType* - type *string* with restriction - enum { 'Fax', 'Email', 'FaxRaw' }
- email - optional; type *string*
- fax - optional; type *string*
- rawFax - optional; type *string*
- NOR type *boolean*
- NOS type *boolean*
- desc - optional; type *string*

Output: CreateNewUser2SoapOut (soap:body, use = literal) [Source code](#)

parameters type *CreateNewUser2Response*

- CreateNewUser2Result - optional; type *string*

9. CreateUser

[Source code](#)

Description: Create new user with NOR/NOS values, callerID and CSID

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/CreateNewUser1

Input: CreateNewUser3SoapIn (soap:body, use = literal) [Source code](#)

```
parameters type CreateNewUser3
  ■ userID type long
  ■ domainName - optional; type string
  ■ userName - optional; type string
  ■ password - optional; type string
  ■ role type Role - type string with restriction - enum { 'DomainAdmin', 'SuperUser', 'StandardUser' }
  ■ isExternalAuth type boolean
  ■ isActive type boolean
  ■ displayName - optional; type string
  ■ firstName - optional; type string
  ■ middleName - optional; type string
  ■ lastName - optional; type string
  ■ companyName - optional; type string
  ■ preferAddrType type AddressType - type string with restriction - enum { 'Fax', 'Email', 'FaxRaw' }
  ■ email - optional; type string
  ■ fax - optional; type string
  ■ rawFax - optional; type string
  ■ NOR type boolean
  ■ NOS type boolean
  ■ desc - optional; type string
  ■ callerID - optional; type string
  ■ csid - optional; type string
```

Output: CreateNewUser3SoapOut (soap:body, use = literal) [Source code](#)

```
parameters type CreateNewUser3Response
  ■ CreateNewUser3Result - optional; type string
```

10. DeactivateUser

[Source code](#)

Description: Set user to be inactive

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/DeactivateUser

Input: DeactivateUserSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type DeactivateUser
  ■ adminID type long
  ■ userID type long
```

Output: DeactivateUserSoapOut (soap:body, use = literal) [Source code](#)

```
parameters type DeactivateUserResponse
  ■ DeactivateUserResult - optional; type string
```

11. DeactivateUsers

[Source code](#)

Description: Set a list of users to be inactive

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/DeactivateUsers

Input: DeactivateUsersSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type DeactivateUsers
  ■ adminID type long
  ■ userIDList - optional; type ArrayOfLong
```

- long - optional, unbounded; type *long*

Output: DeactivateUsersSoapOut (soap:body, use = literal) [Source code](#)

parameters type *DeactivateUsersResponse*

- DeactivateUsersResult - optional; type *string*

12. DelegateMessage

[Source code](#)

Description: Send message on behalf of other user

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/DelegateMessage

Input: DelegateMessageSoapIn (soap:body, use = literal) [Source code](#)

parameters type *DelegateMessage*

- delegateUserID type *long*
- userName - optional; type *string*
- senderName - optional; type *string*
- senderCompName - optional; type *string*
- subject - optional; type *string*
- note - optional; type *string*
- billingCode - optional; type *string*
- scheduleDate - optional; type *string*
- priority type *int*
- isOnHold type *boolean*
- mss type *boolean*
- msf type *boolean*
- trackLabel - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- trackValue - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- recpName - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- recpCompName - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- recpAddress - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- recpRawFax - optional; type *ArrayOfBoolean*
 - boolean - optional, unbounded; type *boolean*
- recpNotifyAddr - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- recplsRRFOn - optional; type *ArrayOfBoolean*
 - boolean - optional, unbounded; type *boolean*
- recplsRRSOn - optional; type *ArrayOfBoolean*
 - boolean - optional, unbounded; type *boolean*
- documentName - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- documentPath - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- documentBytes - optional; type *ArrayOfAnyType*
 - anyType - optional, unbounded, nillable;
- documentsMerge - optional; type *ArrayOfBoolean*
 - boolean - optional, unbounded; type *boolean*

Output: DelegateMessageSoapOut (soap:body, use = literal) [Source code](#)

parameters type *DelegateMessageResponse*

- DelegateMessageResult - optional; type *string*

13. DeleteMessage

[Source code](#)

Description: Deletes message.

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/DeleteMessage>

Input: DeleteMessageSoapIn (soap:body, use = literal) [Source code](#)

parameters type *DeleteMessage*

- msgID - optional; type *string*
- userID type *long*

Output: DeleteMessageSoapOut (soap:body, use = literal) [Source code](#)

parameters type *DeleteMessageResponse*

- DeleteMessageResult - optional; type *string*

14. [DownloadMessageByte](#)

[Source code](#)

Description: Download message's binary with specified message id, delivery number, download type and start page index

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/DownloadMessageByte1>

Input: DownloadMessageByte1SoapIn (soap:body, use = literal) [Source code](#)

parameters type *DownloadMessageByte1*

- messageID - optional; type *string*
- deliveryNum type *int*
- downloadType - optional; type *string*
- startPageIndex type *int*

Output: DownloadMessageByte1SoapOut (soap:body, use = literal) [Source code](#)

parameters type *DownloadMessageByte1Response*

- DownloadMessageByte1Result - optional; type *string*
- output - optional; type *base64Binary*

15. [DownloadMessageByte](#)

[Source code](#)

Description: Download message's binary with specified message id, delivery number, download type, start page index and list of excluded pages

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/DownloadMessageByte1>

Input: DownloadMessageInByte2SoapIn (soap:body, use = literal) [Source code](#)

parameters type *DownloadMessageInByte2*

- messageID - optional; type *string*
- deliveryNum type *int*
- downloadType - optional; type *string*
- startPageIndex type *int*
- excludePageList - optional; type *ArrayOfInt*
 - int - optional, unbounded; type *int*

Output: DownloadMessageInByte2SoapOut (soap:body, use = literal) [Source code](#)

parameters type *DownloadMessageInByte2Response*

- DownloadMessageInByte2Result - optional; type *string*
- output - optional; type *base64Binary*

16. [DownloadMessageByte](#)

[Source code](#)

Description:Download message's binary with specified delivery ID, download type and start page index

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/DownloadMessageByte1>

Input: DownloadMessageInByte3SoapIn (soap:body, use = literal) [Source code](#)

parameters type *DownloadMessageInByte3*

- deliveryID - optional; type *string*
- downloadType - optional; type *string*
- startPageIndex type *int*

Output: DownloadMessageInByte3SoapOut (soap:body, use = literal) [Source code](#)

parameters type *DownloadMessageInByte3Response*

- DownloadMessageInByte3Result - optional; type *string*
- output - optional; type *base64Binary*

17. DownloadMessageByte

[Source code](#)

Description: Download message's binary with specified delivery id, download type, start page index and list of excluded pages

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/DownloadMessageByte1>

Input: DownloadMessageInByte4SoapIn (soap:body, use = literal) [Source code](#)

parameters type *DownloadMessageInByte4*

- deliveryID - optional; type *string*
- downloadType - optional; type *string*
- startPageIndex type *int*
- excludePageList - optional; type *ArrayOfInt*
 - int - optional, unbounded; type *int*

Output: DownloadMessageInByte4SoapOut (soap:body, use = literal) [Source code](#)

parameters type *DownloadMessageInByte4Response*

- DownloadMessageInByte4Result - optional; type *string*
- output - optional; type *base64Binary*

18. ForwardToUser

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/ForwardToUser>

Input: ForwardToUserSoapIn (soap:body, use = literal) [Source code](#)

parameters type *ForwardToUser*

- userID - optional; type *string*
- msgID - optional; type *string*
- usernames - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*

Output: ForwardToUserSoapOut (soap:body, use = literal) [Source code](#)

parameters type *ForwardToUserResponse*

- ForwardToUserResult - optional; type *string*

19. GetDomainList

[Source code](#)

document

Style:

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/GetDomainList>

Input: GetDomainListSoapIn (soap:body, use = literal) [Source code](#)

parameters type *GetDomainList*

- adminID type *long*

Output: GetDomainListSoapOut (soap:body, use = literal) [Source code](#)

parameters type *GetDomainListResponse*

- GetDomainListResult - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*

20. GetMessageDetails

[Source code](#)

Description: Gets message details in xml string

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/GetMessageDetails>

Input: GetMessageDetailsSoapIn (soap:body, use = literal) [Source code](#)

parameters type *GetMessageDetails*

- msgID - optional; type *string*

Output: GetMessageDetailsSoapOut (soap:body, use = literal) [Source code](#)

parameters type *GetMessageDetailsResponse*

- GetMessageDetailsResult - optional; type *string*

21. GetMessageList

[Source code](#)

Description: Get message list based on folder name and isread/unread.

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/GetMessageList3>

Input: GetMessageList3SoapIn (soap:body, use = literal) [Source code](#)

parameters type *GetMessageList3*

- userID type *long*
- folderName - optional; type *string*
- isRead type *boolean*
- sortDescending type *boolean*
- maxResults type *int*

Output: GetMessageList3SoapOut (soap:body, use = literal) [Source code](#)

parameters type *GetMessageList3Response*

- GetMessageList3Result - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*

22. GetMessageList

[Source code](#)

Description: Get message list based on folder name, created date range and is read/unread.

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/GetMessageList3>

Input: GetMessageList2SoapIn (soap:body, use = literal) [Source code](#)

parameters type *GetMessageList2*

- userID type *long*
- folderName - optional; type *string*
- isRead type *boolean*
- startDate - optional; type *string*
- endDate - optional; type *string*
- sortDescending type *boolean*
- maxResults type *int*

Output: GetMessageList2SoapOut (soap:body, use = literal) [Source code](#)

- parameters** type *GetMessageList2Response*
- GetMessageList2Result - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*

23. GetMessageList

[Source code](#)

Description: Get message list based on folder name and created date range.

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/GetMessageList3

Input: GetMessageList1SoapIn (soap:body, use = literal) [Source code](#)

- parameters** type *GetMessageList1*
- userID type *long*
 - folderName - optional; type *string*
 - startDate - optional; type *string*
 - endDate - optional; type *string*
 - sortDescending type *boolean*
 - maxResults type *int*

Output: GetMessageList1SoapOut (soap:body, use = literal) [Source code](#)

- parameters** type *GetMessageList1Response*
- GetMessageList1Result - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*

24. GetMessageList

[Source code](#)

Description: Get message list based on folder name.

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/GetMessageList3

Input: GetMessageListSoapIn (soap:body, use = literal) [Source code](#)

- parameters** type *GetMessageList*
- userID type *long*
 - folderName - optional; type *string*
 - sortDescending type *boolean*
 - maxResults type *int*

Output: GetMessageListSoapOut (soap:body, use = literal) [Source code](#)

- parameters** type *GetMessageListResponse*
- GetMessageListResult - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*

25. GetMessageList

[Source code](#)

Description: Get message list based on list of users, folder name and whether is downloaded externally

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/GetMessageList3>

Input: GetMessageList4SoapIn (soap:body, use = literal) [Source code](#)

```
parameters type GetMessageList4
  ■ userList - optional; typeArrayOfLong
    ■ long - optional, unbounded; type long
  ■ folderName - optional; type string
  ■ isDownloaded type boolean
  ■ sortDescending type boolean
  ■ maxResults type int
```

Output: GetMessageList4SoapOut (soap:body, use = literal) [Source code](#)

```
parameters type GetMessageList4Response
  ■ GetMessageList4Result - optional; typeArrayOfString
    ■ string - optional, unbounded, nillable; type string
```

26. GetMessageList

[Source code](#)

Description: Get message list based on list of users, folder name, whether is downloaded externally, is read, start and end date

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/GetMessageList3>

Input: GetMessageList5SoapIn (soap:body, use = literal) [Source code](#)

```
parameters type GetMessageList5
  ■ userList - optional; typeArrayOfLong
    ■ long - optional, unbounded; type long
  ■ folderName - optional; type string
  ■ isDownloaded type boolean
  ■ isRead type boolean
  ■ startDate - optional; type string
  ■ endDate - optional; type string
  ■ sortDescending type boolean
  ■ maxResults type int
```

Output: GetMessageList5SoapOut (soap:body, use = literal) [Source code](#)

```
parameters type GetMessageList5Response
  ■ GetMessageList5Result - optional; typeArrayOfString
    ■ string - optional, unbounded, nillable; type string
```

27. GetMessageList

[Source code](#)

Description: Get message list based on list of users, folder name, whether is downloaded externally, start and end date

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/GetMessageList3>

Input: GetMessageList6SoapIn (soap:body, use = literal) [Source code](#)

```
parameters type GetMessageList6
  ■ userList - optional; typeArrayOfLong
    ■ long - optional, unbounded; type long
  ■ folderName - optional; type string
  ■ isDownloaded type boolean
  ■ startDate - optional; type string
  ■ endDate - optional; type string
  ■ sortDescending type boolean
  ■ maxResults type int
```

GetMessageList6SoapOut (soap:body, use = literal) [Source code](#)

Output:

```
parameters type GetMessageList6Response
  ■ GetMessageList6Result - optional; typeArrayOfString
    ■ string - optional, unbounded, nillable; type string
```

28. GetMessageList

[Source code](#)

Description: Get message list based on list of users, folder name, whether is downloaded externally, and is read

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/GetMessageList3

Input: GetMessageList7SoapIn (soap:body, use = literal) [Source code](#)

```
parameters type GetMessageList7
  ■ userList - optional; typeArrayOfLong
    ■ long - optional, unbounded; type long
  ■ folderName - optional; type string
  ■ isDownloaded type boolean
  ■ isRead type boolean
  ■ sortDescending type boolean
  ■ maxResults type int
```

Output: GetMessageList7SoapOut (soap:body, use = literal) [Source code](#)

```
parameters type GetMessageList7Response
  ■ GetMessageList7Result - optional; typeArrayOfString
    ■ string - optional, unbounded, nillable; type string
```

29. GetMessageStatus

[Source code](#)

Description: Get message's status.

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/GetMessageStatus

Input: GetMessageStatusSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type GetMessageStatus
  ■ msgID - optional; type string
```

Output: GetMessageStatusSoapOut (soap:body, use = literal) [Source code](#)

```
parameters type GetMessageStatusResponse
  ■ GetMessageStatusResult - optional; type string
```

30. GetMsgFolders

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/GetMsgFolders

Input: GetMsgFoldersSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type GetMsgFolders
  ■ userID type long
```

Output: GetMsgFoldersSoapOut (soap:body, use = literal) [Source code](#)

```
parameters type GetMsgFoldersResponse
  ■ GetMsgFoldersResult - optional; typeArrayOfString
    ■ string - optional, unbounded, nillable; type string
```

31. GetUserDetails

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/GetUserDetailsByUserID

Input: GetUserDetailsByUserIDSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type GetUserDetailsByUserID
  ■ adminID type long
  ■ userID type long
```

Output: GetUserDetailsByUserIDSoapOut (soap:body, use = literal) [Source code](#)

```
parameters type GetUserDetailsByUserIDResponse
  ■ GetUserDetailsByUserIDResult - optional; type string
```

32. GetUserDetails

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/GetUserDetailsByUserID

Input: GetUserDetailsByUserNameSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type GetUserDetailsByUserName
  ■ adminID type long
  ■ userName - optional; type string
```

Output: GetUserDetailsByUserNameSoapOut (soap:body, use = literal) [Source code](#)

```
parameters type GetUserDetailsByUserNameResponse
  ■ GetUserDetailsByUserNameResult - optional; type string
```

33. GetUserFaxSetting

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/GetUserFaxSettingByUserID

Input: GetUserFaxSettingByUserIDSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type GetUserFaxSettingByUserID
  ■ adminID type long
  ■ userID type long
```

Output: GetUserFaxSettingByUserIDSoapOut (soap:body, use = literal) [Source code](#)

```
parameters type GetUserFaxSettingByUserIDResponse
  ■ GetUserFaxSettingByUserIDResult - optional; type string
```

34. GetUserFaxSetting

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/GetUserFaxSettingByUserID

Input: GetUserFaxSettingByUserNameSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type GetUserFaxSettingByUserName
  ■ adminID type long
  ■ userName - optional; type string
```

Output: GetUserFaxSettingByUserNameSoapOut (soap:body, use = literal) [Source code](#)

parameters type *GetUserFaxSettingByUserNameResponse*

- *GetUserFaxSettingByUserNameResult* - optional; type *string*

35. GetUserList

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/GetUserListWithDomainID>

Input: *GetUserListWithDomainIDSoapIn* (soap:body, use = literal) [Source code](#)

parameters type *GetUserListWithDomainID*

- *adminID* type *long*
- *domainID* type *long*

Output: *GetUserListWithDomainIDSoapOut* (soap:body, use = literal) [Source code](#)

parameters type *GetUserListWithDomainIDResponse*

- *GetUserListWithDomainIDResult* - optional; type *ArrayOfString*
 - *string* - optional, unbounded, nillable; type *string*

36. GetUserList

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/GetUserListWithDomainID>

Input: *GetUserListWithDomainNameSoapIn* (soap:body, use = literal) [Source code](#)

parameters type *GetUserListWithDomainName*

- *adminID* type *long*
- *domainName* - optional; type *string*

Output: *GetUserListWithDomainNameSoapOut* (soap:body, use = literal) [Source code](#)

parameters type *GetUserListWithDomainNameResponse*

- *GetUserListWithDomainNameResult* - optional; type *ArrayOfString*
 - *string* - optional, unbounded, nillable; type *string*

37. IsMessageDeleted

[Source code](#)

Description: To check whether the message is being deleted

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/IsMessageDeleted>

Input: *IsMessageDeletedSoapIn* (soap:body, use = literal) [Source code](#)

parameters type *IsMessageDeleted*

- *messageID* - optional; type *string*
- *userID* - optional; type *string*

Output: *IsMessageDeletedSoapOut* (soap:body, use = literal) [Source code](#)

parameters type *IsMessageDeletedResponse*

- *IsMessageDeletedResult* - optional; type *string*

38. IsMessageRead

[Source code](#)

Style: document

Request-response. The endpoint receives a message, and sends a correlated message.

Operation type:

SOAP action: http://faxcoreserver/services/IsMessageRead

Input: IsMessageReadSoapIn (soap:body, use = literal) [Source code](#)

parameters type *IsMessageRead*

- userID type *long*
- msgID - optional; type *string*

Output: IsMessageReadSoapOut (soap:body, use = literal) [Source code](#)

parameters type *IsMessageReadResponse*

- IsMessageReadResult type *boolean*

39. [MarkMsgAsDownloaded](#)

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/MarkMsgAsDownloaded

Input: MarkMsgAsDownloadedSoapIn (soap:body, use = literal) [Source code](#)

parameters type *MarkMsgAsDownloaded*

- userID type *long*
- msgID - optional; type *string*

Output: MarkMsgAsDownloadedSoapOut (soap:body, use = literal) [Source code](#)

parameters type *MarkMsgAsDownloadedResponse*

- MarkMsgAsDownloadedResult - optional; type *string*

40. [MarkMsgAsRead](#)

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/MarkMsgAsRead

Input: MarkMsgAsReadSoapIn (soap:body, use = literal) [Source code](#)

parameters type *MarkMsgAsRead*

- userID type *long*
- msgID - optional; type *string*

Output: MarkMsgAsReadSoapOut (soap:body, use = literal) [Source code](#)

parameters type *MarkMsgAsReadResponse*

- MarkMsgAsReadResult - optional; type *string*

41. [MarkMsgAsUnread](#)

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/MarkMsgAsUnread

Input: MarkMsgAsUnreadSoapIn (soap:body, use = literal) [Source code](#)

parameters type *MarkMsgAsUnread*

- userID type *long*
- msgID - optional; type *string*

Output: MarkMsgAsUnreadSoapOut (soap:body, use = literal) [Source code](#)

parameters type *MarkMsgAsUnreadResponse*

- MarkMsgAsUnreadResult - optional; type *string*

42. RemoveUser

[Source code](#)

Description: Remove user

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: `http://faxcoreserver/services/RemoveUser`

Input: RemoveUserSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type RemoveUser
  ■ adminID type long
  ■ userID type long
```

Output: RemoveUserSoapOut (soap:body, use = literal) [Source code](#)

```
parameters type RemoveUserResponse
  ■ RemoveUserResult - optional; type string
```

43. RetryFailed

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: `http://faxcoreserver/services/RetryFailed`

Input: RetryFailedSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type RetryFailed
  ■ msgID - optional; type string
```

Output: RetryFailedSoapOut (soap:body, use = literal) [Source code](#)

```
parameters type RetryFailedResponse
  ■ RetryFailedResult - optional; type string
```

44. SearchIBRouteRules

[Source code](#)

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: `http://faxcoreserver/services/SearchIBRouteRules`

Input: SearchIBRouteRulesSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type SearchIBRouteRules
  ■ adminID type long
  ■ input - optional; type string
```

Output: SearchIBRouteRulesSoapOut (soap:body, use = literal) [Source code](#)

```
parameters type SearchIBRouteRulesResponse
  ■ SearchIBRouteRulesResult - optional; type string
```

45. SendMessage

[Source code](#)

Description: Send message to multiple recipients with multiple documents

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: `http://faxcoreserver/services/SendCompleteMessage`

Input: SendCompleteMessageSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type SendCompleteMessage
  ■ userID type long
  ■ senderName - optional; type string
  ■ senderCompName - optional; type string
```

- subject - optional; type *string*
- note - optional; type *string*
- billingCode - optional; type *string*
- scheduleDate - optional; type *string*
- priority type *int*
- isOnHold type *boolean*
- mss type *boolean*
- msf type *boolean*
- trackLabel - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- trackValue - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- recpName - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- recpCompName - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- recpAddress - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- recpRawFax - optional; type *ArrayOfBoolean*
 - boolean - optional, unbounded; type *boolean*
- recpNotifyAddr - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- recplsRRFOn - optional; type *ArrayOfBoolean*
 - boolean - optional, unbounded; type *boolean*
- recplsRRSOn - optional; type *ArrayOfBoolean*
 - boolean - optional, unbounded; type *boolean*
- documentName - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- documentPath - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
- documentBytes - optional; type *ArrayOfAnyType*
 - anyType - optional, unbounded, nillable;
- documentsMerge - optional; type *ArrayOfBoolean*
 - boolean - optional, unbounded; type *boolean*

Output: SendCompleteMessageSoapOut (soap:body, use = literal) [Source code](#)

- parameters** type *SendCompleteMessageResponse*
- SendCompleteMessageResult - optional; type *string*

46. SendMessage

[Source code](#)

Description: Basic send message with documents from bytes

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/SendCompleteMessage

Input: BasicSendMessageWithBytesSoapIn (soap:body, use = literal) [Source code](#)

- parameters** type *BasicSendMessageWithBytes*
- userID type *long*
 - senderName - optional; type *string*
 - senderCompName - optional; type *string*
 - subject - optional; type *string*
 - note - optional; type *string*
 - trackLabel - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
 - trackValue - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
 - recpName - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
 - recpAddress - optional; type *ArrayOfString*
 - string - optional, unbounded, nillable; type *string*
 - recpRawFax - optional; type *ArrayOfBoolean*

- boolean - optional, unbounded; type *boolean*
- *documentName* - optional; type *ArrayOfString*
 - *string* - optional, unbounded, nillable; type *string*
- *documentBytes* - optional; type *ArrayOfAnyType*
 - *anyType* - optional, unbounded, nillable;
- *documentsMerge* - optional; type *ArrayOfBoolean*
 - *boolean* - optional, unbounded; type *boolean*

Output: *BasicSendMessageWithBytesSoapOut* (soap:body, use = literal) [Source code](#)

- parameters** type *BasicSendMessageWithBytesResponse*
- *BasicSendMessageWithBytesResult* - optional; type *string*

47. **SendMessage**

[Source code](#)

Description: Send message to multiple recipients with multiple documents

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/SendCompleteMessage>

Input: *SendCompleteMessageWithPresetAgentsSoapIn* (soap:body, use = literal) [Source code](#)

- parameters** type *SendCompleteMessageWithPresetAgents*
- *userID* type *long*
 - *senderName* - optional; type *string*
 - *senderCompName* - optional; type *string*
 - *subject* - optional; type *string*
 - *note* - optional; type *string*
 - *billingCode* - optional; type *string*
 - *scheduleDate* - optional; type *string*
 - *priority* type *int*
 - *isOnHold* type *boolean*
 - *mss* type *boolean*
 - *msf* type *boolean*
 - *trackLabel* - optional; type *ArrayOfString*
 - *string* - optional, unbounded, nillable; type *string*
 - *trackValue* - optional; type *ArrayOfString*
 - *string* - optional, unbounded, nillable; type *string*
 - *recpName* - optional; type *ArrayOfString*
 - *string* - optional, unbounded, nillable; type *string*
 - *recpCompName* - optional; type *ArrayOfString*
 - *string* - optional, unbounded, nillable; type *string*
 - *recpAddress* - optional; type *ArrayOfString*
 - *string* - optional, unbounded, nillable; type *string*
 - *recpRawFax* - optional; type *ArrayOfBoolean*
 - *boolean* - optional, unbounded; type *boolean*
 - *recpNotifyAddr* - optional; type *ArrayOfString*
 - *string* - optional, unbounded, nillable; type *string*
 - *recplsRRFOn* - optional; type *ArrayOfBoolean*
 - *boolean* - optional, unbounded; type *boolean*
 - *recplsRRSON* - optional; type *ArrayOfBoolean*
 - *boolean* - optional, unbounded; type *boolean*
 - *documentName* - optional; type *ArrayOfString*
 - *string* - optional, unbounded, nillable; type *string*
 - *documentPath* - optional; type *ArrayOfString*
 - *string* - optional, unbounded, nillable; type *string*
 - *documentBytes* - optional; type *ArrayOfAnyType*
 - *anyType* - optional, unbounded, nillable;
 - *documentsMerge* - optional; type *ArrayOfBoolean*
 - *boolean* - optional, unbounded; type *boolean*
 - *agentID* - optional; type *ArrayOfInt*
 - *int* - optional, unbounded; type *int*
 - *presetType* - optional; type *ArrayOfInt*
 - *int* - optional, unbounded; type *int*

- presetValue - optional; type *ArrayOfInt*
 - int - optional, unbounded; type *int*

Output: `SendCompleteMessageWithPresetAgentsSoapOut` (soap:body, use = literal) [Source code](#)

- parameters** type *SendCompleteMessageWithPresetAgentsResponse*
- `SendCompleteMessageWithPresetAgentsResult` - optional; type *string*

48. `SendMessage`

[Source code](#)

Description: Send message to multiple recipients with multiple documents

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: `http://faxcoreserver/services/SendCompleteMessage`

Input: `SendCompleteMessageWithPresetAgentsAndByPassRenderSoapIn` (soap:body, use = literal) [Source code](#)

- parameters** type *SendCompleteMessageWithPresetAgentsAndByPassRender*
- `userID` type *long*
 - `senderName` - optional; type *string*
 - `senderCompName` - optional; type *string*
 - `subject` - optional; type *string*
 - `note` - optional; type *string*
 - `billingCode` - optional; type *string*
 - `scheduleDate` - optional; type *string*
 - `priority` type *int*
 - `isOnHold` type *boolean*
 - `mss` type *boolean*
 - `msf` type *boolean*
 - `trackLabel` - optional; type *ArrayOfString*
 - `string` - optional, unbounded, nillable; type *string*
 - `trackValue` - optional; type *ArrayOfString*
 - `string` - optional, unbounded, nillable; type *string*
 - `recpName` - optional; type *ArrayOfString*
 - `string` - optional, unbounded, nillable; type *string*
 - `recpCompName` - optional; type *ArrayOfString*
 - `string` - optional, unbounded, nillable; type *string*
 - `recpAddress` - optional; type *ArrayOfString*
 - `string` - optional, unbounded, nillable; type *string*
 - `recpRawFax` - optional; type *ArrayOfBoolean*
 - `boolean` - optional, unbounded; type *boolean*
 - `recpNotifyAddr` - optional; type *ArrayOfString*
 - `string` - optional, unbounded, nillable; type *string*
 - `recplsRRFOn` - optional; type *ArrayOfBoolean*
 - `boolean` - optional, unbounded; type *boolean*
 - `recplsRRSON` - optional; type *ArrayOfBoolean*
 - `boolean` - optional, unbounded; type *boolean*
 - `documentName` - optional; type *ArrayOfString*
 - `string` - optional, unbounded, nillable; type *string*
 - `documentPath` - optional; type *ArrayOfString*
 - `string` - optional, unbounded, nillable; type *string*
 - `documentBytes` - optional; type *ArrayOfAnyType*
 - `anyType` - optional, unbounded, nillable;
 - `documentsMerge` - optional; type *ArrayOfBoolean*
 - `boolean` - optional, unbounded; type *boolean*
 - `agentID` - optional; type *ArrayOfInt*
 - `int` - optional, unbounded; type *int*
 - `presetType` - optional; type *ArrayOfInt*
 - `int` - optional, unbounded; type *int*
 - `presetValue` - optional; type *ArrayOfInt*
 - `int` - optional, unbounded; type *int*
 - `byPassRender` type *boolean*

Output: `SendCompleteMessageWithPresetAgentsAndByPassRenderSoapOut` (soap:body, use = literal) [Source code](#)

parameters type *SendCompleteMessageWithPresetAgentsAndByPassRenderResponse*

- *SendCompleteMessageWithPresetAgentsAndByPassRenderResult* - optional; type *string*

49. **SendMixedMode**

[Source code](#)

Description: Send message in mixed mode.

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/SendMixedMode>

Input: *SendMixedModeSoapIn* (soap:body, use = literal) [Source code](#)

parameters type *SendMixedMode*

- *userID* - optional; type *string*
- *senderName* - optional; type *string*
- *senderCompName* - optional; type *string*
- *subject* - optional; type *string*
- *note* - optional; type *string*
- *billingCode* - optional; type *string*
- *priority* type *int*
- *scheduleDate* - optional; type *string*
- *isOnHold* type *boolean*
- *mss* type *boolean*
- *msf* type *boolean*
- *useDefaultNotify* type *boolean*
- *useDefaultCP* type *boolean*
- *documents* - optional; type *ArrayOfMsgDocument*
 - *MsgDocument* - optional, unbounded, nillable; type *MsgDocument*
 - *Name* - optional; type *string*
 - *Path* - optional; type *string*
 - *Bytes* - optional; type *base64Binary*
 - *IsMerge* type *boolean*
 - *RemoveFirstPage* type *boolean*
- *recipients* - optional; type *ArrayOfMsgRecipient*
 - *MsgRecipient* - optional, unbounded, nillable; type *MsgRecipient*
 - *Name* - optional; type *string*
 - *CompName* - optional; type *string*
 - *Address* - optional; type *string*
 - *AddressType* type *int*
 - *NotifyAddress* - optional; type *string*
 - *IsRRFOn* type *boolean*
 - *IsRRSON* type *boolean*
- *trackings* - optional; type *ArrayOfMsgTracking*
 - *MsgTracking* - optional, unbounded, nillable; type *MsgTracking*
 - *TrackLabel* - optional; type *string*
 - *TrackValue* - optional; type *string*

Output: *SendMixedModeSoapOut* (soap:body, use = literal) [Source code](#)

parameters type *SendMixedModeResponse*

- *SendMixedModeResult* - optional; type *string*

50. **ServerStatus**

[Source code](#)

Description: Check server status

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: <http://faxcoreserver/services/ServerStatus>

Input: *ServerStatusSoapIn* (soap:body, use = literal) [Source code](#)

parameters type *ServerStatus*

Output: ServerStatusSoapOut (soap:body, use = literal) [Source code](#)

parameters type *ServerStatusResponse*

- ServerStatusResult - optional; type *string*

51. UpdateUser

[Source code](#)

Description: Update user with basic profile

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/UpdateUser-basic

Input: UpdateUser-basicSoapIn (soap:body, use = literal) [Source code](#)

parameters type *UpdateUser-basic*

- adminID type *long*
- userID type *long*
- role type *Role* - type *string* with restriction - enum { 'DomainAdmin', 'SuperUser', 'StandardUser' }
- isExternalAuth type *boolean*
- isActive type *boolean*
- displayName - optional; type *string*
- firstName - optional; type *string*
- middleName - optional; type *string*
- lastName - optional; type *string*
- companyName - optional; type *string*
- preferAddrType type *AddressType* - type *string* with restriction - enum { 'Fax', 'Email', 'FaxRaw' }
- desc - optional; type *string*

Output: UpdateUser-basicSoapOut (soap:body, use = literal) [Source code](#)

parameters type *UpdateUser-basicResponse*

- UpdateUser-basicResult - optional; type *string*

52. UpdateUser

[Source code](#)

Description: Update user with basic profile, local CSID, and caller ID

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/UpdateUser-basic

Input: UpdateUser-extraSoapIn (soap:body, use = literal) [Source code](#)

parameters type *UpdateUser-extra*

- adminID type *long*
- userID type *long*
- role type *Role* - type *string* with restriction - enum { 'DomainAdmin', 'SuperUser', 'StandardUser' }
- isExternalAuth type *boolean*
- isActive type *boolean*
- displayName - optional; type *string*
- firstName - optional; type *string*
- middleName - optional; type *string*
- lastName - optional; type *string*
- companyName - optional; type *string*
- preferAddrType type *AddressType* - type *string* with restriction - enum { 'Fax', 'Email', 'FaxRaw' }
- desc - optional; type *string*
- localCSID - optional; type *string*
- callerID - optional; type *string*

Output: UpdateUser-extraSoapOut (soap:body, use = literal) [Source code](#)

parameters type *UpdateUser-extraResponse*

- UpdateUser-extraResult - optional; type *string*

53. UpdateUser

[Source code](#)

Description: Update user with basic profile, and user's config

Style: document

Operation type: *Request-response*. The endpoint receives a message, and sends a correlated message.

SOAP action: http://faxcoreserver/services/UpdateUser-basic

Input: UpdateUser-advanceSoapIn (soap:body, use = literal) [Source code](#)

```
parameters type UpdateUser-advance
  ■ adminID type long
  ■ userID type long
  ■ role type Role - type string with restriction - enum { 'DomainAdmin', 'SuperUser', 'StandardUser' }
  ■ isExternalAuth type boolean
  ■ isActive type boolean
  ■ displayName - optional; type string
  ■ firstName - optional; type string
  ■ middleName - optional; type string
  ■ lastName - optional; type string
  ■ companyName - optional; type string
  ■ preferAddrType type AddressType - type string with restriction - enum { 'Fax', 'Email', 'FaxRaw' }
  ■ desc - optional; type string
  ■ config - optional; type ArrayOfUserConfig
    ■ UserConfig - optional, unbounded, nillable; type UserConfig
      ■ ConfigName - optional; type string
      ■ ConfigValue - optional; type string
```

Output: UpdateUser-advanceSoapOut (soap:body, use = literal) [Source code](#)

```
parameters type UpdateUser-advanceResponse
  ■ UpdateUser-advanceResult - optional; type string
```

WSDL source code

```
<?xml version="1.0"?>
<wSDL:definitions targetNamespace="http://faxcoreserver/services/"
  xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
  xmlns:ttns="http://faxcoreserver/services/"
  xmlns:s="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
  xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
  xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
>
  <wSDL:types>
    <s:schema elementFormDefault="qualified" targetNamespace="http://faxcoreserver/services/">
      <s:element name="AuthenticateUser">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="userName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="password" type="s:string" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="AuthenticateUserResponse">
        <s:complexType>
```



```

        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="AuthenticateUserResult" type="s:long" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="ServerStatus">
    <s:complexType />
</s:element>
<s:element name="ServerStatusResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="ServerStatusResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="SendCompleteMessage">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
            <s:element minOccurs="0" maxOccurs="1" name="senderName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="senderCompName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="subject" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="note" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="billingCode" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="scheduleDate" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="priority" type="s:int" />
            <s:element minOccurs="1" maxOccurs="1" name="isOnHold" type="s:boolean" />
            <s:element minOccurs="1" maxOccurs="1" name="mss" type="s:boolean" />
            <s:element minOccurs="1" maxOccurs="1" name="msf" type="s:boolean" />
            <s:element minOccurs="0" maxOccurs="1" name="trackLabel" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="trackValue" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="recpName" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="recpCompName" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="recpAddress" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="recpRawFax" type="tns:ArrayOfBoolean" />
            <s:element minOccurs="0" maxOccurs="1" name="recpNotifyAddr" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="recplsRFRon" type="tns:ArrayOfBoolean" />
            <s:element minOccurs="0" maxOccurs="1" name="recplsRRSOn" type="tns:ArrayOfBoolean" />
            <s:element minOccurs="0" maxOccurs="1" name="documentName" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="documentPath" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="documentBytes" type="tns:ArrayOfAnyType" />
            <s:element minOccurs="0" maxOccurs="1" name="documentIsMerge" type="tns:ArrayOfBoolean" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:complexType name="ArrayOfString">
    <s:sequence>
        <s:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true" type="s:string" />
    </s:sequence>
</s:complexType>
<s:complexType name="ArrayOfBoolean">

```

```

    <s:sequence>
      <s:element minOccurs="0" maxOccurs="unbounded" name="boolean" type="s:boolean" />
    </s:sequence>
  </s:complexType>
<s:complexType name="ArrayOfAnyType">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="unbounded" name="anyType" nillable="true" />
  </s:sequence>
</s:complexType>
<s:element name="SendCompleteMessageResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="SendCompleteMessageResult" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="SendMixedMode">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="userID" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="senderName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="senderCompName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="subject" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="note" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="billingCode" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="priority" type="s:int" />
      <s:element minOccurs="0" maxOccurs="1" name="scheduleDate" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="isOnHold" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="mss" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="msf" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="useDefaultNotify" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="useDefaultCP" type="s:boolean" />
      <s:element minOccurs="0" maxOccurs="1" name="documents" type="tns:ArrayOfMsgDocument" />
      <s:element minOccurs="0" maxOccurs="1" name="recipients" type="tns:ArrayOfMsgRecipient" />
      <s:element minOccurs="0" maxOccurs="1" name="trackings" type="tns:ArrayOfMsgTracking" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:complexType name="ArrayOfMsgDocument">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="unbounded" name="MsgDocument" nillable="true"
      type="tns:MsgDocument" />
  </s:sequence>
</s:complexType>
<s:complexType name="MsgDocument">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="1" name="Name" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="Path" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="Bytes" type="s:base64Binary" />
    <s:element minOccurs="1" maxOccurs="1" name="IsMerge" type="s:boolean" />
    <s:element minOccurs="1" maxOccurs="1" name="RemoveFirstPage" type="s:boolean" />
  </s:sequence>

```

```

        </s:sequence>
    </s:complexType>
    <s:complexType name="ArrayOfMsgRecipient">
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="unbounded" name="MsgRecipient" nillable="true"
                type="tns:MsgRecipient" />
        </s:sequence>
    </s:complexType>
    <s:complexType name="MsgRecipient">
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="Name" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="CompName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="Address" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="AddressType" type="s:int" />
            <s:element minOccurs="0" maxOccurs="1" name="NotifyAddress" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="IsRRFOn" type="s:boolean" />
            <s:element minOccurs="1" maxOccurs="1" name="IsRRSON" type="s:boolean" />
        </s:sequence>
    </s:complexType>
    <s:complexType name="ArrayOfMsgTracking">
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="unbounded" name="MsgTracking" nillable="true" type="tns:MsgTracking"
                />
        </s:sequence>
    </s:complexType>
    <s:complexType name="MsgTracking">
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="TrackLabel" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="TrackValue" type="s:string" />
        </s:sequence>
    </s:complexType>
    <s:element name="SendMixedModeResponse">
        <s:complexType>
            <s:sequence>
                <s:element minOccurs="0" maxOccurs="1" name="SendMixedModeResult" type="s:string" />
            </s:sequence>
        </s:complexType>
    </s:element>
    <s:element name="BasicSendMessageWithBytes">
        <s:complexType>
            <s:sequence>
                <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
                <s:element minOccurs="0" maxOccurs="1" name="senderName" type="s:string" />
                <s:element minOccurs="0" maxOccurs="1" name="senderCompName" type="s:string" />
                <s:element minOccurs="0" maxOccurs="1" name="subject" type="s:string" />
                <s:element minOccurs="0" maxOccurs="1" name="note" type="s:string" />
                <s:element minOccurs="0" maxOccurs="1" name="trackLabel" type="tns:ArrayOfString" />
                <s:element minOccurs="0" maxOccurs="1" name="trackValue" type="tns:ArrayOfString" />
                <s:element minOccurs="0" maxOccurs="1" name="recpName" type="tns:ArrayOfString" />
                <s:element minOccurs="0" maxOccurs="1" name="recpAddress" type="tns:ArrayOfString" />
                <s:element minOccurs="0" maxOccurs="1" name="recpRawFax" type="tns:ArrayOfBoolean" />
                <s:element minOccurs="0" maxOccurs="1" name="documentName" type="tns:ArrayOfString" />
            </s:sequence>
        </s:complexType>
    </s:element>

```

```

        <s:element minOccurs="0" maxOccurs="1" name="documentBytes" type="tns:ArrayOfAnyType" />
        <s:element minOccurs="0" maxOccurs="1" name="documentIsMerge" type="tns:ArrayOfBoolean" />
    </s:sequence>
</s:complexType>
</s:element>
<s:element name="BasicSendMessageWithBytesResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="BasicSendMessageWithBytesResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="SendCompleteMessageWithPresetAgents">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
            <s:element minOccurs="0" maxOccurs="1" name="senderName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="senderCompName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="subject" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="note" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="billingCode" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="scheduleDate" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="priority" type="s:int" />
            <s:element minOccurs="1" maxOccurs="1" name="isOnHold" type="s:boolean" />
            <s:element minOccurs="1" maxOccurs="1" name="mss" type="s:boolean" />
            <s:element minOccurs="1" maxOccurs="1" name="msf" type="s:boolean" />
            <s:element minOccurs="0" maxOccurs="1" name="trackLabel" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="trackValue" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="recpName" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="recpCompName" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="recpAddress" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="recpRawFax" type="tns:ArrayOfBoolean" />
            <s:element minOccurs="0" maxOccurs="1" name="recpNotifyAddr" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="recplsRRFOn" type="tns:ArrayOfBoolean" />
            <s:element minOccurs="0" maxOccurs="1" name="recplsRRSOn" type="tns:ArrayOfBoolean" />
            <s:element minOccurs="0" maxOccurs="1" name="documentName" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="documentPath" type="tns:ArrayOfString" />
            <s:element minOccurs="0" maxOccurs="1" name="documentBytes" type="tns:ArrayOfAnyType" />
            <s:element minOccurs="0" maxOccurs="1" name="documentIsMerge" type="tns:ArrayOfBoolean" />
            <s:element minOccurs="0" maxOccurs="1" name="agentID" type="tns:ArrayOfInt" />
            <s:element minOccurs="0" maxOccurs="1" name="presetType" type="tns:ArrayOfInt" />
            <s:element minOccurs="0" maxOccurs="1" name="presetValue" type="tns:ArrayOfInt" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:complexType name="ArrayOfInt">
    <s:sequence>
        <s:element minOccurs="0" maxOccurs="unbounded" name="int" type="s:int" />
    </s:sequence>
</s:complexType>

```

```

<s:element name="SendCompleteMessageWithPresetAgentsResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="SendCompleteMessageWithPresetAgentsResult"
        type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="SendCompleteMessageWithPresetAgentsAndByPassRender">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
      <s:element minOccurs="0" maxOccurs="1" name="senderName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="senderCompName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="subject" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="note" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="billingCode" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="scheduleDate" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="priority" type="s:int" />
      <s:element minOccurs="1" maxOccurs="1" name="isOnHold" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="mss" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="msf" type="s:boolean" />
      <s:element minOccurs="0" maxOccurs="1" name="trackLabel" type="tns:ArrayOfString" />
      <s:element minOccurs="0" maxOccurs="1" name="trackValue" type="tns:ArrayOfString" />
      <s:element minOccurs="0" maxOccurs="1" name="recpName" type="tns:ArrayOfString" />
      <s:element minOccurs="0" maxOccurs="1" name="recpCompName" type="tns:ArrayOfString" />
      <s:element minOccurs="0" maxOccurs="1" name="recpAddress" type="tns:ArrayOfString" />
      <s:element minOccurs="0" maxOccurs="1" name="recpRawFax" type="tns:ArrayOfBoolean" />
      <s:element minOccurs="0" maxOccurs="1" name="recpNotifyAddr" type="tns:ArrayOfString" />
      <s:element minOccurs="0" maxOccurs="1" name="recplsRRFOn" type="tns:ArrayOfBoolean" />
      <s:element minOccurs="0" maxOccurs="1" name="recplsRRSOn" type="tns:ArrayOfBoolean" />
      <s:element minOccurs="0" maxOccurs="1" name="documentName" type="tns:ArrayOfString" />
      <s:element minOccurs="0" maxOccurs="1" name="documentPath" type="tns:ArrayOfString" />
      <s:element minOccurs="0" maxOccurs="1" name="documentBytes" type="tns:ArrayOfAnyType" />
      <s:element minOccurs="0" maxOccurs="1" name="documentIsMerge" type="tns:ArrayOfBoolean" />
      <s:element minOccurs="0" maxOccurs="1" name="agentID" type="tns:ArrayOfInt" />
      <s:element minOccurs="0" maxOccurs="1" name="presetType" type="tns:ArrayOfInt" />
      <s:element minOccurs="0" maxOccurs="1" name="presetValue" type="tns:ArrayOfInt" />
      <s:element minOccurs="1" maxOccurs="1" name="byPassRender" type="s:boolean" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="SendCompleteMessageWithPresetAgentsAndByPassRenderResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1"
        name="SendCompleteMessageWithPresetAgentsAndByPassRenderResult" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="DelegateMessage">

```

```

<s:complexType>
  <s:sequence>
    <s:element minOccurs="1" maxOccurs="1" name="delegateUserID" type="s:long" />
    <s:element minOccurs="0" maxOccurs="1" name="userName" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="senderName" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="senderCompName" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="subject" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="note" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="billingCode" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="scheduleDate" type="s:string" />
    <s:element minOccurs="1" maxOccurs="1" name="priority" type="s:int" />
    <s:element minOccurs="1" maxOccurs="1" name="isOnHold" type="s:boolean" />
    <s:element minOccurs="1" maxOccurs="1" name="mss" type="s:boolean" />
    <s:element minOccurs="1" maxOccurs="1" name="msf" type="s:boolean" />
    <s:element minOccurs="0" maxOccurs="1" name="trackLabel" type="tns:ArrayOfString" />
    <s:element minOccurs="0" maxOccurs="1" name="trackValue" type="tns:ArrayOfString" />
    <s:element minOccurs="0" maxOccurs="1" name="recpName" type="tns:ArrayOfString" />
    <s:element minOccurs="0" maxOccurs="1" name="recpCompName" type="tns:ArrayOfString" />
    <s:element minOccurs="0" maxOccurs="1" name="recpAddress" type="tns:ArrayOfString" />
    <s:element minOccurs="0" maxOccurs="1" name="recpRawFax" type="tns:ArrayOfBoolean" />
    <s:element minOccurs="0" maxOccurs="1" name="recpNotifyAddr" type="tns:ArrayOfString" />
    <s:element minOccurs="0" maxOccurs="1" name="recplsRRFOn" type="tns:ArrayOfBoolean" />
    <s:element minOccurs="0" maxOccurs="1" name="recplsRRSOn" type="tns:ArrayOfBoolean" />
    <s:element minOccurs="0" maxOccurs="1" name="documentName" type="tns:ArrayOfString" />
    <s:element minOccurs="0" maxOccurs="1" name="documentPath" type="tns:ArrayOfString" />
    <s:element minOccurs="0" maxOccurs="1" name="documentBytes" type="tns:ArrayOfAnyType" />
    <s:element minOccurs="0" maxOccurs="1" name="documentIsMerge" type="tns:ArrayOfBoolean" />
  </s:sequence>
</s:complexType>
</s:element>
<s:element name="DelegateMessageResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="DelegateMessageResult" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageDetails">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="msgID" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageDetailsResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetMessageDetailsResult" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>

```

```

<s:element name="GetMessageStatus">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="msgID" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageStatusResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetMessageStatusResult" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="IsMessageDeleted">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="messageID" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="userID" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="IsMessageDeletedResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="IsMessageDeletedResult" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="DeleteMessage">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="msgID" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="DeleteMessageResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="DeleteMessageResult" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="CancelMessage">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="msgID" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="CancelMessageResponse">
  <s:complexType>

```

```

        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="CancelMessageResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="RetryFailed">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="msgID" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="RetryFailedResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="RetryFailedResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="ForwardToUser">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="userID" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="msgID" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="userames" type="tns:ArrayOfString" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="ForwardToUserResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="ForwardToUserResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="AssignMessage">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="userID" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="msgID" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="userames" type="tns:ArrayOfString" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="AssignMessageResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="AssignMessageResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="GetMessageList3">

```



```

</s:sequence>
</s:complexType>
</s:element>
<s:element name="GetMessageList3Response">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetMessageList3Result" type="tns:ArrayOfString" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageList2">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
      <s:element minOccurs="0" maxOccurs="1" name="folderName" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="isRead" type="s:boolean" />
      <s:element minOccurs="0" maxOccurs="1" name="startDate" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="endDate" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="sortDescending" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="maxResults" type="s:int" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageList2Response">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetMessageList2Result" type="tns:ArrayOfString" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageList1">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
      <s:element minOccurs="0" maxOccurs="1" name="folderName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="startDate" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="endDate" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="sortDescending" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="maxResults" type="s:int" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageList1Response">
  <s:complexType>
    <s:sequence>

```

```

        <s:element minOccurs="0" maxOccurs="1" name="GetMessageList1Result" type="tns:ArrayOfString" />
    </s:sequence>
</s:complexType>
</s:element>
<s:element name="GetMessageList">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
            <s:element minOccurs="0" maxOccurs="1" name="folderName" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="sortDescending" type="s:boolean" />
            <s:element minOccurs="1" maxOccurs="1" name="maxResults" type="s:int" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="GetMessageListResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetMessageListResult" type="tns:ArrayOfString" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="CreateNewUser1">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
            <s:element minOccurs="0" maxOccurs="1" name="domainName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="userName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="password" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="role" type="tns:Role" />
            <s:element minOccurs="1" maxOccurs="1" name="isExternalAuth" type="s:boolean" />
            <s:element minOccurs="1" maxOccurs="1" name="isActive" type="s:boolean" />
            <s:element minOccurs="0" maxOccurs="1" name="displayName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="firstName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="middleName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="lastName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="companyName" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="preferAddrType" type="tns:AddressType" />
            <s:element minOccurs="0" maxOccurs="1" name="email" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="fax" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="rawFax" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="desc" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:simpleType name="Role">
    <s:restriction base="s:string">
        <s:enumeration value="DomainAdmin" />
        <s:enumeration value="SuperUser" />
        <s:enumeration value="StandardUser" />
    </s:restriction>
</s:simpleType>

```

```

<s:simpleType name="AddressType">
  <s:restriction base="s:string">
    <s:enumeration value="Fax" />
    <s:enumeration value="Email" />
    <s:enumeration value="FaxRaw" />
  </s:restriction>
</s:simpleType>
<s:element name="CreateNewUser1Response">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="CreateNewUser1Result" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="CreateNewUser2">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
      <s:element minOccurs="0" maxOccurs="1" name="domainName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="userName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="password" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="role" type="tns:Role" />
      <s:element minOccurs="1" maxOccurs="1" name="isExternalAuth" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="isActive" type="s:boolean" />
      <s:element minOccurs="0" maxOccurs="1" name="displayName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="firstName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="middleName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="lastName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="companyName" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="preferAddrType" type="tns:AddressType" />
      <s:element minOccurs="0" maxOccurs="1" name="email" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="fax" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="rawFax" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="NOR" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="NOS" type="s:boolean" />
      <s:element minOccurs="0" maxOccurs="1" name="desc" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="CreateNewUser2Response">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="CreateNewUser2Result" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="CreateNewUser3">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
      <s:element minOccurs="0" maxOccurs="1" name="domainName" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>

```

```

<s:element minOccurs="0" maxOccurs="1" name="userName" type="s:string" />
<s:element minOccurs="0" maxOccurs="1" name="password" type="s:string" />
<s:element minOccurs="1" maxOccurs="1" name="role" type="tns:Role" />
<s:element minOccurs="1" maxOccurs="1" name="isExternalAuth" type="s:boolean" />
<s:element minOccurs="1" maxOccurs="1" name="isActive" type="s:boolean" />
<s:element minOccurs="0" maxOccurs="1" name="displayName" type="s:string" />
<s:element minOccurs="0" maxOccurs="1" name="firstName" type="s:string" />
<s:element minOccurs="0" maxOccurs="1" name="middleName" type="s:string" />
<s:element minOccurs="0" maxOccurs="1" name="lastName" type="s:string" />
<s:element minOccurs="0" maxOccurs="1" name="companyName" type="s:string" />
<s:element minOccurs="1" maxOccurs="1" name="preferAddrType" type="tns:AddressType" />
<s:element minOccurs="0" maxOccurs="1" name="email" type="s:string" />
<s:element minOccurs="0" maxOccurs="1" name="fax" type="s:string" />
<s:element minOccurs="0" maxOccurs="1" name="rawFax" type="s:string" />
<s:element minOccurs="1" maxOccurs="1" name="NOR" type="s:boolean" />
<s:element minOccurs="1" maxOccurs="1" name="NOS" type="s:boolean" />
<s:element minOccurs="0" maxOccurs="1" name="desc" type="s:string" />
<s:element minOccurs="0" maxOccurs="1" name="callerID" type="s:string" />
<s:element minOccurs="0" maxOccurs="1" name="csid" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
<s:element name="CreateNewUser3Response">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="CreateNewUser3Result" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="UpdateUser-basic">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
      <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
      <s:element minOccurs="1" maxOccurs="1" name="role" type="tns:Role" />
      <s:element minOccurs="1" maxOccurs="1" name="isExternalAuth" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="isActive" type="s:boolean" />
      <s:element minOccurs="0" maxOccurs="1" name="displayName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="firstName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="middleName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="lastName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="companyName" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="preferAddrType" type="tns:AddressType" />
      <s:element minOccurs="0" maxOccurs="1" name="desc" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="UpdateUser-basicResponse">
  <s:complexType>
    <s:sequence>

```

```

        <s:element minOccurs="0" maxOccurs="1" name="UpdateUser-basicResult" type="s:string" />
    </s:sequence>
</s:complexType>
</s:element>
<s:element name="UpdateUser-extra">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
            <s:element minOccurs="1" maxOccurs="1" name="role" type="tns:Role" />
            <s:element minOccurs="1" maxOccurs="1" name="isExternalAuth" type="s:boolean" />
            <s:element minOccurs="1" maxOccurs="1" name="isActive" type="s:boolean" />
            <s:element minOccurs="0" maxOccurs="1" name="displayName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="firstName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="middleName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="lastName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="companyName" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="preferAddrType" type="tns:AddressType" />
            <s:element minOccurs="0" maxOccurs="1" name="desc" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="localCSID" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="callerID" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="UpdateUser-extraResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="UpdateUser-extraResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="UpdateUser-advance">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
            <s:element minOccurs="1" maxOccurs="1" name="role" type="tns:Role" />
            <s:element minOccurs="1" maxOccurs="1" name="isExternalAuth" type="s:boolean" />
            <s:element minOccurs="1" maxOccurs="1" name="isActive" type="s:boolean" />
            <s:element minOccurs="0" maxOccurs="1" name="displayName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="firstName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="middleName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="lastName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="companyName" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="preferAddrType" type="tns:AddressType" />
            <s:element minOccurs="0" maxOccurs="1" name="desc" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="config" type="tns:ArrayOfUserConfig" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:complexType name="ArrayOfUserConfig">

```

```

        <s:sequence>
            <s:element minOccurs="0" maxOccurs="unbounded" name="UserConfig" nillable="true" type="tns:UserConfig" />
        </s:sequence>
    </s:complexType>
    <s:complexType name="UserConfig">
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="ConfigName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="ConfigValue" type="s:string" />
        </s:sequence>
    </s:complexType>
    <s:element name="UpdateUser-advanceResponse">
        <s:complexType>
            <s:sequence>
                <s:element minOccurs="0" maxOccurs="1" name="UpdateUser-advanceResult" type="s:string" />
            </s:sequence>
        </s:complexType>
    </s:element>
    <s:element name="RemoveUser">
        <s:complexType>
            <s:sequence>
                <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
                <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
            </s:sequence>
        </s:complexType>
    </s:element>
    <s:element name="RemoveUserResponse">
        <s:complexType>
            <s:sequence>
                <s:element minOccurs="0" maxOccurs="1" name="RemoveUserResult" type="s:string" />
            </s:sequence>
        </s:complexType>
    </s:element>
    <s:element name="DeactivateUser">
        <s:complexType>
            <s:sequence>
                <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
                <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
            </s:sequence>
        </s:complexType>
    </s:element>
    <s:element name="DeactivateUserResponse">
        <s:complexType>
            <s:sequence>
                <s:element minOccurs="0" maxOccurs="1" name="DeactivateUserResult" type="s:string" />
            </s:sequence>
        </s:complexType>
    </s:element>
    <s:element name="DeactivateUsers">
        <s:complexType>
            <s:sequence>
                <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
                <s:element minOccurs="0" maxOccurs="1" name="userIDList" type="tns:ArrayOfLong" />
            </s:sequence>
        </s:complexType>
    </s:element>

```

```

        </s:sequence>
    </s:complexType>
</s:element>
<s:complexType name="ArrayOfLong">
    <s:sequence>
        <s:element minOccurs="0" maxOccurs="unbounded" name="long" type="s:long" />
    </s:sequence>
</s:complexType>
<s:element name="DeactivateUsersResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="DeactivateUsersResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="ActivateUser">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="ActivateUserResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="ActivateUserResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="ActivateUsers">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
            <s:element minOccurs="0" maxOccurs="1" name="userIDList" type="tns:ArrayOfLong" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="ActivateUsersResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="ActivateUsersResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="CreateInboundRoute">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
            <s:element minOccurs="0" maxOccurs="1" name="forwardedUserName" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="condition1" type="tns:ConditionType" />
            <s:element minOccurs="0" maxOccurs="1" name="conditionExp1" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>

```

```

        <s:element minOccurs="1" maxOccurs="1" name="isAndOperator" type="s:boolean" />
        <s:element minOccurs="1" maxOccurs="1" name="condition2" type="tns:ConditionType" />
        <s:element minOccurs="0" maxOccurs="1" name="conditionExp2" type="s:string" />
        <s:element minOccurs="1" maxOccurs="1" name="isActive" type="s:boolean" />
        <s:element minOccurs="1" maxOccurs="1" name="priority" type="s:int" />
    </s:sequence>
</s:complexType>
</s:element>
<s:simpleType name="ConditionType">
    <s:restriction base="s:string">
        <s:enumeration value="None" />
        <s:enumeration value="BarCode" />
        <s:enumeration value="FaxPortNo" />
        <s:enumeration value="RoutingDigits" />
        <s:enumeration value="RemoteCallerID" />
        <s:enumeration value="RemoteCSID" />
        <s:enumeration value="ServerName" />
    </s:restriction>
</s:simpleType>
<s:element name="CreateInboundRouteResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="CreateInboundRouteResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="DownloadMessageByte1">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="messageID" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="deliveryNum" type="s:int" />
            <s:element minOccurs="0" maxOccurs="1" name="downloadType" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="startPageIndex" type="s:int" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="DownloadMessageByte1Response">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="DownloadMessageByte1Result" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="output" type="s:base64Binary" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="DownloadMessageInByte2">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="messageID" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="deliveryNum" type="s:int" />
            <s:element minOccurs="0" maxOccurs="1" name="downloadType" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="startPageIndex" type="s:int" />
        </s:sequence>
    </s:complexType>
</s:element>

```



```

        <s:element minOccurs="0" maxOccurs="1" name="excludePageList" type="tns:ArrayOfInt" />
    </s:sequence>
</s:complexType>
</s:element>
<s:element name="DownloadMessageInByte2Response">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="DownloadMessageInByte2Result" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="output" type="s:base64Binary" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="DownloadMessageInByte3">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="deliveryID" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="downloadType" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="startPageIndex" type="s:int" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="DownloadMessageInByte3Response">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="DownloadMessageInByte3Result" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="output" type="s:base64Binary" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="DownloadMessageInByte4">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="deliveryID" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="downloadType" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="startPageIndex" type="s:int" />
            <s:element minOccurs="0" maxOccurs="1" name="excludePageList" type="tns:ArrayOfInt" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="DownloadMessageInByte4Response">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="DownloadMessageInByte4Result" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="output" type="s:base64Binary" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="MarkMsgAsRead">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
            <s:element minOccurs="0" maxOccurs="1" name="msgID" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>

```

```

        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="MarkMsgAsReadResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="MarkMsgAsReadResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="MarkMsgAsUnread">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
            <s:element minOccurs="0" maxOccurs="1" name="msgID" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="MarkMsgAsUnreadResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="MarkMsgAsUnreadResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="IsMessageRead">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
            <s:element minOccurs="0" maxOccurs="1" name="msgID" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="IsMessageReadResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="IsMessageReadResult" type="s:boolean" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="GetMsgFolders">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="GetMsgFoldersResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetMsgFoldersResult" type="tns:ArrayOfString" />
        </s:sequence>
    </s:complexType>

```

```

</s:element>
<s:element name="GetUserListWithDomainID">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
      <s:element minOccurs="1" maxOccurs="1" name="domainID" type="s:long" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetUserListWithDomainIDResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetUserListWithDomainIDResult"
        type="tns:ArrayOfString" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetUserListWithDomainName">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
      <s:element minOccurs="0" maxOccurs="1" name="domainName" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetUserListWithDomainNameResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetUserListWithDomainNameResult"
        type="tns:ArrayOfString" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetUserDetailsByUserID">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
      <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetUserDetailsByUserIDResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetUserDetailsByUserIDResult" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetUserDetailsByUserName">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
      <s:element minOccurs="0" maxOccurs="1" name="userName" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>

```

```

        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="GetUserDetailsByUserNameResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetUserDetailsByUserNameResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="SearchIBRouteRules">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
            <s:element minOccurs="0" maxOccurs="1" name="input" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="SearchIBRouteRulesResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="SearchIBRouteRulesResult" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="GetDomainList">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="GetDomainListResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetDomainListResult" type="tns:ArrayOfString" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="GetUserFaxSettingByUserID">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
            <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="GetUserFaxSettingByUserIDResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetUserFaxSettingByUserIDResult" type="s:string" />
        </s:sequence>
    </s:complexType>

```

```

</s:element>
<s:element name="GetUserFaxSettingByUserName">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="adminID" type="s:long" />
      <s:element minOccurs="0" maxOccurs="1" name="userName" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetUserFaxSettingByUserNameResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetUserFaxSettingByUserNameResult" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="MarkMsgAsDownloaded">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="userID" type="s:long" />
      <s:element minOccurs="0" maxOccurs="1" name="msgID" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="MarkMsgAsDownloadedResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="MarkMsgAsDownloadedResult" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageList4">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="userList" type="tns:ArrayOfLong" />
      <s:element minOccurs="0" maxOccurs="1" name="folderName" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="isDownloaded" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="sortDescending" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="maxResults" type="s:int" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageList4Response">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetMessageList4Result" type="tns:ArrayOfString" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageList5">
  <s:complexType>
    <s:sequence>

```

```

<:element minOccurs="0" maxOccurs="1" name="userList" type="tns:ArrayOfLong" />
<:element minOccurs="0" maxOccurs="1" name="folderName" type="s:string" />
<:element minOccurs="1" maxOccurs="1" name="isDownloaded" type="s:boolean" />
<:element minOccurs="1" maxOccurs="1" name="isRead" type="s:boolean" />
<:element minOccurs="0" maxOccurs="1" name="startDate" type="s:string" />
<:element minOccurs="0" maxOccurs="1" name="endDate" type="s:string" />
<:element minOccurs="1" maxOccurs="1" name="sortDescending" type="s:boolean" />
<:element minOccurs="1" maxOccurs="1" name="maxResults" type="s:int" />
</s:sequence>
</s:complexType>
</s:element>
<s:element name="GetMessageList5Response">
  <s:complexType>
    <s:sequence>
      <:element minOccurs="0" maxOccurs="1" name="GetMessageList5Result" type="tns:ArrayOfString" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageList6">
  <s:complexType>
    <s:sequence>
      <:element minOccurs="0" maxOccurs="1" name="userList" type="tns:ArrayOfLong" />
      <:element minOccurs="0" maxOccurs="1" name="folderName" type="s:string" />
      <:element minOccurs="1" maxOccurs="1" name="isDownloaded" type="s:boolean" />
      <:element minOccurs="0" maxOccurs="1" name="startDate" type="s:string" />
      <:element minOccurs="0" maxOccurs="1" name="endDate" type="s:string" />
      <:element minOccurs="1" maxOccurs="1" name="sortDescending" type="s:boolean" />
      <:element minOccurs="1" maxOccurs="1" name="maxResults" type="s:int" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageList6Response">
  <s:complexType>
    <s:sequence>
      <:element minOccurs="0" maxOccurs="1" name="GetMessageList6Result" type="tns:ArrayOfString" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageList7">
  <s:complexType>
    <s:sequence>
      <:element minOccurs="0" maxOccurs="1" name="userList" type="tns:ArrayOfLong" />
      <:element minOccurs="0" maxOccurs="1" name="folderName" type="s:string" />
      <:element minOccurs="1" maxOccurs="1" name="isDownloaded" type="s:boolean" />
      <:element minOccurs="1" maxOccurs="1" name="isRead" type="s:boolean" />
      <:element minOccurs="1" maxOccurs="1" name="sortDescending" type="s:boolean" />
      <:element minOccurs="1" maxOccurs="1" name="maxResults" type="s:int" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetMessageList7Response">

```

```

                <s:complexType>
                    <s:sequence>
                        <s:element minOccurs="0" maxOccurs="1" name="GetMessageList7Result" type="tns:ArrayOfString" />
                    </s:sequence>
                </s:complexType>
            </s:element>
        </s:schema>
    </wSDL:types>
    <wSDL:message name="AuthenticateUserSoapIn">
        <wSDL:part name="parameters" element="tns:AuthenticateUser" />
    </wSDL:message>
    <wSDL:message name="AuthenticateUserSoapOut">
        <wSDL:part name="parameters" element="tns:AuthenticateUserResponse" />
    </wSDL:message>
    <wSDL:message name="ServerStatusSoapIn">
        <wSDL:part name="parameters" element="tns:ServerStatus" />
    </wSDL:message>
    <wSDL:message name="ServerStatusSoapOut">
        <wSDL:part name="parameters" element="tns:ServerStatusResponse" />
    </wSDL:message>
    <wSDL:message name="SendCompleteMessageSoapIn">
        <wSDL:part name="parameters" element="tns:SendCompleteMessage" />
    </wSDL:message>
    <wSDL:message name="SendCompleteMessageSoapOut">
        <wSDL:part name="parameters" element="tns:SendCompleteMessageResponse" />
    </wSDL:message>
    <wSDL:message name="SendMixedModeSoapIn">
        <wSDL:part name="parameters" element="tns:SendMixedMode" />
    </wSDL:message>
    <wSDL:message name="SendMixedModeSoapOut">
        <wSDL:part name="parameters" element="tns:SendMixedModeResponse" />
    </wSDL:message>
    <wSDL:message name="BasicSendMessageWithBytesSoapIn">
        <wSDL:part name="parameters" element="tns:BasicSendMessageWithBytes" />
    </wSDL:message>
    <wSDL:message name="BasicSendMessageWithBytesSoapOut">
        <wSDL:part name="parameters" element="tns:BasicSendMessageWithBytesResponse" />
    </wSDL:message>
    <wSDL:message name="SendCompleteMessageWithPresetAgentsSoapIn">
        <wSDL:part name="parameters" element="tns:SendCompleteMessageWithPresetAgents" />
    </wSDL:message>
    <wSDL:message name="SendCompleteMessageWithPresetAgentsSoapOut">
        <wSDL:part name="parameters" element="tns:SendCompleteMessageWithPresetAgentsResponse" />
    </wSDL:message>
    <wSDL:message name="SendCompleteMessageWithPresetAgentsAndByPassRenderSoapIn">
        <wSDL:part name="parameters" element="tns:SendCompleteMessageWithPresetAgentsAndByPassRender" />
    </wSDL:message>
    <wSDL:message name="SendCompleteMessageWithPresetAgentsAndByPassRenderSoapOut">
        <wSDL:part name="parameters" element="tns:SendCompleteMessageWithPresetAgentsAndByPassRenderResponse" />
    </wSDL:message>
    <wSDL:message name="DelegateMessageSoapIn">
        <wSDL:part name="parameters" element="tns:DelegateMessage" />
    </wSDL:message>

```

```
</wsdl:message>
<wsdl:message name="DelegateMessageSoapOut">
  <wsdl:part name="parameters" element="tns:DelegateMessageResponse" />
</wsdl:message>
<wsdl:message name="GetMessageDetailsSoapIn">
  <wsdl:part name="parameters" element="tns:GetMessageDetails" />
</wsdl:message>
<wsdl:message name="GetMessageDetailsSoapOut">
  <wsdl:part name="parameters" element="tns:GetMessageDetailsResponse" />
</wsdl:message>
<wsdl:message name="GetMessageStatusSoapIn">
  <wsdl:part name="parameters" element="tns:GetMessageStatus" />
</wsdl:message>
<wsdl:message name="GetMessageStatusSoapOut">
  <wsdl:part name="parameters" element="tns:GetMessageStatusResponse" />
</wsdl:message>
<wsdl:message name="IsMessageDeletedSoapIn">
  <wsdl:part name="parameters" element="tns:IsMessageDeleted" />
</wsdl:message>
<wsdl:message name="IsMessageDeletedSoapOut">
  <wsdl:part name="parameters" element="tns:IsMessageDeletedResponse" />
</wsdl:message>
<wsdl:message name="DeleteMessageSoapIn">
  <wsdl:part name="parameters" element="tns>DeleteMessage" />
</wsdl:message>
<wsdl:message name="DeleteMessageSoapOut">
  <wsdl:part name="parameters" element="tns>DeleteMessageResponse" />
</wsdl:message>
<wsdl:message name="CancelMessageSoapIn">
  <wsdl:part name="parameters" element="tns:CancelMessage" />
</wsdl:message>
<wsdl:message name="CancelMessageSoapOut">
  <wsdl:part name="parameters" element="tns:CancelMessageResponse" />
</wsdl:message>
<wsdl:message name="RetryFailedSoapIn">
  <wsdl:part name="parameters" element="tns:RetryFailed" />
</wsdl:message>
<wsdl:message name="RetryFailedSoapOut">
  <wsdl:part name="parameters" element="tns:RetryFailedResponse" />
</wsdl:message>
<wsdl:message name="ForwardToUserSoapIn">
  <wsdl:part name="parameters" element="tns:ForwardToUser" />
</wsdl:message>
<wsdl:message name="ForwardToUserSoapOut">
  <wsdl:part name="parameters" element="tns:ForwardToUserResponse" />
</wsdl:message>
<wsdl:message name="AssignMessageSoapIn">
  <wsdl:part name="parameters" element="tns:AssignMessage" />
</wsdl:message>
<wsdl:message name="AssignMessageSoapOut">
  <wsdl:part name="parameters" element="tns:AssignMessageResponse" />
</wsdl:message>
```



```
<wsdl:message name="GetMessageList3SoapIn">
  <wsdl:part name="parameters" element="tns:GetMessageList3" />
</wsdl:message>
<wsdl:message name="GetMessageList3SoapOut">
  <wsdl:part name="parameters" element="tns:GetMessageList3Response" />
</wsdl:message>
<wsdl:message name="GetMessageList2SoapIn">
  <wsdl:part name="parameters" element="tns:GetMessageList2" />
</wsdl:message>
<wsdl:message name="GetMessageList2SoapOut">
  <wsdl:part name="parameters" element="tns:GetMessageList2Response" />
</wsdl:message>
<wsdl:message name="GetMessageList1SoapIn">
  <wsdl:part name="parameters" element="tns:GetMessageList1" />
</wsdl:message>
<wsdl:message name="GetMessageList1SoapOut">
  <wsdl:part name="parameters" element="tns:GetMessageList1Response" />
</wsdl:message>
<wsdl:message name="GetMessageListSoapIn">
  <wsdl:part name="parameters" element="tns:GetMessageList" />
</wsdl:message>
<wsdl:message name="GetMessageListSoapOut">
  <wsdl:part name="parameters" element="tns:GetMessageListResponse" />
</wsdl:message>
<wsdl:message name="CreateNewUser1SoapIn">
  <wsdl:part name="parameters" element="tns:CreateNewUser1" />
</wsdl:message>
<wsdl:message name="CreateNewUser1SoapOut">
  <wsdl:part name="parameters" element="tns:CreateNewUser1Response" />
</wsdl:message>
<wsdl:message name="CreateNewUser2SoapIn">
  <wsdl:part name="parameters" element="tns:CreateNewUser2" />
</wsdl:message>
<wsdl:message name="CreateNewUser2SoapOut">
  <wsdl:part name="parameters" element="tns:CreateNewUser2Response" />
</wsdl:message>
<wsdl:message name="CreateNewUser3SoapIn">
  <wsdl:part name="parameters" element="tns:CreateNewUser3" />
</wsdl:message>
<wsdl:message name="CreateNewUser3SoapOut">
  <wsdl:part name="parameters" element="tns:CreateNewUser3Response" />
</wsdl:message>
<wsdl:message name="UpdateUser-basicSoapIn">
  <wsdl:part name="parameters" element="tns:UpdateUser-basic" />
</wsdl:message>
<wsdl:message name="UpdateUser-basicSoapOut">
  <wsdl:part name="parameters" element="tns:UpdateUser-basicResponse" />
</wsdl:message>
<wsdl:message name="UpdateUser-extraSoapIn">
  <wsdl:part name="parameters" element="tns:UpdateUser-extra" />
</wsdl:message>
<wsdl:message name="UpdateUser-extraSoapOut">
```

```

        <wsdl:part name="parameters" element="tns:UpdateUser-extraResponse" />
    </wsdl:message>
    <wsdl:message name="UpdateUser-advanceSoapIn">
        <wsdl:part name="parameters" element="tns:UpdateUser-advance" />
    </wsdl:message>
    <wsdl:message name="UpdateUser-advanceSoapOut">
        <wsdl:part name="parameters" element="tns:UpdateUser-advanceResponse" />
    </wsdl:message>
    <wsdl:message name="RemoveUserSoapIn">
        <wsdl:part name="parameters" element="tns:RemoveUser" />
    </wsdl:message>
    <wsdl:message name="RemoveUserSoapOut">
        <wsdl:part name="parameters" element="tns:RemoveUserResponse" />
    </wsdl:message>
    <wsdl:message name="DeactivateUserSoapIn">
        <wsdl:part name="parameters" element="tns:DeactivateUser" />
    </wsdl:message>
    <wsdl:message name="DeactivateUserSoapOut">
        <wsdl:part name="parameters" element="tns:DeactivateUserResponse" />
    </wsdl:message>
    <wsdl:message name="DeactivateUsersSoapIn">
        <wsdl:part name="parameters" element="tns:DeactivateUsers" />
    </wsdl:message>
    <wsdl:message name="DeactivateUsersSoapOut">
        <wsdl:part name="parameters" element="tns:DeactivateUsersResponse" />
    </wsdl:message>
    <wsdl:message name="ActivateUserSoapIn">
        <wsdl:part name="parameters" element="tns:ActivateUser" />
    </wsdl:message>
    <wsdl:message name="ActivateUserSoapOut">
        <wsdl:part name="parameters" element="tns:ActivateUserResponse" />
    </wsdl:message>
    <wsdl:message name="ActivateUsersSoapIn">
        <wsdl:part name="parameters" element="tns:ActivateUsers" />
    </wsdl:message>
    <wsdl:message name="ActivateUsersSoapOut">
        <wsdl:part name="parameters" element="tns:ActivateUsersResponse" />
    </wsdl:message>
    <wsdl:message name="CreateInboundRouteSoapIn">
        <wsdl:part name="parameters" element="tns:CreateInboundRoute" />
    </wsdl:message>
    <wsdl:message name="CreateInboundRouteSoapOut">
        <wsdl:part name="parameters" element="tns:CreateInboundRouteResponse" />
    </wsdl:message>
    <wsdl:message name="DownloadMessageByte1SoapIn">
        <wsdl:part name="parameters" element="tns:DownloadMessageByte1" />
    </wsdl:message>
    <wsdl:message name="DownloadMessageByte1SoapOut">
        <wsdl:part name="parameters" element="tns:DownloadMessageByte1Response" />
    </wsdl:message>
    <wsdl:message name="DownloadMessageInByte2SoapIn">
        <wsdl:part name="parameters" element="tns:DownloadMessageInByte2" />
    </wsdl:message>

```

```
</wsdl:message>
<wsdl:message name="DownloadMessageInByte2SoapOut">
  <wsdl:part name="parameters" element="tns:DownloadMessageInByte2Response" />
</wsdl:message>
<wsdl:message name="DownloadMessageInByte3SoapIn">
  <wsdl:part name="parameters" element="tns:DownloadMessageInByte3" />
</wsdl:message>
<wsdl:message name="DownloadMessageInByte3SoapOut">
  <wsdl:part name="parameters" element="tns:DownloadMessageInByte3Response" />
</wsdl:message>
<wsdl:message name="DownloadMessageInByte4SoapIn">
  <wsdl:part name="parameters" element="tns:DownloadMessageInByte4" />
</wsdl:message>
<wsdl:message name="DownloadMessageInByte4SoapOut">
  <wsdl:part name="parameters" element="tns:DownloadMessageInByte4Response" />
</wsdl:message>
<wsdl:message name="MarkMsgAsReadSoapIn">
  <wsdl:part name="parameters" element="tns:MarkMsgAsRead" />
</wsdl:message>
<wsdl:message name="MarkMsgAsReadSoapOut">
  <wsdl:part name="parameters" element="tns:MarkMsgAsReadResponse" />
</wsdl:message>
<wsdl:message name="MarkMsgAsUnreadSoapIn">
  <wsdl:part name="parameters" element="tns:MarkMsgAsUnread" />
</wsdl:message>
<wsdl:message name="MarkMsgAsUnreadSoapOut">
  <wsdl:part name="parameters" element="tns:MarkMsgAsUnreadResponse" />
</wsdl:message>
<wsdl:message name="IsMessageReadSoapIn">
  <wsdl:part name="parameters" element="tns:IsMessageRead" />
</wsdl:message>
<wsdl:message name="IsMessageReadSoapOut">
  <wsdl:part name="parameters" element="tns:IsMessageReadResponse" />
</wsdl:message>
<wsdl:message name="GetMsgFoldersSoapIn">
  <wsdl:part name="parameters" element="tns:GetMsgFolders" />
</wsdl:message>
<wsdl:message name="GetMsgFoldersSoapOut">
  <wsdl:part name="parameters" element="tns:GetMsgFoldersResponse" />
</wsdl:message>
<wsdl:message name="GetUserListWithDomainIDSoapIn">
  <wsdl:part name="parameters" element="tns:GetUserListWithDomainID" />
</wsdl:message>
<wsdl:message name="GetUserListWithDomainIDSoapOut">
  <wsdl:part name="parameters" element="tns:GetUserListWithDomainIDResponse" />
</wsdl:message>
<wsdl:message name="GetUserListWithDomainNameSoapIn">
  <wsdl:part name="parameters" element="tns:GetUserListWithDomainName" />
</wsdl:message>
<wsdl:message name="GetUserListWithDomainNameSoapOut">
  <wsdl:part name="parameters" element="tns:GetUserListWithDomainNameResponse" />
</wsdl:message>
```

```

<wsdl:message name="GetUserDetailsByUserIDSoapIn">
  <wsdl:part name="parameters" element="tns:GetUserDetailsByUserID" />
</wsdl:message>
<wsdl:message name="GetUserDetailsByUserIDSoapOut">
  <wsdl:part name="parameters" element="tns:GetUserDetailsByUserIDResponse" />
</wsdl:message>
<wsdl:message name="GetUserDetailsByUserNameSoapIn">
  <wsdl:part name="parameters" element="tns:GetUserDetailsByUserName" />
</wsdl:message>
<wsdl:message name="GetUserDetailsByUserNameSoapOut">
  <wsdl:part name="parameters" element="tns:GetUserDetailsByUserNameResponse" />
</wsdl:message>
<wsdl:message name="SearchIBRouteRulesSoapIn">
  <wsdl:part name="parameters" element="tns:SearchIBRouteRules" />
</wsdl:message>
<wsdl:message name="SearchIBRouteRulesSoapOut">
  <wsdl:part name="parameters" element="tns:SearchIBRouteRulesResponse" />
</wsdl:message>
<wsdl:message name="GetDomainListSoapIn">
  <wsdl:part name="parameters" element="tns:GetDomainList" />
</wsdl:message>
<wsdl:message name="GetDomainListSoapOut">
  <wsdl:part name="parameters" element="tns:GetDomainListResponse" />
</wsdl:message>
<wsdl:message name="GetUserFaxSettingByUserIDSoapIn">
  <wsdl:part name="parameters" element="tns:GetUserFaxSettingByUserID" />
</wsdl:message>
<wsdl:message name="GetUserFaxSettingByUserIDSoapOut">
  <wsdl:part name="parameters" element="tns:GetUserFaxSettingByUserIDResponse" />
</wsdl:message>
<wsdl:message name="GetUserFaxSettingByUserNameSoapIn">
  <wsdl:part name="parameters" element="tns:GetUserFaxSettingByUserName" />
</wsdl:message>
<wsdl:message name="GetUserFaxSettingByUserNameSoapOut">
  <wsdl:part name="parameters" element="tns:GetUserFaxSettingByUserNameResponse" />
</wsdl:message>
<wsdl:message name="MarkMsgAsDownloadedSoapIn">
  <wsdl:part name="parameters" element="tns:MarkMsgAsDownloaded" />
</wsdl:message>
<wsdl:message name="MarkMsgAsDownloadedSoapOut">
  <wsdl:part name="parameters" element="tns:MarkMsgAsDownloadedResponse" />
</wsdl:message>
<wsdl:message name="GetMessageList4SoapIn">
  <wsdl:part name="parameters" element="tns:GetMessageList4" />
</wsdl:message>
<wsdl:message name="GetMessageList4SoapOut">
  <wsdl:part name="parameters" element="tns:GetMessageList4Response" />
</wsdl:message>
<wsdl:message name="GetMessageList5SoapIn">
  <wsdl:part name="parameters" element="tns:GetMessageList5" />
</wsdl:message>
<wsdl:message name="GetMessageList5SoapOut">

```

```

<wsdl:part name="parameters" element="tns:GetMessageList5Response" />
</wsdl:message>
<wsdl:message name="GetMessageList6SoapIn">
  <wsdl:part name="parameters" element="tns:GetMessageList6" />
</wsdl:message>
<wsdl:message name="GetMessageList6SoapOut">
  <wsdl:part name="parameters" element="tns:GetMessageList6Response" />
</wsdl:message>
<wsdl:message name="GetMessageList7SoapIn">
  <wsdl:part name="parameters" element="tns:GetMessageList7" />
</wsdl:message>
<wsdl:message name="GetMessageList7SoapOut">
  <wsdl:part name="parameters" element="tns:GetMessageList7Response" />
</wsdl:message>
<wsdl:portType name="wfApiSoap">
  <wsdl:operation name="AuthenticateUser">
    <wsdl:documentation>Authenticate user via SDK</wsdl:documentation>
    <wsdl:input message="tns:AuthenticateUserSoapIn" />
    <wsdl:output message="tns:AuthenticateUserSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="ServerStatus">
    <wsdl:documentation>Check server status</wsdl:documentation>
    <wsdl:input message="tns:ServerStatusSoapIn" />
    <wsdl:output message="tns:ServerStatusSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="SendMessage">
    <wsdl:documentation>Send message to multiple recipients with multiple documents</wsdl:documentation>
    <wsdl:input name="SendCompleteMessage" message="tns:SendCompleteMessageSoapIn" />
    <wsdl:output name="SendCompleteMessage" message="tns:SendCompleteMessageSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="SendMixedMode">
    <wsdl:documentation>Send message in mixed mode.</wsdl:documentation>
    <wsdl:input message="tns:SendMixedModeSoapIn" />
    <wsdl:output message="tns:SendMixedModeSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="SendMessage">
    <wsdl:documentation>Basic send message with documents from bytes</wsdl:documentation>
    <wsdl:input name="BasicSendMessageWithBytes" message="tns:BasicSendMessageWithBytesSoapIn" />
    <wsdl:output name="BasicSendMessageWithBytes" message="tns:BasicSendMessageWithBytesSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="SendMessage">
    <wsdl:documentation>Send message to multiple recipients with multiple documents</wsdl:documentation>
    <wsdl:input name="SendCompleteMessageWithPresetAgents" message="tns:SendCompleteMessageWithPresetAgentsSoapIn" />
    <wsdl:output name="SendCompleteMessageWithPresetAgents" message="tns:SendCompleteMessageWithPresetAgentsSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="SendMessage">
    <wsdl:documentation>Send message to multiple recipients with multiple documents</wsdl:documentation>
    <wsdl:input name="SendCompleteMessageWithPresetAgentsAndByPassRender"
      message="tns:SendCompleteMessageWithPresetAgentsAndByPassRenderSoapIn" />
    <wsdl:output name="SendCompleteMessageWithPresetAgentsAndByPassRender"
      message="tns:SendCompleteMessageWithPresetAgentsAndByPassRenderSoapOut" />
  </wsdl:operation>

```

```

</wsdl:operation>
<wsdl:operation name="DelegateMessage">
  <wsdl:documentation>Send message on behalf of other user</wsdl:documentation>
  <wsdl:input message="tns:DelegateMessageSoapIn" />
  <wsdl:output message="tns:DelegateMessageSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetMessageDetails">
  <wsdl:documentation>Gets message details in xml string</wsdl:documentation>
  <wsdl:input message="tns:GetMessageDetailsSoapIn" />
  <wsdl:output message="tns:GetMessageDetailsSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetMessageStatus">
  <wsdl:documentation>Get message's status.</wsdl:documentation>
  <wsdl:input message="tns:GetMessageStatusSoapIn" />
  <wsdl:output message="tns:GetMessageStatusSoapOut" />
</wsdl:operation>
<wsdl:operation name="IsMessageDeleted">
  <wsdl:documentation>To check whether the message is being deleted</wsdl:documentation>
  <wsdl:input message="tns:IsMessageDeletedSoapIn" />
  <wsdl:output message="tns:IsMessageDeletedSoapOut" />
</wsdl:operation>
<wsdl:operation name="DeleteMessage">
  <wsdl:documentation>Deletes message.</wsdl:documentation>
  <wsdl:input message="tns>DeleteMessageSoapIn" />
  <wsdl:output message="tns>DeleteMessageSoapOut" />
</wsdl:operation>
<wsdl:operation name="CancelMessage">
  <wsdl:input message="tns:CancelMessageSoapIn" />
  <wsdl:output message="tns:CancelMessageSoapOut" />
</wsdl:operation>
<wsdl:operation name="RetryFailed">
  <wsdl:input message="tns:RetryFailedSoapIn" />
  <wsdl:output message="tns:RetryFailedSoapOut" />
</wsdl:operation>
<wsdl:operation name="ForwardToUser">
  <wsdl:input message="tns:ForwardToUserSoapIn" />
  <wsdl:output message="tns:ForwardToUserSoapOut" />
</wsdl:operation>
<wsdl:operation name="AssignMessage">
  <wsdl:input message="tns:AssignMessageSoapIn" />
  <wsdl:output message="tns:AssignMessageSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <wsdl:documentation>Get message list based on folder name and isread/unread.</wsdl:documentation>
  <wsdl:input name="GetMessageList3" message="tns:GetMessageList3SoapIn" />
  <wsdl:output name="GetMessageList3" message="tns:GetMessageList3SoapOut" />
</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <wsdl:documentation>Get message list based on folder name, created date range and is read/unread.</wsdl:documentation>
  <wsdl:input name="GetMessageList2" message="tns:GetMessageList2SoapIn" />
  <wsdl:output name="GetMessageList2" message="tns:GetMessageList2SoapOut" />

```

```

</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <wsdl:documentation>Get message list based on folder name and created date range.</wsdl:documentation>
  <wsdl:input name="GetMessageList1" message="tns:GetMessageList1SoapIn" />
  <wsdl:output name="GetMessageList1" message="tns:GetMessageList1SoapOut" />
</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <wsdl:documentation>Get message list based on folder name.</wsdl:documentation>
  <wsdl:input message="tns:GetMessageListSoapIn" />
  <wsdl:output message="tns:GetMessageListSoapOut" />
</wsdl:operation>
<wsdl:operation name="CreateUser">
  <wsdl:documentation>Create new user with default NOR/NOS=false</wsdl:documentation>
  <wsdl:input name="CreateNewUser1" message="tns:CreateNewUser1SoapIn" />
  <wsdl:output name="CreateNewUser1" message="tns:CreateNewUser1SoapOut" />
</wsdl:operation>
<wsdl:operation name="CreateUser">
  <wsdl:documentation>Create new user with NOR/NOS values</wsdl:documentation>
  <wsdl:input name="CreateNewUser2" message="tns:CreateNewUser2SoapIn" />
  <wsdl:output name="CreateNewUser2" message="tns:CreateNewUser2SoapOut" />
</wsdl:operation>
<wsdl:operation name="CreateUser">
  <wsdl:documentation>Create new user with NOR/NOS values,callerID and CSID</wsdl:documentation>
  <wsdl:input name="CreateNewUser3" message="tns:CreateNewUser3SoapIn" />
  <wsdl:output name="CreateNewUser3" message="tns:CreateNewUser3SoapOut" />
</wsdl:operation>
<wsdl:operation name="UpdateUser">
  <wsdl:documentation>Update user with basic profile</wsdl:documentation>
  <wsdl:input name="UpdateUser-basic" message="tns:UpdateUser-basicSoapIn" />
  <wsdl:output name="UpdateUser-basic" message="tns:UpdateUser-basicSoapOut" />
</wsdl:operation>
<wsdl:operation name="UpdateUser">
  <wsdl:documentation>Update user with basic profile, local CSID, and caller ID</wsdl:documentation>
  <wsdl:input name="UpdateUser-extra" message="tns:UpdateUser-extraSoapIn" />
  <wsdl:output name="UpdateUser-extra" message="tns:UpdateUser-extraSoapOut" />
</wsdl:operation>
<wsdl:operation name="UpdateUser">
  <wsdl:documentation>Update user with basic profile, and user's config</wsdl:documentation>
  <wsdl:input name="UpdateUser-advance" message="tns:UpdateUser-advanceSoapIn" />
  <wsdl:output name="UpdateUser-advance" message="tns:UpdateUser-advanceSoapOut" />
</wsdl:operation>
<wsdl:operation name="RemoveUser">
  <wsdl:documentation>Remove user</wsdl:documentation>
  <wsdl:input message="tns:RemoveUserSoapIn" />
  <wsdl:output message="tns:RemoveUserSoapOut" />
</wsdl:operation>
<wsdl:operation name="DeactivateUser">
  <wsdl:documentation>Set user to be inactive</wsdl:documentation>
  <wsdl:input message="tns:DeactivateUserSoapIn" />
  <wsdl:output message="tns:DeactivateUserSoapOut" />
</wsdl:operation>

```

```

<wsdl:operation name="DeactivateUsers">
  <wsdl:documentation>Set a list of users to be inactive</wsdl:documentation>
  <wsdl:input message="tns:DeactivateUsersSoapIn" />
  <wsdl:output message="tns:DeactivateUsersSoapOut" />
</wsdl:operation>
<wsdl:operation name="ActivateUser">
  <wsdl:documentation>Set user to be active</wsdl:documentation>
  <wsdl:input message="tns:ActivateUserSoapIn" />
  <wsdl:output message="tns:ActivateUserSoapOut" />
</wsdl:operation>
<wsdl:operation name="ActivateUsers">
  <wsdl:documentation>Set a list of users to be active</wsdl:documentation>
  <wsdl:input message="tns:ActivateUsersSoapIn" />
  <wsdl:output message="tns:ActivateUsersSoapOut" />
</wsdl:operation>
<wsdl:operation name="CreateInboundRoute">
  <wsdl:input message="tns:CreateInboundRouteSoapIn" />
  <wsdl:output message="tns:CreateInboundRouteSoapOut" />
</wsdl:operation>
<wsdl:operation name="DownloadMessageByte">
  <wsdl:documentation>Download message's binary with specified message id, delivery number, download type and start page
  index</wsdl:documentation>
  <wsdl:input name="DownloadMessageByte1" message="tns:DownloadMessageByte1SoapIn" />
  <wsdl:output name="DownloadMessageByte1" message="tns:DownloadMessageByte1SoapOut" />
</wsdl:operation>
<wsdl:operation name="DownloadMessageByte">
  <wsdl:documentation>Download message's binary with specified message id, delivery number, download type, start page index and
  list of excluded pages</wsdl:documentation>
  <wsdl:input name="DownloadMessageInByte2" message="tns:DownloadMessageInByte2SoapIn" />
  <wsdl:output name="DownloadMessageInByte2" message="tns:DownloadMessageInByte2SoapOut" />
</wsdl:operation>
<wsdl:operation name="DownloadMessageByte">
  <wsdl:documentation>Download message's binary with specified delivery ID, download type and start page
  index</wsdl:documentation>
  <wsdl:input name="DownloadMessageInByte3" message="tns:DownloadMessageInByte3SoapIn" />
  <wsdl:output name="DownloadMessageInByte3" message="tns:DownloadMessageInByte3SoapOut" />
</wsdl:operation>
<wsdl:operation name="DownloadMessageByte">
  <wsdl:documentation>Download message's binary with specified delivery id, download type, start page index and list of excluded
  pages</wsdl:documentation>
  <wsdl:input name="DownloadMessageInByte4" message="tns:DownloadMessageInByte4SoapIn" />
  <wsdl:output name="DownloadMessageInByte4" message="tns:DownloadMessageInByte4SoapOut" />
</wsdl:operation>
<wsdl:operation name="MarkMsgAsRead">
  <wsdl:input message="tns:MarkMsgAsReadSoapIn" />
  <wsdl:output message="tns:MarkMsgAsReadSoapOut" />
</wsdl:operation>
<wsdl:operation name="MarkMsgAsUnread">
  <wsdl:input message="tns:MarkMsgAsUnreadSoapIn" />
  <wsdl:output message="tns:MarkMsgAsUnreadSoapOut" />
</wsdl:operation>
<wsdl:operation name="IsMessageRead">

```



```

<wsdl:input message="tns:IsMessageReadSoapIn" />
<wsdl:output message="tns:IsMessageReadSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetMsgFolders">
  <wsdl:input message="tns:GetMsgFoldersSoapIn" />
  <wsdl:output message="tns:GetMsgFoldersSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetUserList">
  <wsdl:input name="GetUserListWithDomainID" message="tns:GetUserListWithDomainIDSoapIn" />
  <wsdl:output name="GetUserListWithDomainID" message="tns:GetUserListWithDomainIDSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetUserList">
  <wsdl:input name="GetUserListWithDomainName" message="tns:GetUserListWithDomainNameSoapIn" />
  <wsdl:output name="GetUserListWithDomainName" message="tns:GetUserListWithDomainNameSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetUserDetails">
  <wsdl:input name="GetUserDetailsByUserID" message="tns:GetUserDetailsByUserIDSoapIn" />
  <wsdl:output name="GetUserDetailsByUserID" message="tns:GetUserDetailsByUserIDSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetUserDetails">
  <wsdl:input name="GetUserDetailsByUserName" message="tns:GetUserDetailsByUserNameSoapIn" />
  <wsdl:output name="GetUserDetailsByUserName" message="tns:GetUserDetailsByUserNameSoapOut" />
</wsdl:operation>
<wsdl:operation name="SearchIBRouteRules">
  <wsdl:input message="tns:SearchIBRouteRulesSoapIn" />
  <wsdl:output message="tns:SearchIBRouteRulesSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetDomainList">
  <wsdl:input message="tns:GetDomainListSoapIn" />
  <wsdl:output message="tns:GetDomainListSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetUserFaxSetting">
  <wsdl:input name="GetUserFaxSettingByUserID" message="tns:GetUserFaxSettingByUserIDSoapIn" />
  <wsdl:output name="GetUserFaxSettingByUserID" message="tns:GetUserFaxSettingByUserIDSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetUserFaxSetting">
  <wsdl:input name="GetUserFaxSettingByUserName" message="tns:GetUserFaxSettingByUserNameSoapIn" />
  <wsdl:output name="GetUserFaxSettingByUserName" message="tns:GetUserFaxSettingByUserNameSoapOut" />
</wsdl:operation>
<wsdl:operation name="MarkMsgAsDownloaded">
  <wsdl:input message="tns:MarkMsgAsDownloadedSoapIn" />
  <wsdl:output message="tns:MarkMsgAsDownloadedSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <wsdl:documentation>Get message list based on list of users, folder name and whether is downloaded externally</wsdl:documentation>
  <wsdl:input name="GetMessageList4" message="tns:GetMessageList4SoapIn" />
  <wsdl:output name="GetMessageList4" message="tns:GetMessageList4SoapOut" />
</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <wsdl:documentation>Get message list based on list of users, folder name, whether is downloaded externally, is read, start and

```

```

end date</wsdl:documentation>

<wsdl:input name="GetMessageList5" message="tns:GetMessageList5SoapIn" />
<wsdl:output name="GetMessageList5" message="tns:GetMessageList5SoapOut" />
</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <wsdl:documentation>Get message list based on list of users, folder name, whether is downloaded externally, start and end
  date</wsdl:documentation>
  <wsdl:input name="GetMessageList6" message="tns:GetMessageList6SoapIn" />
  <wsdl:output name="GetMessageList6" message="tns:GetMessageList6SoapOut" />
</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <wsdl:documentation>Get message list based on list of users, folder name, whether is downloaded externally, and is
  read</wsdl:documentation>
  <wsdl:input name="GetMessageList7" message="tns:GetMessageList7SoapIn" />
  <wsdl:output name="GetMessageList7" message="tns:GetMessageList7SoapOut" />
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="wfApiSoap" type="tns:wfApiSoap">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" />
  <wsdl:operation name="AuthenticateUser">
    <soap:operation soapAction="http://faxcoreserver/services/AuthenticateUser" style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="ServerStatus">
    <soap:operation soapAction="http://faxcoreserver/services/ServerStatus" style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="SendMessage">
    <soap:operation soapAction="http://faxcoreserver/services/SendCompleteMessage" style="document" />
    <wsdl:input name="SendCompleteMessage">
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output name="SendCompleteMessage">
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="SendMixedMode">
    <soap:operation soapAction="http://faxcoreserver/services/SendMixedMode" style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>

```

```

        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SendMessage">
    <soap:operation soapAction="http://faxcoreserver/services/BasicSendMessageWithBytes" style="document" />
    <wsdl:input name="BasicSendMessageWithBytes">
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output name="BasicSendMessageWithBytes">
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SendMessage">
    <soap:operation soapAction="http://faxcoreserver/services/SendCompleteMessageWithPresetAgents" style="document" />
    <wsdl:input name="SendCompleteMessageWithPresetAgents">
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output name="SendCompleteMessageWithPresetAgents">
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SendMessage">
    <soap:operation soapAction="http://faxcoreserver/services/SendCompleteMessageWithPresetAgentsAndByPassRender" style="document" />
    <wsdl:input name="SendCompleteMessageWithPresetAgentsAndByPassRender">
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output name="SendCompleteMessageWithPresetAgentsAndByPassRender">
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DelegateMessage">
    <soap:operation soapAction="http://faxcoreserver/services/DelegateMessage" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageDetails">
    <soap:operation soapAction="http://faxcoreserver/services/GetMessageDetails" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageStatus">
    <soap:operation soapAction="http://faxcoreserver/services/GetMessageStatus" style="document" />
    <wsdl:input>

```

```

        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="IsMessageDeleted">
    <soap:operation soapAction="http://faxcoreserver/services/IsMessageDeleted" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DeleteMessage">
    <soap:operation soapAction="http://faxcoreserver/services/DeleteMessage" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CancelMessage">
    <soap:operation soapAction="http://faxcoreserver/services/CancelMessage" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="RetryFailed">
    <soap:operation soapAction="http://faxcoreserver/services/RetryFailed" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ForwardToUser">
    <soap:operation soapAction="http://faxcoreserver/services/ForwardToUser" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="AssignMessage">

```

```

<soap:operation soapAction="http://faxcoreserver/services/AssignMessage" style="document" />
<wsdl:input>
  <soap:body use="literal" />
</wsdl:input>
<wsdl:output>
  <soap:body use="literal" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <soap:operation soapAction="http://faxcoreserver/services/GetMessageList3" style="document" />
  <wsdl:input name="GetMessageList3">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="GetMessageList3">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <soap:operation soapAction="http://faxcoreserver/services/GetMessageList2" style="document" />
  <wsdl:input name="GetMessageList2">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="GetMessageList2">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <soap:operation soapAction="http://faxcoreserver/services/GetMessageList1" style="document" />
  <wsdl:input name="GetMessageList1">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="GetMessageList1">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <soap:operation soapAction="http://faxcoreserver/services/GetMessageList" style="document" />
  <wsdl:input>
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CreateUser">
  <soap:operation soapAction="http://faxcoreserver/services/CreateNewUser1" style="document" />
  <wsdl:input name="CreateNewUser1">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="CreateNewUser1">
    <soap:body use="literal" />
  </wsdl:output>

```

```

</wsdl:operation>
<wsdl:operation name="CreateUser">
  <soap:operation soapAction="http://faxcoreserver/services/CreateNewUser2" style="document" />
  <wsdl:input name="CreateNewUser2">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="CreateNewUser2">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CreateUser">
  <soap:operation soapAction="http://faxcoreserver/services/CreateNewUser3" style="document" />
  <wsdl:input name="CreateNewUser3">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="CreateNewUser3">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="UpdateUser">
  <soap:operation soapAction="http://faxcoreserver/services/UpdateUser-basic" style="document" />
  <wsdl:input name="UpdateUser-basic">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="UpdateUser-basic">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="UpdateUser">
  <soap:operation soapAction="http://faxcoreserver/services/UpdateUser-extra" style="document" />
  <wsdl:input name="UpdateUser-extra">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="UpdateUser-extra">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="UpdateUser">
  <soap:operation soapAction="http://faxcoreserver/services/UpdateUser-advance" style="document" />
  <wsdl:input name="UpdateUser-advance">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="UpdateUser-advance">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="RemoveUser">
  <soap:operation soapAction="http://faxcoreserver/services/RemoveUser" style="document" />
  <wsdl:input>
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output>

```

```

        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DeactivateUser">
    <soap:operation soapAction="http://faxcoreserver/services/DeactivateUser" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DeactivateUsers">
    <soap:operation soapAction="http://faxcoreserver/services/DeactivateUsers" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ActivateUser">
    <soap:operation soapAction="http://faxcoreserver/services/ActivateUser" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ActivateUsers">
    <soap:operation soapAction="http://faxcoreserver/services/ActivateUsers" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CreateInboundRoute">
    <soap:operation soapAction="http://faxcoreserver/services/CreateInboundRoute" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DownloadMessageByte">
    <soap:operation soapAction="http://faxcoreserver/services/DownloadMessageByte1" style="document" />
    <wsdl:input name="DownloadMessageByte1">
        <soap:body use="literal" />
    </wsdl:input>

```

```

</wsdl:input>
<wsdl:output name="DownloadMessageByte1">
  <soap:body use="literal" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="DownloadMessageByte">
  <soap:operation soapAction="http://faxcoreserver/services/DownloadMessageInByte2" style="document" />
  <wsdl:input name="DownloadMessageInByte2">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="DownloadMessageInByte2">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DownloadMessageByte">
  <soap:operation soapAction="http://faxcoreserver/services/DownloadMessageInByte3" style="document" />
  <wsdl:input name="DownloadMessageInByte3">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="DownloadMessageInByte3">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DownloadMessageByte">
  <soap:operation soapAction="http://faxcoreserver/services/DownloadMessageInByte4" style="document" />
  <wsdl:input name="DownloadMessageInByte4">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="DownloadMessageInByte4">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="MarkMsgAsRead">
  <soap:operation soapAction="http://faxcoreserver/services/MarkMsgAsRead" style="document" />
  <wsdl:input>
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="MarkMsgAsUnread">
  <soap:operation soapAction="http://faxcoreserver/services/MarkMsgAsUnread" style="document" />
  <wsdl:input>
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="IsMessageRead">
  <soap:operation soapAction="http://faxcoreserver/services/IsMessageRead" style="document" />

```



```

        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetMsgFolders">
        <soap:operation soapAction="http://faxcoreserver/services/GetMsgFolders" style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetUserList">
        <soap:operation soapAction="http://faxcoreserver/services/GetUserListWithDomainID" style="document" />
        <wsdl:input name="GetUserListWithDomainID">
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output name="GetUserListWithDomainID">
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetUserList">
        <soap:operation soapAction="http://faxcoreserver/services/GetUserListWithDomainName" style="document" />
        <wsdl:input name="GetUserListWithDomainName">
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output name="GetUserListWithDomainName">
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetUserDetails">
        <soap:operation soapAction="http://faxcoreserver/services/GetUserDetailsByUserID" style="document" />
        <wsdl:input name="GetUserDetailsByUserID">
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output name="GetUserDetailsByUserID">
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetUserDetails">
        <soap:operation soapAction="http://faxcoreserver/services/GetUserDetailsByUserName" style="document" />
        <wsdl:input name="GetUserDetailsByUserName">
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output name="GetUserDetailsByUserName">
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>

```

```

<wsdl:operation name="SearchBRouteRules">
  <soap:operation soapAction="http://faxcoreserver/services/SearchBRouteRules" style="document" />
  <wsdl:input>
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetDomainList">
  <soap:operation soapAction="http://faxcoreserver/services/GetDomainList" style="document" />
  <wsdl:input>
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetUserFaxSetting">
  <soap:operation soapAction="http://faxcoreserver/services/GetUserFaxSettingByUserID" style="document" />
  <wsdl:input name="GetUserFaxSettingByUserID">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="GetUserFaxSettingByUserID">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetUserFaxSetting">
  <soap:operation soapAction="http://faxcoreserver/services/GetUserFaxSettingByUserName" style="document" />
  <wsdl:input name="GetUserFaxSettingByUserName">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="GetUserFaxSettingByUserName">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="MarkMsgAsDownloaded">
  <soap:operation soapAction="http://faxcoreserver/services/MarkMsgAsDownloaded" style="document" />
  <wsdl:input>
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
  <soap:operation soapAction="http://faxcoreserver/services/GetMessageList4" style="document" />
  <wsdl:input name="GetMessageList4">
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output name="GetMessageList4">
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>

```

```

        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetMessageList">
        <soap:operation soapAction="http://faxcoreserver/services/GetMessageList5" style="document" />
        <wsdl:input name="GetMessageList5">
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output name="GetMessageList5">
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetMessageList">
        <soap:operation soapAction="http://faxcoreserver/services/GetMessageList6" style="document" />
        <wsdl:input name="GetMessageList6">
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output name="GetMessageList6">
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetMessageList">
        <soap:operation soapAction="http://faxcoreserver/services/GetMessageList7" style="document" />
        <wsdl:input name="GetMessageList7">
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output name="GetMessageList7">
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
</wsdl:binding>
<wsdl:binding name="wfApiSoap12" type="tns:wfApiSoap">
    <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" />
    <wsdl:operation name="AuthenticateUser">
        <soap12:operation soapAction="http://faxcoreserver/services/AuthenticateUser" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="ServerStatus">
        <soap12:operation soapAction="http://faxcoreserver/services/ServerStatus" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SendMessage">
        <soap12:operation soapAction="http://faxcoreserver/services/SendCompleteMessage" style="document" />

```

```

    <wsdl:input name="SendCompleteMessage">
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output name="SendCompleteMessage">
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SendMixedMode">
    <soap12:operation soapAction="http://faxcoreserver/services/SendMixedMode" style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SendMessage">
    <soap12:operation soapAction="http://faxcoreserver/services/BasicSendMessageWithBytes" style="document" />
    <wsdl:input name="BasicSendMessageWithBytes">
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output name="BasicSendMessageWithBytes">
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SendMessage">
    <soap12:operation soapAction="http://faxcoreserver/services/SendCompleteMessageWithPresetAgents" style="document" />
    <wsdl:input name="SendCompleteMessageWithPresetAgents">
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output name="SendCompleteMessageWithPresetAgents">
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SendMessage">
    <soap12:operation soapAction="http://faxcoreserver/services/SendCompleteMessageWithPresetAgentsAndByPassRender"
style="document" />
    <wsdl:input name="SendCompleteMessageWithPresetAgentsAndByPassRender">
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output name="SendCompleteMessageWithPresetAgentsAndByPassRender">
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DelegateMessage">
    <soap12:operation soapAction="http://faxcoreserver/services/DelegateMessage" style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>

```

```

</wsdl:operation>
<wsdl:operation name="GetMessageDetails">
  <soap12:operation soapAction="http://faxcoreserver/services/GetMessageDetails" style="document" />
  <wsdl:input>
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageStatus">
  <soap12:operation soapAction="http://faxcoreserver/services/GetMessageStatus" style="document" />
  <wsdl:input>
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="IsMessageDeleted">
  <soap12:operation soapAction="http://faxcoreserver/services/IsMessageDeleted" style="document" />
  <wsdl:input>
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DeleteMessage">
  <soap12:operation soapAction="http://faxcoreserver/services/DeleteMessage" style="document" />
  <wsdl:input>
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CancelMessage">
  <soap12:operation soapAction="http://faxcoreserver/services/CancelMessage" style="document" />
  <wsdl:input>
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="RetryFailed">
  <soap12:operation soapAction="http://faxcoreserver/services/RetryFailed" style="document" />
  <wsdl:input>
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output>

```

```

        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ForwardToUser">
    <soap12:operation soapAction="http://faxcoreserver/services/ForwardToUser" style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="AssignMessage">
    <soap12:operation soapAction="http://faxcoreserver/services/AssignMessage" style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
    <soap12:operation soapAction="http://faxcoreserver/services/GetMessageList3" style="document" />
    <wsdl:input name="GetMessageList3">
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output name="GetMessageList3">
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
    <soap12:operation soapAction="http://faxcoreserver/services/GetMessageList2" style="document" />
    <wsdl:input name="GetMessageList2">
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output name="GetMessageList2">
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
    <soap12:operation soapAction="http://faxcoreserver/services/GetMessageList1" style="document" />
    <wsdl:input name="GetMessageList1">
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output name="GetMessageList1">
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
    <soap12:operation soapAction="http://faxcoreserver/services/GetMessageList" style="document" />
    <wsdl:input>
        <soap12:body use="literal" />

```

```

</wsdl:input>
<wsdl:output>
  <soap12:body use="literal" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="CreateUser">
  <soap12:operation soapAction="http://faxcoreserver/services/CreateNewUser1" style="document" />
  <wsdl:input name="CreateNewUser1">
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output name="CreateNewUser1">
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CreateUser">
  <soap12:operation soapAction="http://faxcoreserver/services/CreateNewUser2" style="document" />
  <wsdl:input name="CreateNewUser2">
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output name="CreateNewUser2">
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CreateUser">
  <soap12:operation soapAction="http://faxcoreserver/services/CreateNewUser3" style="document" />
  <wsdl:input name="CreateNewUser3">
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output name="CreateNewUser3">
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="UpdateUser">
  <soap12:operation soapAction="http://faxcoreserver/services/UpdateUser-basic" style="document" />
  <wsdl:input name="UpdateUser-basic">
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output name="UpdateUser-basic">
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="UpdateUser">
  <soap12:operation soapAction="http://faxcoreserver/services/UpdateUser-extra" style="document" />
  <wsdl:input name="UpdateUser-extra">
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output name="UpdateUser-extra">
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="UpdateUser">
  <soap12:operation soapAction="http://faxcoreserver/services/UpdateUser-advance" style="document" />

```

```

        <wsdl:input name="UpdateUser-advance">
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output name="UpdateUser-advance">
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="RemoveUser">
        <soap12:operation soapAction="http://faxcoreserver/services/RemoveUser" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="DeactivateUser">
        <soap12:operation soapAction="http://faxcoreserver/services/DeactivateUser" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="DeactivateUsers">
        <soap12:operation soapAction="http://faxcoreserver/services/DeactivateUsers" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="ActivateUser">
        <soap12:operation soapAction="http://faxcoreserver/services/ActivateUser" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="ActivateUsers">
        <soap12:operation soapAction="http://faxcoreserver/services/ActivateUsers" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>

```



```

<wsdl:operation name="CreatelnboundRoute">
  <soap12:operation soapAction="http://faxcoreserver/services/CreatelnboundRoute" style="document" />
  <wsdl:input>
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DownloadMessageByte">
  <soap12:operation soapAction="http://faxcoreserver/services/DownloadMessageByte1" style="document" />
  <wsdl:input name="DownloadMessageByte1">
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output name="DownloadMessageByte1">
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DownloadMessageByte">
  <soap12:operation soapAction="http://faxcoreserver/services/DownloadMessageInByte2" style="document" />
  <wsdl:input name="DownloadMessageInByte2">
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output name="DownloadMessageInByte2">
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DownloadMessageByte">
  <soap12:operation soapAction="http://faxcoreserver/services/DownloadMessageInByte3" style="document" />
  <wsdl:input name="DownloadMessageInByte3">
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output name="DownloadMessageInByte3">
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DownloadMessageByte">
  <soap12:operation soapAction="http://faxcoreserver/services/DownloadMessageInByte4" style="document" />
  <wsdl:input name="DownloadMessageInByte4">
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output name="DownloadMessageInByte4">
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="MarkMsgAsRead">
  <soap12:operation soapAction="http://faxcoreserver/services/MarkMsgAsRead" style="document" />
  <wsdl:input>
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal" />
  </wsdl:output>

```

```

        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="MarkMsgAsUnread">
        <soap12:operation soapAction="http://faxcoreserver/services/MarkMsgAsUnread" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="IsMessageRead">
        <soap12:operation soapAction="http://faxcoreserver/services/IsMessageRead" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetMsgFolders">
        <soap12:operation soapAction="http://faxcoreserver/services/GetMsgFolders" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetUserList">
        <soap12:operation soapAction="http://faxcoreserver/services/GetUserListWithDomainID" style="document" />
        <wsdl:input name="GetUserListWithDomainID">
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output name="GetUserListWithDomainID">
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetUserList">
        <soap12:operation soapAction="http://faxcoreserver/services/GetUserListWithDomainName" style="document" />
        <wsdl:input name="GetUserListWithDomainName">
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output name="GetUserListWithDomainName">
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetUserDetails">
        <soap12:operation soapAction="http://faxcoreserver/services/GetUserDetailsByUserID" style="document" />
        <wsdl:input name="GetUserDetailsByUserID">
            <soap12:body use="literal" />
        </wsdl:input>
    
```

```

        <wsdl:output name="GetUserDetailsByUserID">
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetUserDetails">
        <soap12:operation soapAction="http://faxcoreserver/services/GetUserDetailsByUserName" style="document" />
        <wsdl:input name="GetUserDetailsByUserName">
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output name="GetUserDetailsByUserName">
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SearchIBRouteRules">
        <soap12:operation soapAction="http://faxcoreserver/services/SearchIBRouteRules" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetDomainList">
        <soap12:operation soapAction="http://faxcoreserver/services/GetDomainList" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetUserFaxSetting">
        <soap12:operation soapAction="http://faxcoreserver/services/GetUserFaxSettingByUserID" style="document" />
        <wsdl:input name="GetUserFaxSettingByUserID">
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output name="GetUserFaxSettingByUserID">
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetUserFaxSetting">
        <soap12:operation soapAction="http://faxcoreserver/services/GetUserFaxSettingByUserName" style="document" />
        <wsdl:input name="GetUserFaxSettingByUserName">
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output name="GetUserFaxSettingByUserName">
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="MarkMsgAsDownloaded">
        <soap12:operation soapAction="http://faxcoreserver/services/MarkMsgAsDownloaded" style="document" />
        <wsdl:input>

```

```

        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
    <soap12:operation soapAction="http://faxcoreserver/services/GetMessageList4" style="document" />
    <wsdl:input name="GetMessageList4">
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output name="GetMessageList4">
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
    <soap12:operation soapAction="http://faxcoreserver/services/GetMessageList5" style="document" />
    <wsdl:input name="GetMessageList5">
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output name="GetMessageList5">
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
    <soap12:operation soapAction="http://faxcoreserver/services/GetMessageList6" style="document" />
    <wsdl:input name="GetMessageList6">
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output name="GetMessageList6">
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetMessageList">
    <soap12:operation soapAction="http://faxcoreserver/services/GetMessageList7" style="document" />
    <wsdl:input name="GetMessageList7">
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output name="GetMessageList7">
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="wfApi">
    <wsdl:port name="wfApiSoap" binding="tns:wfApiSoap">
        <soap:address location="http://co.faxcore.com/webservices/wfapi.asmx" />
    </wsdl:port>
    <wsdl:port name="wfApiSoap12" binding="tns:wfApiSoap12">
        <soap12:address location="http://co.faxcore.com/webservices/wfapi.asmx" />
    </wsdl:port>
</wsdl:service>
</wsdl:definitions>

```