



1 **COMMITTEE**
2 **DRAFT**

ISO/IEC CD
14543-5-22

3 Information technology –
4 Home Electronic System (HES) Architecture –
5 Intelligent Grouping and Resource Sharing for HES
6 Class 2 & Class 3 – Part 5-22: Application Profile –
7 File Profile

CONTENTS

8			
9	Foreword		7
10	Introduction		8
11	1 Scope.....		7
12	2 Normative References		7
13	3 Definitions and Abbreviations.....		7
14	3.1 Abbreviations		7
15	4 Conformance.....		8
16	5 System.....		8
17	5.1 Application Scenario		8
18	5.2 Design Criteria		8
19	5.2.1 Relationship between File Profile and IGRS.....		8
20	5.2.2 Relationship between File Profile and Transport Protocols		8
21	5.2.3 Relationship between File Profile and Existing File Sharing System		8
22	5.2.4 Relationship between File Profile and File Format.....		9
23	5.2.5 Device Supported by File Profile.....		9
24	5.3 Interaction Model		9
25	6 Interaction Flow of IGRS File Profile		10
26	6.1 Overview		10
27	6.2 Simple Interaction Flow.....		10
28	6.3 Complete Interaction Flow		11
29	7 FileServer		14
30	7.1 Overview		14
31	7.2 FileAccessManagement Service.....		14
32	7.2.1 Overview		14
33	7.2.2 FileAccessManagement Service Type.....		14
34	7.2.3 FileAccessManagement Service Attribute		14
35	7.2.4 Data Types of FileAccessManagement Service		15
36	7.2.5 Invocation Interface of FileAccessManagement Service		15
37	7.2.6 FileAccessManagement Service Error Codes Definition		23
38	7.3 FileConnectionManagement Service		24
39	7.3.1 Overview		24
40	7.3.2 FileConnectionManagement Service Type		24
41	7.3.3 FileConnectionManagement Service Attribute		24
42	7.3.4 FileConnectionManagement Service Data Type		24
43	7.3.5 Invocation Interface of FileConnectionManagement Service		25
44	7.3.6 FileConnectionManagement Service Error Codes Definition		26
45	8 FileClient.....		27
46	8.1 Overview		27
47	8.2 FileClient Engine		27
48	9 Session		27
49	9.1 Session Setup		27
50	9.1.1 Overview		27
51	9.1.2 Session Setup Condition		27
52	9.1.3 Session Setup Process		27
53	9.2 Session Termination		28
54	10 Service Invocation Message Format.....		32

55	10.1 Overview	32
56	10.2 Service Invocation Request Message	32
57	10.3 Service invocation response message	33
58	10.4 Content Directory Object Update Notification Message	34
59	10.5 Service Attribute Update Notification Message	35
60	Annex A (Normative Annex) Specific Description of Data Type Generation Rules	37
61	A.1 Type_ObjectId	37
62	A.1.1 Overview	37
63	A.1.2 IGRS File/Directory Object Identifier Format	37
64	A.2 ObjectType	37
65	A.3 Type_FilterRule	37
66	A.3.1 Syntactic Definition	37
67	A.3.2 Semantic Description	37
68	A.4 Type_SortRule	37
69	A.4.1 Syntactic Definition	37
70	A.4.2 Semantic Description	38
71	A.5 Content Directory Object Update Notification Message	38
72	A.5.1 Definition	38
73	A.6 Service Attribute Update Notification Message	38
74	A.6.1 Definition	38
75	Annex B (Normative Annex) Information device – IGRS – File Profile Message Format	39
76	B.1 Overview	39
77	B.1.1 Service Invocation Request Message	39
78	B.1.2 Service Invocation Response Message	40
79	B.1.3 File/Directory Object Update Notification Message	41
80	B.1.4 Service Attribute Update Notification Message	43
81	B.2 FileAccessManagement Service	44
82	B.2.1 Service Data Type	44
83	B.2.2 Service Invocation Message Format	46
84	B.3 FileConnectionManagement Service	55
85	B.3.1 Service Data Type	55
86	B.3.2 Service Invocation Message Format	56
87	Annex C (Normative Annex) File Profile – FileAccessManagement Service	
88	Description.wsdl	59
89	Annex D (Normative Annex) File Profile – FileConnectionManagement Service	
90	Description.wsdl	69
91	References	72
92		

93

FIGURES

94 Figure 1 - IGRS File Profile Interaction Model..... 9

95 Figure 2 - Simple Interaction Flow Model.....11

96 Figure 3 - Complete Interaction Flow of Shared File.....13

97

TABLES

98 Table 1 - FileAccessManagement Service Attribute List14

99 Table 2 - FileAccessManagement Service Data Type.....15

100 Table 3 – FileConnectionManagement Service Attribute.....24

101 Table 4 – FileConnectionManagement Service Data Type.....24

102 Table 5 - Service Invocation Request Message.....32

103 Table 6 - Service Invocation Response Message33

104 Table 7 - Content Directory Object Update Notification Message.....34

105 Table 8 - Service Attribute Update Notification Message35

106 Table B 1 - IGRS File Profile Service Invocation Request Message.....39

107 Table B 2 - Service Invocation Response Message40

108 Table B 3 - File/Directory Object Update Notification Message.....41

109 Table B 4 - Service Attribute Update Notification Message.....43

110

Foreword

111 ISO (International Organization for Standardization) and IEC (International Electrotechnical Commission) form the
112 specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the
113 development of International Standards through technical committees established by the respective organization to
114 deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual
115 interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take
116 part in the work.

- 117 1) In the field of information technology, ISO and IEC have established a joint technical committee ISO/IEC JTC 1.
- 118 2) The formal decisions or agreements of ISO and IEC on technical matters express, as nearly as possible, an
119 international consensus of opinion on the relevant subjects since each technical committee has representation
120 from all interested National Committees.
- 121 3) The documents produced have the form of recommendations for international use and are published in the form of
122 standards, technical reports, technical specifications or guides and they are accepted by the National Committees
123 in that sense.
- 124 4) In order to promote international unification, ISO National Members and IEC National Committees undertake to
125 apply ISO and IEC International Standards transparently to the maximum extent possible in their national and
126 regional standards. Any divergence between the IEC standard and the corresponding national or regional standard
127 shall be clearly indicated in the latter.
- 128 5) ISO and IEC provide no marking procedure to indicate its approval and cannot be rendered responsible for any
129 equipment declared to be in conformity with one of its standards.
- 130 6) Attention is drawn to the possibility that some of the elements of this standard may be the subject of patent rights.
131 The IEC shall not be held responsible for identifying any or all such patent rights.

132 ISO/IEC 14543-5-22 was prepared by Joint Technical Committee 1 of ISO/IEC, Information technology, Subcommittee
133 SC 25, Interconnection of Information Technology Equipment.

134 This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

135

Introduction

136 ISO/IEC 14543-5, Intelligent Grouping and Resource Sharing for HES (IGRS), is divided into six
137 parts:

138 ➤ **IGRS Part 5-1: Core Protocol**

- 139 • Specifies the TCP/IP protocol stack as the basis and the HTTP protocol as the message-
140 exchanging framework among devices.
- 141 • Defines a series of device and service interaction/invocation standards, including device
142 and service discovery protocol, device and service description, service invocation,
143 security mechanisms, etc.
- 144 • Specifies core protocols for a type of home network that supports streaming media and
145 other high-speed data transport within a home.

146 ➤ **IGRS Part 5-2: Application Profile**

- 147 • Based on the IGRS Core Protocol.
- 148 • Defines a device and service interaction mechanism, as well as application interfaces
149 used in IGRS Basic Applications.

150 ➤ **IGRS Part 5-3: Basic Application**

- 151 • Includes an IGRS basic application list.
- 152 • Defines a basic application framework.
- 153 • Addresses operation specifics (device grouping, service description template, etc.),
154 function definitions, and service invocation interfaces.

155 ➤ **IGRS Part 5-4: Device Validation**

- 156 • Defines a standard method to validate an IGRS-compliant device.

157 ➤ **IGRS Part 5-5: Device Types**

- 158 • Defines IGRS Device types used in IGRS applications.

159 ➤ **IGRS Part 5-6: Service Types**

- 160 • Defines basic service types used in IGRS applications.

161 **Information technology –**
 162 **Home Electronic System (HES) Architecture – Intelligent grouping and**
 163 **resource sharing for HES Class 2 & Class 3 – Part 2: Application profile –**
 164 **File Profile**

165 **1 Scope**

166 This part of ISO/IEC xxxx standard specifies the file data streaming application profile, device
 167 interaction flow model, the request and response messages in the device interaction process,
 168 and the service description format of the devices based on Intelligent Grouping and Resource
 169 Sharing (IGRS) Part 1: Core Protocol.

170 This document is applicable to resource sharing and service collaboration of file data stream
 171 among computers, consumer electronics, and communication devices in a Local Area Network
 172 (LAN) or Personal Area Network (PAN) environment, especially in a wireless dynamic network.

173 **2 Normative References**

174 The following standards and specifications contain provisions that, through reference in this text,
 175 constitute normative provisions of this standard. For all referenced articles with dates, the
 176 revised edition after that date shall not be applicable to this document. However, the reader is
 177 encouraged to consider whether to use the latest edition of such articles. For all referenced
 178 articles without dates, the latest edition of such articles shall be applicable to this document.

179 ISO/IEC 14543-5-1 Information Technology - Information Device Intelligent Grouping and
 180 Resource Sharing Specification Part 1: Core Protocol

181 **3 Definitions and Abbreviations**

182 **3.1 Abbreviations**

183 The following acronyms and abbreviations are used in this standard:

FAMS	FileAccessManagement Service
FC	FileClient
FCMS	FileConnectionManagement Service
FS	FileServer

184 **4 Conformance**

185 For conformance to this international standard the following applies:

- 186 • IGRS File profile interaction model shall conform to the system specification described in
187 Clause 5.
- 188 • File service interaction flow models shall conform to specifications described in Clause 6.
- 189 • The relevant File device (File Server, File Client) and service (FAMS, FCMS) shall conform to
190 the specification defined in Clause 7 and 8 respectively.
- 191 • File invocation session setup procedure and message formats shall conform to specifications
192 defined in Clause 9 and 10.

193 **5 System**

194 **5.1 Application Scenario**

195 In small office and home environment, there are many kinds of digital devices used, such as PC,
196 Notebook, PDA, DC, DV, MP3, MP4, mobile phone, Set-Top-Box, etc. Typically, the user may
197 need to browse, copy, delete, read and write data files stored in these devices over the network.

198 File profile is designed specifically for this objective. It setups a middleware layer between the
199 users and different devices. This layer hides the underlying networking protocols, devices,
200 operating systems and file operation modes while providing a set of commonly shared file access
201 APIs for the application program developers. The user can access the files in devices
202 transparently in a network environment, without having to consider the devices and operating
203 systems.

204 File profile enables the client to access shared files on servers from different locations, and it
205 includes the following functions:

- 206 a) Retrieve client file access identity verification;
- 207 b) Browse shared file/directory on server;
- 208 c) Manage uploading/downloading of file/directory;
- 209 d) Support simple management of file/directory in the shared directory of FileServer;
- 210 e) Support file/directory update event subscription on FileServer;
- 211 f) Support service update event subscription;
- 212 g) Out-of-band transport negotiation when uploading/downloading file.

213 **5.2 Design Criteria**

214 **5.2.1 Relationship between File Profile and IGRS**

215 File profile includes two parts: one specifies the functional interfaces of IGRS
216 FileAccessManagement service and FileConnectionManagement service; and the other specifies
217 the interaction logical flows of these service functional interfaces.

218 **5.2.2 Relationship between File Profile and Transport Protocols**

219 File profile is a higher layer application profile in the IGRS framework and does not rely on
220 specific transport protocols. Therefore it shall support any transport protocols such as FTP,
221 HTTP, etc.

222 **5.2.3 Relationship between File Profile and Existing File Sharing System**

223 File profile defines a set of standardized file sharing access interface based on IGRS network. It
224 does not put any restrictions on the backend implementation. File profile shall support either
225 private systems developed by users, or existing file sharing systems such as ActiveDirectory,
226 SMB, NFS, etc.

227 **5.2.4 Relationship between File Profile and File Format**

228 File profile is independent of any specific file format, which means it shall support interactions
229 with files of any arbitrary format.

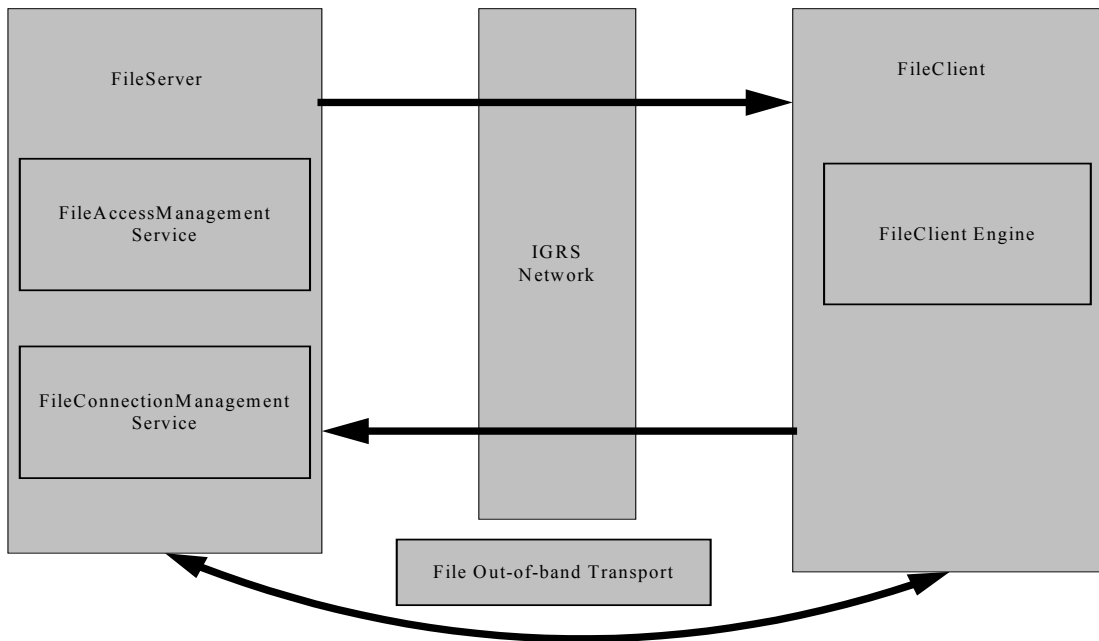
230 **5.2.5 Device Supported by File Profile**

231 Theoretically, file profile can run on any IGRS devices, such as PC, Notebook, PDA, DC, DV,
232 MP3, MP4, mobile phone, Set-Top-Box, etc.

233 **5.3 Interaction Model**

234 IGRS file profile is composed of two logical devices, FileServer and FileClient. FileServer is a
235 device that provides content and it includes two services: FileAccessManagement Service
236 (FAMS) and FileConnectionManagement Service (FCMS). FileClient is a device that accesses
237 content and it includes FileClient Engine. The user can access and manage the contents
238 published by FileServer from FileClient. All interaction functions shall be completed though the
239 services and engine running on these devices.

240 The interaction model covers two services, FileAccessManagement Service and
241 FileConnectionManagement Service, both of which run on FileServer. FileAccessManagement
242 Service is responsible for a majority of functions in the file interaction framework, including file
243 browsing management, file uploading/downloading management, client access rights
244 authentication, etc. FileConnectionManagement Service is used in out-of-band transport protocol
245 negotiation between FileServer and FileClient. FileClient Engine is a client application program
246 responsible for interactions with FileServer. IGRS file profile interaction model is shown as
247 Figure 1.



248
249

Figure 1 - IGRS File Profile Interaction Model

250 6 Interaction Flow of IGRS File Profile

251 6.1 Overview

252 Based on the basic rights setting of the shared files on FileServer, the interaction flow of IGRS
253 file profile can be classified into two categories: simple interaction flow (see Section 6.2) and
254 complete interaction flow (see Section 5.3). In simple interaction flow, only the required
255 interfaces of FileAccessManagement Service and FileConnectionManagement Service on
256 FileServer are applicable in this process. FileClient can browse, download shared file and
257 directory, or retrieve attributes of shared file and directory. In complete interaction flow, the
258 applicable interfaces include all required interfaces of FileAccessManagement Service and
259 FileConnectionManagement Service on FileServer and any optional interfaces except those
260 relate to subscription. FileClient can perform management operations on shared files and
261 directories, such as copy, move, delete, modify attributes, etc, or upload file to FileServer, in
262 addition to browse, search, download shared file and directory, or retrieve attributes of shared
263 file and directory. The detailed descriptions about services and interfaces in interaction flow can
264 be found in Annex B (File Profile Message Format).

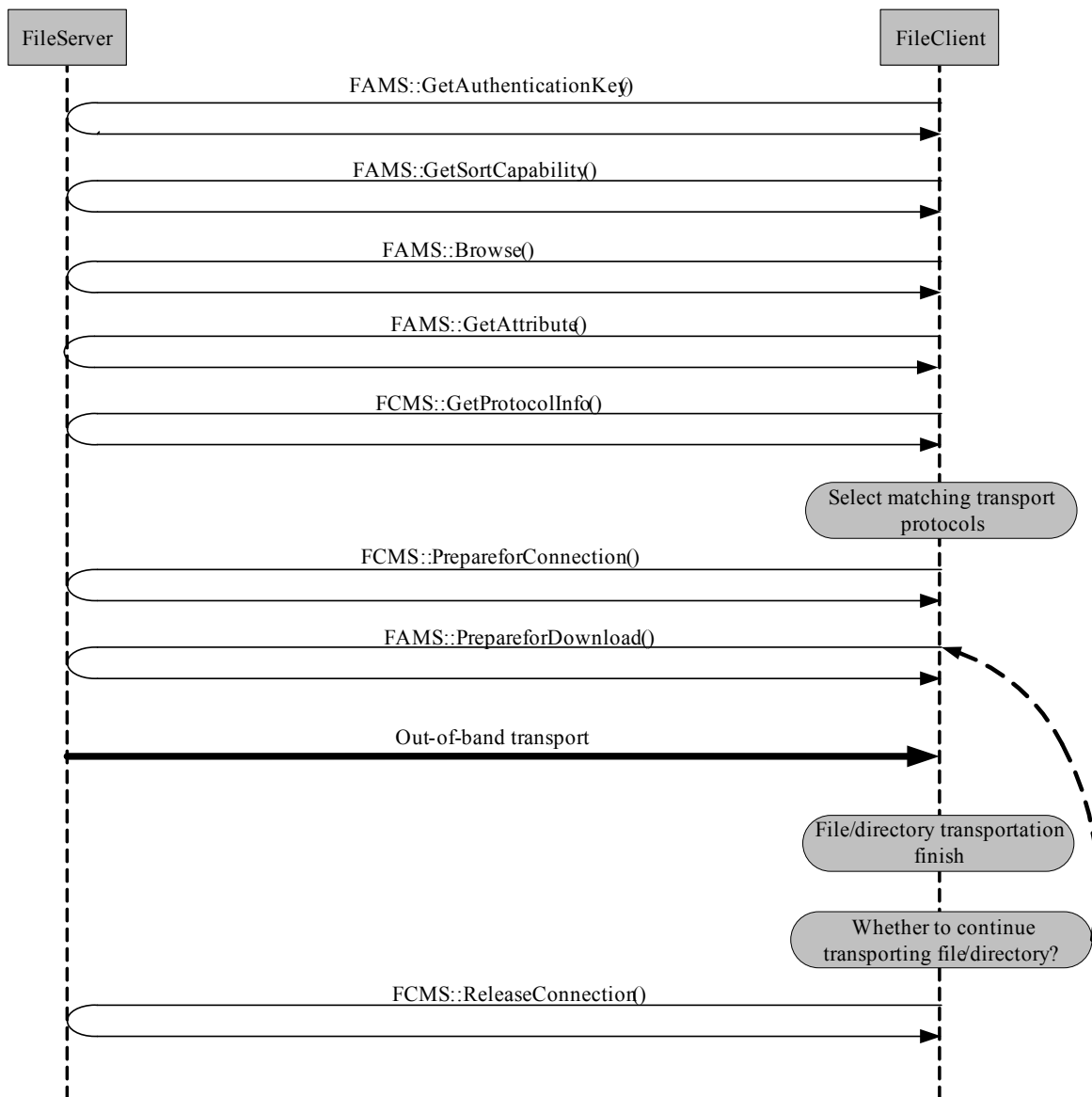
265 6.2 Simple Interaction Flow

266 In simple interaction flow (Figure 2), only the required interfaces of FileAccessManagement
267 Service and FileConnectionManagement Service on FileServer are applicable. FileClient shall
268 access shared file and directory in read-only mode, and is allowed to browse, download shared
269 file/directory or retrieve attributes of shared file/directory. However, it shall not be allowed to
270 modify shared file/directory.

271 In simple interaction flow, the invocation process of IGRS service interfaces between FileClient
272 and FileServer is described as follows:

- 273 a) IGRS service discovery: Through IGRS service discovery mechanism, FileClient
274 discovers FileAccessManagement Service and FileConnectionManagement Service on
275 FileServer;
- 276 b) Retrieve access right: FileClient retrieves authentication key to get access right to
277 shared file on FileServer through invoking FAMS::GetAuthenticationKey() interface of
278 FileAccessManagement Service on FileServer. FileClient can use any combination
279 information of DeviceID/DeviceName, Username/Password, and Third-party
280 authentication mode of the local device, to retrieve authentication key from FileServer;
- 281 c) Browse shared file/directory:
- 282 1) Retrieve sorting capability of FileServer: FileClient can retrieve shared file/directory
283 sorting capability of FileServer through invoking FAMS::GetSortCapability()
284 interface of FileAccessManagement Service on FileServer. For example, it can sort
285 according to file names or time of modification;
 - 286 2) Browse shared file/directory on FileServer: FileClient can browse all files and sub-
287 directory information in any specified shared directory through invoking
288 FAMS::Browse() interface of FileAccessManagement Service on FileServer;
 - 289 3) Retrieve attributes of shared file/directory: FileClient can retrieve attributes of any
290 specified shared file/directory through invoking FAMS::GetAttribute() interface of
291 FileAccessManagement Service on FileServer;
- 292 d) Setup connection with FileServer:
- 293 1) Retrieve transport protocols supported by FileServer: FileClient retrieves transport
294 protocols supported by FileServer, through invoking FCMS::GetProtocolInfo()
295 interface of FileConnectionManagement Service on FileServer;
 - 296 2) Select matching transport protocols: Using retrieved transport protocols supported
297 by FileServer according to 1) in step d), FileClient selects matching transport
298 protocols supported by the device;
 - 299 3) Connection preparation: FileClient notifies FileServer to prepare for connection
300 setup, and retrieve connection identifier for connection management to use in
301 subsequent interaction process, through invoking FCMS::PrepareforConnection()
302 interface of FileConnectionManagement Service on FileServer;

- 303 e) Download shared file/directory from FileServer: FileClient can setup the downloading of
- 304 shared file/directory, and retrieve the URI list of downloaded shared file/directory from
- 305 FileServer, through invoking FAMS::PrepareforDownload() interface of
- 306 FileAccessManagement Service on FileServer;
- 307 f) File/directory transport: FileClient and FileServer use out-of-band transport protocols
- 308 and the URI retrieved in step e) to transport data. When the specified file/directory has
- 309 been transported, it goes to back to step e) to transport other file/directory if required;
- 310 g) Close connection and release resource: When the file transport has been completed,
- 311 FileClient notifies FileServer to close the connection between them and release the
- 312 resource, through invoking FCMS::ReleaseConnection() interface of
- 313 FileConnectionManagement Service on FileServer.



314
315

Figure 2 - Simple Interaction Flow Model

316 6.3 Complete Interaction Flow

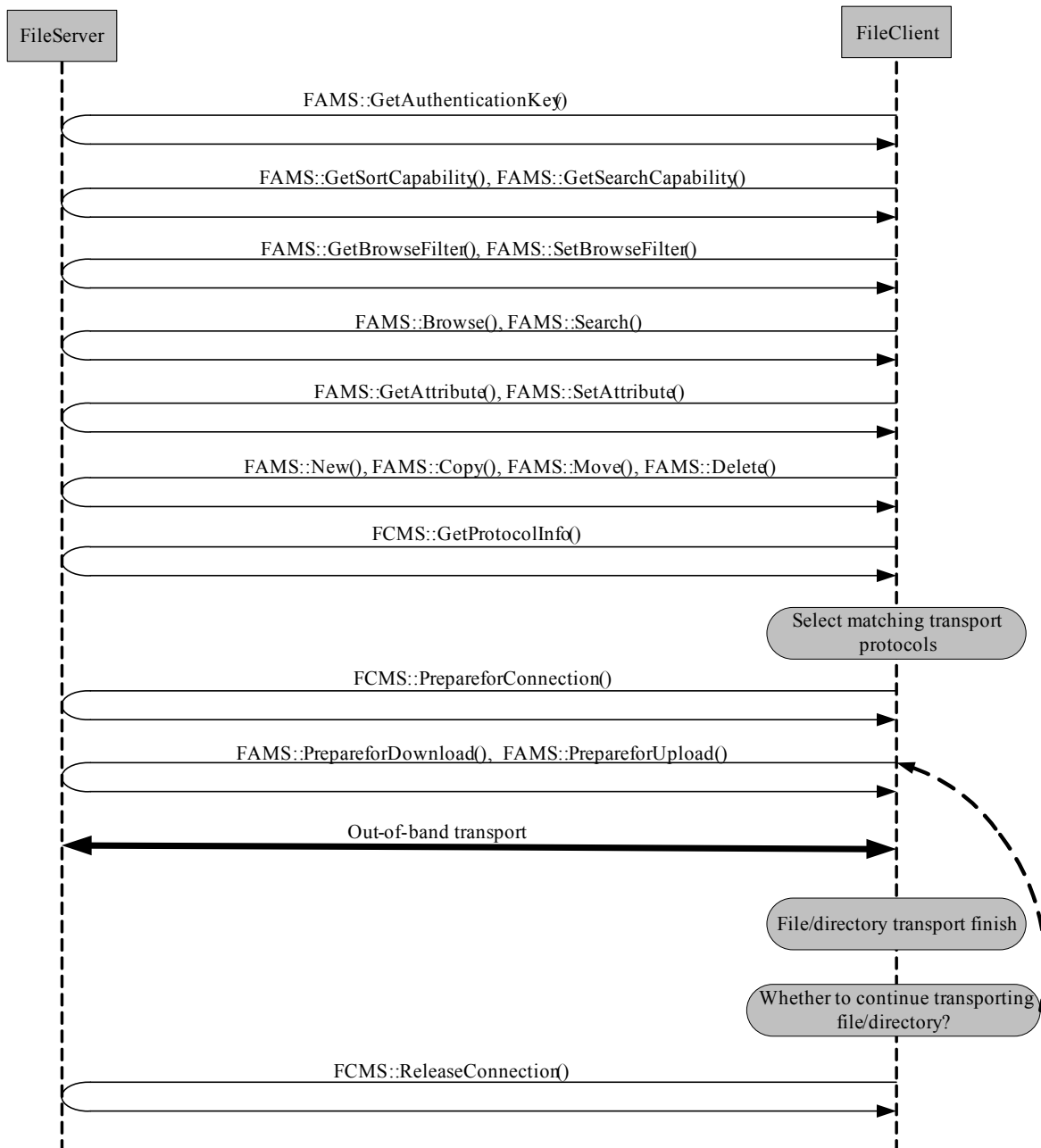
317 In complete interaction flow (Figure 3), the applicable interfaces include all required interfaces of
 318 FileAccessManagement Service and FileConnectionManagement Service on FileServer and any
 319 optional interfaces except those relate to subscription. FileClient can perform management

320 operations on shared files and directories, such as copy, move, delete, modify attributes, etc, or
321 upload file to FileServer, in addition to browse, search, download shared file and directory, or
322 retrieve attributes of shared file and directory.

323 In complete interaction flow, the invocation process of IGRS service interfaces between
324 FileClient and FileServer is described as follows:

- 325 a) IGRS service discovery: Through IGRS service discovery mechanism, FileClient
326 discovers FileAccessManagement Service and FileConnectionManagement Service on
327 FileServer;
- 328 b) Retrieve access right: FileClient retrieves authentication key to get access right to
329 shared file on FileServer through invoking FAMS::GetAuthenticationKey() interface of
330 FileAccessManagement Service on FileServer. FileClient can use any combination
331 information of DeviceID/DeviceName, UserName/Password, and Third-party
332 authentication mode of the local device, to retrieve authentication key from FileServer;
- 333 c) Browse and search shared file/directory on FileServer:
- 334 1) Retrieve sorting and searching capability of FileServer: FileClient can retrieve
335 shared file/directory sorting capability of FileServer through invoking
336 FAMS::GetSortCapability() interface of FileAccessManagement Service on
337 FileServer. For example, it can sort according to file names or time of modification.
338 It can also retrieve shared file/directory searching capability of FileServer through
339 invoking FAMS::GetSearchCapability () interface of FileAccessManagement Service
340 on FileServer. For example, it can search according to file names or time of
341 modification;
- 342 2) Retrieve or set browsing filter of shared file/directory: FileClient can retrieve or set
343 browsing file of shared file/directory through invoking FAMS::GetBrowseFilter() or
344 FAMS::SetBrowseFilter() interface of FileAccessManagement Service on
345 FileServer;
- 346 3) Browse or search shared file/directory on FileServer: FileClient can browse all files
347 and sub-directory information in any specified shared directory through invoking
348 FAMS::Browse() interface of FileAccessManagement Service on FileServer; It can
349 also search shared file/directory according to designated criteria through invoking
350 FAMS::Search() interface of FileAccessManagement Service on FileServer;
- 351 4) Retrieve or set attributes of shared file/directory: FileClient can retrieve attributes of
352 specified shared file/directory through invoking FAMS::GetAttribute() interface of
353 FileAccessManagement Service on FileServer. It can also set attributes of specified
354 shared file/directory through invoking FAMS::SetAttribute() interface of
355 FileAccessManagement Service on FileServer;
- 356 d) Manage shared file/directory: FileClient can create new shared file/directory in specified
357 shared directory on FileServer through invoking FAMS::New() interface of
358 FileAccessManagement Service on FileServer; or delete shared file/directory on
359 FileServer, through invoking FAMS::Delete() interface of FileAccessManagement
360 Service; or copy a shared file/directory on FileServer to a specified shared directory
361 through invoking FAMS::Copy() interface of FileAccessManagement Service; or move a
362 shared file/directory to specified shared directory through invoking FAMS::Move()
363 interface of FileAccessManagement Service;
- 364 e) Setup Connection with FileServer
- 365 1) Retrieve transport protocols supported by FileServer: FileClient retrieves transport
366 protocols supported by FileServer, through invoking FCMS::GetProtocolInfo()
367 interface of FileConnectionManagement Service on FileServer;
- 368 2) Select matching transport protocols: Using retrieved transport protocols supported
369 by FileServer according to 1) in step e), FileClient selects matching transport
370 protocols supported by the device;
- 371 3) Connection preparation: FileClient notifies FileServer to prepare for connection
372 setup, and retrieve connection identifier for connection management to use in
373 subsequent interaction process through invoking FCMS::PrepareforConnection()
374 interface of FileConnectionManagement Service in FileServer;
- 375 f) Download shared file/directory from FileServer or upload shared file/directory to
376 FileServer: FileClient can setup the downloading of shared file/directory, and retrieve

- 377 the URI list of shared file/directory to be downloaded through invoking
378 FAMS::PrepareforDownload() interface of FileAccessManagement Service on
379 FileServer; or setup file/directory uploading of shared directory, and retrieve the URI of
380 the shared directory through invoking FAMS::PrepareforUpload() interface of
381 FileAccessManagement Service;
382 g) File transport: FileClient and FileServer use out-of-band transport protocols and the URI
383 retrieved in step f) to download shared files from FileServer or upload client file to the
384 designated shared directory on FileServer. After the transporting of the specified file is
385 completed, it returns to step f) if required;
386 h) Close connection and release resource: When the file transport has been completed,
387 FileClient notifies FileServer to close the connection between them and release the
388 resource through invoking FCMS::ReleaseConnection() interface of
389 FileConnectionManagement Service on FileServer.



390
391

Figure 3 - Complete Interaction Flow of Shared File

392 7 FileServer**393 7.1 Overview**

394 FileServer is the device that provides file content, and it includes FileAccessManagement
395 Service (FAMS) and FileConnectionManagement Service (FCMS).

396 The functions of FileServer include:

- 397 a) Based on the authentication information provided by FileClient, FileServer assigns
398 corresponding authentication key to FileClient (recommended validity only during the
399 session). This key is matched with FileClient access right such that the key can be used
400 as the identifier for subsequent file access by the FileClient;
- 401 b) Provide to the network with the file/directory that can be browsed or searched by
402 FileClient;
- 403 c) Support simple management of file/directory in shared directory on FileServer.
- 404 d) Manage the uploading/downloading of file/directory;
- 405 e) Support FileServer file/directory update event subscription;
- 406 f) Support service update event subscription.

407 7.2 FileAccessManagement Service**408 7.2.1 Overview**

409 FileAccessManagement Service provides the following functions to FileClient:

- 410 a) Provide authentication for FileClient, and thus give corresponding file access right to
411 FileClient;
- 412 b) Allow FileClient to retrieve sorting/searching capability supported by FileServer;
- 413 c) Allow FileClient to browse content directory in the network provided by FileServer;
- 414 d) Allow FileClient to search specified file/directory;
- 415 e) Allow FileClient to retrieve and modify attributes of file/directory;
- 416 f) Allow FileClient to retrieve and set browsing filter;
- 417 g) Allow FileClient to subscribe to file/directory object update event;
- 418 h) Allow FileClient to subscribe to FileAccessManagement Service update event;
- 419 i) Support FileClient to upload and download the specified file/directory.

420 The detailed description of this service is found in Annex C. Profile - FileAccessManagement
421 Service Description.wsd).

422 7.2.2 FileAccessManagement Service Type

423 FileAccessManagement Service Type is defined as:

424 urn:IGRS:Service:ServiceType:FileAccessManagement:1

425 7.2.3 FileAccessManagement Service Attribute

426 Table 1 contains the File Access Management Service Attribute List.

427 **Table 1 - FileAccessManagement Service Attribute List**

Service Attribute Name	Data Type	Field Explanation
SortCaps	Type_SortCapability	File/directory object attribute sets sorted by searching or browsing results. Empty set denotes that FileAccessManagement Service does not support any file/directory object attributes sorting. If the returned set isn't empty, then any element in the set can be used as the sorting rule provided by FileAccessManagement Service.
SearchCaps	Type_SearchCapability	File/directory object attribute sets that set searching rule. Empty set denotes that FileAccessManagement Service does not support any file/directory object attributes sorting. If the returned set isn't empty, then any element in the set can be used as the searching rule provided by FileAccessManagement Service.

428

429 **7.2.4 Data Types of FileAccessManagement Service**

430 Table 2 contains the File Access Management Service Data Type.

431

Table 2 - FileAccessManagement Service Data Type

Data Type Name	Data Type	Field Explanation
Type_UserAuthenticationInfo	String	Authentication information provided by FileClient
Type_AuthenticationKey	String	Authentication key provided to FileClient by FileAccessManagement Service. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
Type_InstanceId	UnsignedInt	The session identifier between IGRS client and IGRS Service
Type_ObjectId	String	Only Object identifier that identifies file/directory object. See detailed definition of Type_ObjectId in Annex A.1
Type_SubscriptionId	String	Subscription identifier.
Type_ObjectIdList	String	Object identifier list.
Type_SortCapability	String	File/directory object attribute sets sorted by searching or browsing results. Empty set denotes that FileAccessManagement Service does not support any file/directory object attributes sorting. If the returned set isn't empty, then any element in the set can be used as the sorting rule provided by FileAccessManagement Service.
Type_SearchCapability	String	File/directory object attribute sets that set searching rule. Empty set denotes that FileAccessManagement Service does not support any file/directory object attributes sorting. If the returned set isn't empty, then any element in the set can be uses as the searching rule provided by FileAccessManagement Service.
Type_FilterRule	String	Represent searching rule when searching file/directory object. See detailed definition of Type_FilterRule in Annex A.3
Type_SortRule	String	Represent sorting rule when sorting file/directory attribute list. See detailed definition of Type_SortRule in Annex A.4
Type_ObjectAttribute	String	File/directory object attribute set, which includes ObjectType element to distinguish between file and directory object. See detailed definition of ObjectType in Annex A.2
Type_ServiceAttributeName	String	Service attribute name
Type_DeleteMode	String	Mode to delete file/directory (temporary/permanent delete)
Type_ObjectList	String	Result list of file/directory object.
Type_Count	Int	Object number
Type_ObjectURI	String	URI of Object on FileServer
Type_ObjectURITreeList	String	Tree list of the file/directory object list on FileServer, which includes file/directory object URI and attribute. See detailed definition in the description of Annex B.

432

433 **7.2.5 Invocation Interface of FileAccessManagement Service**434 **7.2.5.1 GetAuthenticationKey**

435 Function description: Retrieve assigned authentication key of FileServer based on the
 436 authentication information. The authentication key specifies the access rights of FileClient. The
 437 authentication key is recommended to be only in effect during a session. During the subsequent
 438 service invocations by FileClient, the authentication key is required to be used as the input
 439 parameter.

440 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
InstanceId	Input	Type_InstanceId	Session identifier between IGRS client and IGRS service.
UserAuthenticationInfo	Input	Type_UserAuthenticationInfo	Authentication information provided by FileClient. See detailed definition in Annex B.
AuthenticationKey	Output	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.

441 Return Value: Success: 0

442 Failure: E1, E2, E3 (see Error Codes in section 7.2.6)

443 **7.2.5.2 GetSortCapability**

444 Function description: Retrieve file/directory sorting capability supported by
445 FileAccessManagement Service on FileServer such that FileClient can use the file/directory
446 attribute set as the sorting rule, such as name, time of creation, etc.

447 Input/Output parameter:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
SortCaps	Output	Type_SortCapability	File/directory object attribute sets sorted by searching or browsing results. Empty set denotes that FileAccessManagement Service does not support any file/directory object attributes sorting. If the returned set isn't empty, then any element in the set can be used as the sorting rule provided by FileAccessManagement Service.

448 Return Value: Success: 0

449 Failure: E1, E2, E3, E9, E10 (see Error Codes in section 7.2.6)

450 **7.2.5.3 GetSearchCapability**

451 Function description: Retrieve file/directory searching capability supported by
452 FileAccessManagement Service on FileServer, such that FileClient can use the file/directory
453 attribute set as the searching rule, such as name, time of creation, etc.

454 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.

Parameter	Input/Output	Data Type	Field Explanation
SearchCaps	Output	Type_SearchCapability	File/directory object attribute sets that set searching rule. Empty set denotes that FileAccessManagement Service does not support any file/directory object attributes searching. If the returned set isn't empty, then any element in the set can be used as the searching rule provided by FileAccessManagement Service.

455 Return Value: Success: 0

456 Failure: E1, E2, E3, E9, E10 (see Error Codes in section 7.2.6)

457 7.2.5.4 Browse

458 Function description: Return to the lower level file/directory list of the specified directory on
459 FileServer, and get back the basic attributes of each object.

460 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
ObjectId	Input	Type_ObjectId	Only Object identifier to identify file/directory object.
BrowseFilter	Input	Type_FilterRule	Browsing rule of file object in the current directory. This parameter value has two instances: BrowseFilter value is empty. Then the preset filtering rule is used during the browsing process. The setting of the preset filtering rule is done by SetBrowseFilter interface. BrowseFilter value is not empty. Then this value is used as the browsing filtering rule of the service invocation. This value will not affect the value of the preset filtering rule.
StartOffset	Input	Type_Count	Offset between the starting item of the returned result and the first item in all file/directory lists that complies with the browsing rule. This value shall start from 0.
RequestedCount	Input	Type_Count	Maximum number of browsing items returned. -1 denotes that all browsing results are returned.
SortRule	Input	Type_SortRule	Sorting rule of the browsed directory object. This attribute can be retrieved through GetSortCapability interface.
Result	Output	Type_ObjectList	Result list of browsed file/directory file object.
NumberReturned	Output	Type_Count	Optional. The number of directory and file objects returned from the browsing results.
NumberTotalMatched	Output	Type_Count	Total number of file and directory objects that complies with the browsing rule.

461 Return Value: Success: 0

462 Failure: E1, E2, E3, E6, E7, E9, E10 (see Error Codes in section 7.2.6)

463 **7.2.5.5 GetAttribute**

464 Function description: Retrieve attribute set of the specified file/directory object on FileServer.

465 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
ObjectId	Input	Type_ObjectId	Only Object identifier to identify file/directory object.
ObjectAttribute	Output	Type_ObjectAttribute	Attribute set of file/directory object. See detailed definition in Annex B

466 Return Value: Success: 0

467 Failure: E1, E2, E3, E7, E9, E10 (see Error Codes in section 7.2.6)

468 **7.2.5.6 SetAttribute**

469 Function description: Set attribute of the specified file/directory object on FileServer.

470 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
ObjectId	Input	Type_ObjectId	Only Object identifier to identify file/directory object.
ObjectAttribute	Input	Type_ObjectAttribute	The attribute of object to be set. Only the setting of object name and initial access rights are required to be supported. See detailed definition in Annex B

471 Return Value: Success: 0

472 Failure: E1, E2, E3, E7, E9, E10 (see Error Codes in section 7.2.6)

473 **7.2.5.7 Search**

474 Function description: Search for the specified file/directory object on FileServer. That is to
 475 search for all of the object information in a specified directory (the recursive search of the object
 476 information in a sub-directory is included), and to support searching in multiple parallel
 477 directories.

478 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is

Parameter	Input/Output	Data Type	Field Explanation
			over.
ObjectIdList	Input	Type_ObjectIdList	Object identifier list within the search scope.
SearchRule	Input	Type_FilterRule	Searching rule of the file/directory object
StartOffset	Input	Type_Count	Offset between the starting item of the returned result and the first item in all file/directory lists that complies with the searching rule. This value shall start from 0.
RequestedCount	Input	Type_Count	Maximum number of searching items returned. -1 denotes that all searching results are returned.
Result	Output	Type_ObjectList	Result of file/directory object list returned according to the searching rule.
NumberReturned	Output	Type_Count	The number of file/directory objects returned from searching results.
NumberTotalMatched	Output	Type_Count	Optional, total number of file/directory objects that complies with the searching rule.

479 Return Value: Success: 0

480 Failure: E1, E2, E3, E6, E7, E9, E10 (see Error Codes in section 7.2.6)

481 7.2.5.8 GetBrowseFilter

482 Function description: Retrieve the current browsing filter rule. This is a preset browsing filter rule
483 used by FileAccessManagement Service.

484 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
BrowseFilter	Output	Type_FilterRule	Current browsing filter rule used.

485 Return Value: Success: 0

486 Failure: E1, E2, E3, E9, E10 (see Error Codes in section 7.2.6)

487 7.2.5.9 SetBrowseFilter

488 Function description: Set the browsing filter rule. Only SetBrowseFilter interface can change the
489 preset browsing filter rule, and the preset browsing filter rule shall remain the same until the next
490 invocation of SetBrowseFilter interface.

491 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
BrowseFilter	Input	Type_FilterRule	Browsing filter rule to be set.

492 Return Value: Success: 0
493 Failure: E1, E2, E3, E9, E10 (See Error Codes in section 7.2.6)

494 7.2.5.10 New

495 Function description: Create a new file/directory object in the directory object specified by
496 FileServer. The attribute of the object is specified by parameter ObjectAttribute.

497 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
ParentId	Input	Type_ObjectId	The object identifier of the parent directory object in which the file will be created.
ObjectAttribute	Input	Type_ObjectAttribute	The attribute of the new object, including object name and the initial access rights.
ObjectId	Output	Type_ObjectId	Only object identifier of the new object assigned by FileAccessManagement Service.

498 Return Value: Success: 0
499 Failure: E1, E2, E3, E7, E9, E10, E11 (See Error Codes in section 7.2.6)

500 7.2.5.11 Copy

501 Function description: Copy the specified file/directory object to the specified target path, while
502 reserving the source object. The file directory structure of the new directory shall follow the tree
503 structure of the source directory at the time of copying.

504 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
SourceObjectId	Input	Type_ObjectId	The object identifier of the source object assigned by FileAccessManagement Service.
DestParentId	Input	Type_ObjectId	The object identifier of the target directory.
DestObjectId	Output	Type_ObjectId	The object identifier of the new object assigned by FileAccessManagement Service.
DestObjectAttribute	Output	Type_ObjectAttribute	The object attribute of the new object.

505 Return Value: Success: 0
506 Failure: E1, E2, E3, E7, E9, E10, E11 (See Error Codes in section 7.2.6)

507 7.2.5.12 Move

508 Function description: Copy the specified file/directory to the specified target destination, and
509 delete the source object. The tree structure of the directory shall remain the same at the time of
510 moving.

511 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
SourceObjectId	Input	Type_ObjectId	The object identifier of the source object assigned by FileAccessManagement Service.
DestParentId	Input	Type_ObjectId	The object identifier of the target directory
DestObjectId	Output	Type_ObjectId	The object identifier of the new object assigned by FileAccessManagement Service.
DestObjectAttribute	Output	Type_ObjectAttribute	The object attribute of the new object.

512 Return Value: Success: 0

513 Failure: E1, E2, E3, E7, E9, E10, E11 (See Error Codes in section 7.2.6)

514 **7.2.5.13 Delete**515 Function description: Delete the specified file/directory object on FileServer, including the sub-
516 directories and files of this directory.

517 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
ObjectId	Input	Type_ObjectId	The object identifier of the object to be deleted.
DeleteMode	Input	Type_DeleteMode	The mode of object deletion (temporary/permanent delete)

518 Return Value: Success: 0

519 Failure: E1, E2, E3, E7, E9, E10, E11 (See Error Codes in section 7.2.6)

520 **7.2.5.14 PrepareforDownload**521 Function description: Prepare for file download, and pre-detect the feasibility of the file/ directory
522 download. If download is allowed by FileServer, then the URI of the source to be transported
523 shall be returned to the client, so that the FileClient application can download the file/directory
524 through out-of-band data transport; or else corresponding error codes shall be returned.

525 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
SourceObjectIdList	Input	Type_ObjectIdList	The object identifier list of the file/directory object to be

Parameter	Input/Output	Data Type	Field Explanation
			downloaded.
SourceObjectURITreeList	Output	Type_ObjectURITreeList	The tree list of the downloaded file/directory object list on FileServer, including the URI and the attributes of the file/directory object.

526 Return Value: Success: 0

527 Failure: E1, E2, E3, E7, E9, E10 (See Error Codes in section 7.2.6)

528 7.2.5.15 PrepareforUpload

529 Function description: Verify the file/directory objects to be uploaded by client, including whether
530 the storage space of the server is sufficient and has object with the same object name, etc. If
531 upload is allowed by FileServer, the URI of the target object to be transported shall be returned
532 to the client, so that the FileClient application can upload the file/directory through out-of-band
533 data transport; or else corresponding error codes shall be returned.

534 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
ObjectAttribute	Input	Type_ObjectAttribute	The attribute of the uploaded file/directory object.
DestParentId	Input	Type_ObjectId	The parent directory of the uploading object on FileServer.
DestParentURI	Output	Type_ObjectURI	The URI of the object on FileServer. This parameter denotes the URI of the destination directory of the uploading objects on FileServer.

535 Return Value: Success: 0

536 Failure: E1, E2, E3, E7, E8, E9, E10, E11 (See Error Codes in section 7.2.6)

537 7.2.5.16 SubscribeObjectChange

538 Function description: Subscribe to the update event of file/directory objects on FileServer. If an
539 object is changed, then FileAccessManagement Service shall notify FileClient of the event.

540 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
ObjectId	Input	Type_ObjectId	The object identifier of the shared object to be subscribed.

541 Return Value: Success: 0

542 Failure: E1, E2, E3, E4, E5, E7, E9, E10 (See Error Codes in section 7.2.6)

543 7.2.5.17 UnsubscribeObjectChange

544 Function description: Unsubscribe to the update event of file/directory objects on FileServer.

545 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
ObjectId	Input	Type_ObjectId	The object identifier of the shared object to be unsubscribed.

546 Return Value: Success: 0

547 Failure: E1, E2, E3, E4, E5, E7, E9, E10 (See Error Codes in section 7.2.6)

548 7.2.5.18 SubscribeServiceAttribute

549 Function description: Subscribe to the update event of service attributes. If an attribute changes,
550 FileAccessManagement Service shall notify FileClient.

551 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
ServiceAttributeName	Input	Type_ServiceAttributeName	The name of the service attributes to be subscribed.
SubscriptionId	Output	Type_SubscriptionId	The subscription identifier

552 Return Value: Success: 0

553 Failure: E1, E2, E3, E4, E5, E7, E9, E10 (See Error Codes in section 7.2.6)

554 7.2.5.19 UnsubscribeServiceAttribute

555 Function description: Unsubscribe to the update event of the service attribute.

556 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
AuthenticationKey	Input	Type_AuthenticationKey	FileAccessManagement Service assigns authentication key to FileClient. The authentication key specifies the access rights of FileClient. It is recommended that the key is only valid during this session. It is disabled when the session is over.
SubscriptionId	Input	Type_SubscriptionId	The subscription identifier to be unsubscribed.

557 Return Value: Success: 0

558 Failure: E1, E2, E3, E4, E5, E7, E9, E10 (See Error Codes in section 7.2.6)

559 7.2.6 FileAccessManagement Service Error Codes Definition

Error Code No.	Error Code Name	Value	Field Explanation
0	RETURN_SUCCESS	0	Success
E1	RETURN_FAILED	1	Failure
E2	RETURN_INVALIDPARA	2	Invalid parameter

Error Code No.	Error Code Name	Value	Field Explanation
E3	RETURN_ERRORFORMATPARA	3	Parameter format error
E4	RETURN_INVALIDSUBSCRIBEID	4	Invalid subscription identifier
E5	RETURN_NOTALLOWEDTOSUBSCRIBE	5	Subscription not allowed
E6	RETURN_OFFSETOVERFLOW	6	Offset overflow
E7	RETURN_OBJECTINEXISTENCE	7	Object not exist
E8	RETURN_SPACENOTENOUGH	10	Not enough space
E9	RETURN_INVALIDAUTHENTICATIONKEY	11	Invalid authentication key
E10	RETURN_RIGHTNOTMATCHED	12	Rights not matched
E11	RETURN_OBJECTNAMEEXISTED	13	Object name already exist
E12	RETURN_INTERFACEINEXISTENCE	14	Interface not exist

560

561 **7.3 FileConnectionManagement Service**562 **7.3.1 Overview**

563 FileConnectionManagement Service is used to create and manage the connection between
564 FileServer and FileClient. FileServer can support and manage multiple active connections at any
565 one time through FileConnectionManagement Service. The detailed service description is shown
566 in Annex D (File Profile - FileConnectionManagement Service Description.wsdl)

567 **7.3.2 FileConnectionManagement Service Type**

568 FileConnectionManagement Service Type is defined as:
569 urn:IGRS:Service:ServiceType:FileConnectionManagement:1

570 **7.3.3 FileConnectionManagement Service Attribute**

571 Table 3 contains the File Connection Management Service Attribute List.

572 **Table 3 – FileConnectionManagement Service Attribute**

Service Attribute Name	Data Type	Field Explanation
ProtocolInfoList	Type_ProtocolInfo List	The transport protocol list supported by FileServer.
IPList	Type_IPList	The valid IP address list of FileServer.

573

574 **7.3.4 FileConnectionManagement Service Data Type**

575 Table 4 contains the File Connection Management Service Data Type.

576 **Table 4 – FileConnectionManagement Service Data Type**

Data Type Name	Data Type	Field Explanation
Type_ProtocolInfo	String	Transport protocol
Type_ProtocolInfoList	String	Transport protocol list supported by FileServer.
Type_IPList	String	The valid IP address list of FileServer.
Type_ConnectionId	UnsignedInt	The connection identifier managed by FileConnectionManagement Service on FileServer.
Type_SubscriptionId	UnsignedInt	The subscription identifier
Type_ServiceAttributeName	String	The name of the service attribute to be subscribed.

577

578 **7.3.5 Invocation Interface of FileConnectionManagement Service**579 **7.3.5.1 GetProtocollInfo**

580 Function description: Retrieve the transport protocol supported by FileServer, including the
581 transport control protocol name, network port, the content format and other protocol information,
582 as well as the valid IP address list of FileServer.

583 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
ProtocollInfoList	Output	Type_ProtocollInfoList	The transport protocol list supported by FileServer.
IP List	Output	Type_IPList	The valid IP address list of FileServer. The format is shown in Annex B

584 Return Value: Success: 0

585 Failure: E1, E2, E3, E6, E8, E9 (See Error Codes in section 7.3.6)

586 **7.3.5.2 PrepareforConnection**

587 Function description: Retrieve the ConnectionId used to send and receive data.

588 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
RemoteProtocollInfo	Input	Type_ProtocollInfo	The transport protocol
ConnectionId	Output	Type_ConnectionId	The connection identifier managed by FileConnectionManagement service on FileServer.

589 Return Value: Success: 0

590 Failure: E1, E2, E3, E6, E7, E8, E9 (See Error Codes in section 7.3.6)

591 **7.3.5.3 ReleaseConnection**

592 Function description: Notify the device to close the connection corresponding to the
593 ConnectionId, and release resource.

594 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
ConnectionId	Input	Type_ConnectionId	The connection identifier managed by FileConnectionManagement service on FileServer.

595 Return Value: Success: 0

596 Failure: E1, E2, E3, E6, E7 (See Error Codes in section 7.3.6)

597 **7.3.5.4 GetActiveConnectionIdList**

598 Function description: Retrieve the current active connection identifier list on FileServer.

599 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
ConnectionIdList	Output	Type_ConnectionIdList	The current active connection identifier list on the current device.

600 Return Value: Success: 0

601 Failure: E1, E2, E3, E6, E7, E8, E9 (See Error Codes in section 7.3.6)

602 **7.3.5.5 GetCurrentConnectionInfo**

603 Function description: Retrieve the connection information of FileServer that corresponds to the
604 connection identifier.

605 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
ConnectionId	Input	Type_ConnectionId	The connection identifier on the current device.
ProtocolInfo	Output	Type_ProtocolInfo	The transport and control protocol used for file transport
ConnectionState	Output	Type_ConnectionState	The current connection state

606 Return Value: Success: 0

607 Failure: E1, E2, E3, E6, E7, E8, E9 (See Error Codes in section 7.3.6)

608 **7.3.5.6 SubscribeServiceAttribute**

609 Function description: Subscribe to the update event of service attributes. If an attribute is
610 changed, FileConnectionManagement Service shall notify the FileClient.

611 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
ServiceAttributeName	Input	Type_ServiceAttributeName	The name of the service attribute to be subscribed.
SubscriptionId	Output	Type_SubscriptionId	The subscription identifier

612 Return Value: Success: 0

613 Failure: E1, E2, E3, E4, E5, E6, E7, E8, E9 (See Error Codes in section 7.3.6)

614 **7.3.5.7 UnsubscribeServiceAttribute**

615 Function description: Unsubscribe to the update event of file/directory objects on FileServer.

616 Input/Output parameters:

Parameter	Input/Output	Data Type	Field Explanation
SubscriptionId	Input	Type_SubscriptionId	The subscription identifier of the object to be unsubscribed.

617 Return Value: Success: 0

618 Failure: E1, E3, E4, E5, E6, E7, E8, E9 (See Error Codes in section 7.3.6)

619 **7.3.6 FileConnectionManagement Service Error Codes Definition**

Error Code No.	Error Code Name	Value	Field Explanation
0	RETURN_SUCCESS	0	Success
E1	RETURN_FAILED	1	Failure
E2	RETURN_INVALIDPARA	2	Invalid parameter
E3	RETURN_ERRORFORMATPARA	3	Parameter format error
E4	RETURN_INVALIDSUBSCRIBEID	4	Invalid subscription identifier
E5	RETURN_NOTALLOWEDTOSUBSCRIBE	5	Subscription not allowed
E6	RETURN_CONNECTIONDISABLED	8	Connection disabled
E7	RETURN_INVALIDCONNECTIONID	9	Invalid connection identifier
E8	RETURN_INVALIDAUTHENTICATIONKEY	11	Invalid Authentication Key
E9	RETURN_RIGHTNOTMATCHED	12	Rights not matched
E10	RETURN_INTERFACEINEXISTENCE	14	Interface not exist

620 **8 FileClient**

621 **8.1 Overview**

622 FileClient can access FileServer. After retrieving the authentication key from FileServer,
623 FileClient can send service request messages to retrieve FileAccessManagement Service
624 function provided by FileServer. The functions of FileClient are as follows:

- 625 a) Browse the shared file/directory provided by FileServer;
- 626 b) Manage the file/directory on FileServer, including modify the file/directory attribute, create,
627 copy and move file/directory, etc.;
- 628 c) Upload and download files from FileServer;
- 629 d) Synchronize with FileServer by subscribing to the service attributes update event on
630 FileServer.

631 **8.2 FileClient Engine**

632 FileClient engine is an application program that is used to interact with FileServer. It includes
633 functions such as providing authentication information, user interface, invoking various interfaces
634 on FileServer to retrieve the functions provided by FileServer etc.

635 **9 Session**

636 **9.1 Session Setup**

637 **9.1.1 Overview**

638 After discovering a service on the target IGRS Device through the service discovery mechanism,
639 an IGRS Client may establish a supporting environment for subsequent service access through
640 the session setup mechanism based on a Device Pipe.

641 An IGRS Service may attain service access control by gathering the authorised user and device
642 access list of this service. IGRS Services should set up the maximum number of concurrent
643 access.

644 IGRS Device relationships are determined by whether devices are located in the same device
645 group, whether the IGRS Client device is the trusted device of the IGRS Service device, and
646 whether the IGRS Client device is the specified trusted device of the IGRS Service device.

647 During the session setup process, the IGRS Device providing an IGRS Service shall acquire
648 information about the IGRS Client device, such as the user ID and client authentication
649 information. Based on this information, the IGRS Client access right shall be authenticated.
650 Meanwhile, concurrent control shall be realised according to the concurrent support capabilities
651 of IGRS Services. Only one session shall exist between one IGRS Client and one IGRS Service
652 at the same time. After a session is successfully established, an IGRS Client may invoke IGRS
653 Services including function invocation, data query, data retrieval, etc. through invoking the
654 interfaces of the IGRS Services.

655 **9.1.2 Session Setup Condition**

656 IGRS Client can obtain the security mechanism description of this service through corresponding
657 IGRS service advertisement or the detailed service description document. If an IGRS Client
658 wants to invoke this service, it shall first request session setup with this service according to the
659 security mechanism description requirement.

660 **9.1.3 Session Setup Process**

661 Detailed session setup process is shown in section 10.5 of ISO/IEC 14543-5-1 Information
662 Technology - Information Device Intelligent Grouping and Resource Sharing Specification Part 1:
663 Core Protocol.

664 9.2 Session Termination

665 When an IGRS Client finishes invoking IGRS Services, the IGRS Client shall terminate the
666 session with the IGRS Service (please refer to section 10.5 of ISO/IEC 14543-5-1 Information
667 Technology - Information Device Intelligent Grouping and Resource Sharing Specification Part 1:
668 Core Protocol).

669 **10 Service Invocation Message Format**670 **10.1 Overview**

671 IGRS service invocation message format shall conform to the service invocation requirements
672 specified in ISO/IEC 14543-5-1.

673 **10.2 Service Invocation Request Message**

674 The Client can initiate “request/response” service invocation based on the session with the target
675 service. The definition of the service invocation request message is shown in Table 5:

676

Table 5 - Service Invocation Request Message

Message	Field Explanation
M-POST /IGRS HTTP/1.1	HTTP EXTENDED COMMAND LINE
Host: Target Host IP address:Port	Required field
01-IGRSVersion: IGRS/1.0	Required field, IGRS version number
01-IGRSMessageType:InvokeServiceRequest	Required field, content shall be this
01-SourceDeviceId: requesting device identifier	Required field, and type is URI, defined in 8.1.2 of ISO/IEC 14543-5-1
01-TargetDeviceId: target device identifier	Required field, type is URI, defined in 8.1.2 of ISO/IEC 14543-5-1
Content-Length: Message body length	Required field
Content-type:text/xml; charset=utf-8	Required field
MAN:"http://www.igrs.org/spec1.0";ns=01	Required field
MAN:"http://schemas.xmlsoap.org/soap/envelope/";ns=02	Required field
02-SoapAction:"IGRS-InvokeService-Request"	Required field
	Shall be empty
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">	Required field
<SOAP-ENV:Body>	Required field
<Session xmlns="http://www.igrs.org/spec1.0"request="NeedResponse">	Required field, request attribute indicates that the invocation message needs response
<SourceClientId> <i>Source client identifier</i> </SourceClientId>	Required field, type is 32-bit UnsignedInt
<TargetServiceId> <i>Target service identifier</i> </TargetServiceId>	Required field, type is 32-bit UnsignedInt
<SequenceId> <i>Invocation request sequence Id</i> </SequenceId>	Required field, type is 32-bit UnsignedInt
<!--Here is the start of the specific invocation request message-->	
<Invocation interface name>	Required field
<Input parameter name> <i>Input parameter value</i> </Input parameter name>	The service invocation interface that meets the requirements of ISO/IEC 14543-5-1.
.....	
<Input parameter name> <i>Input parameter value</i> </Input parameter name>	The service invocation interface that meets the requirements of ISO/IEC 14543-5-1.

Message	Field Explanation
</Invocation interface name>	Required field
<!--Here is the end of the specific invocation request message-->	
</Session>	Required field
</SOAP-ENV:Body>	Required field
</SOAP-ENV:Envelope>	Required field
Note: Italics indicate where content is to be inserted; all other text in message definitions is fixed.	

677

678 **10.3 Service invocation response message**

679 When an IGRS service receives an invocation request with response requirement, it should
680 return invocation response message. Service invocation response message is sent to the client
681 who is invoking the service based on the same session. Message definition is shown in Table 6:

682

Table 6 - Service Invocation Response Message

Message	Field Explanation
HTTP/1.1 200 OK	HTTP Command line
Ext:	Required field
Cache-control:no-cache="Ext"	Required field
MAN:"http://www.igrs.org/spec1.0";ns=01	Required field
01-IGRSVersion: IGRS/1.0	Required field, IGRS Version number
01-IGRSMessageType:InvokeServiceResponse	Required field, content shall be this
01-TargetDeviceId: target device identifier	Required field, type is URI
01-SourceDeviceId: requesting device identifier	Required field, type is URI
01-AcknowledgedId: Service pipe message response sequence Id	Required field, Type is 32 bit UnsignedInt
Content-Length: Message body length	Required field
Content-type:text/xml; charset=utf-8	Required field
MAN:"http://schemas.xmlsoap.org/soap/envelope/";ns=02	Required field
02-SoapAction:"IGRS-InvokeService-Response"	Required field
	Shall be empty
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">	Required field
<SOAP-ENV:Body>	Required field
<Session xmlns="http://www.igrs.org/spec1.0">	Required field
<SourceServiceId> <i>Source service identifier</i> </SourceServiceId>	Required field, Type is 32 bit UnsignedInt
<TargetClientId> <i>Target client identifier</i> </TargetClientId>	Required field, Type is 32 bit UnsignedInt
<AcknowledgedId> <i>Invocation response message sequence Id</i> </AcknowledgedId>	Required field, Type is 32 bit UnsignedInt, It's the same as the invocation request SequenceId of the request message.
<ReturnCode> <i>Invocation response status code</i> </ReturnCode>	Required field, defined in Section 11 of ISO/IEC 14543-5-1

Message	Field Explanation
<!--Here is the specific invocation response message-->	
<FileReturnCode> <i>File universal error return code</i> </FileReturnCode>	Optional field, Indicate the error response message independent of the service invocation, such as the service invocation is not supported. There shall be one and only one return code returned between FileReturnCode and ReturnCode.
<Invocation interface name>	Required field
<ReturnCode> <i>Service invocation response status code</i> </ReturnCode>	Optional field. Conform to service invocation return code definition. There shall be one and only one return code returned between FileReturnCode and ReturnCode.
< Output parameter> <i>Output parameter value</i> </ Output parameter>	Shall meet service interface definition requirement
.....	
< Output parameter> <i>Output parameter value</i> </ Output parameter>	Shall meet service interface definition requirement.
</ Invocation interface name>	Required field
<!--Here is the specific invocation response message-->	
</Session>	Required field
</SOAP-ENV:Body>	Required field
</SOAP-ENV:Envelope>	Required field

683

684 **10.4 Content Directory Object Update Notification Message**

685 FileAccessManagement service can notify client based on session setup with client when the
686 content directory object changes. This message doesn't require any response. If content
687 directory object update event is subscribed to a multi-directory structure, the update message
688 should be sent only to the lowest directory object. Message definition is shown in Table 7:

689

Table 7 - Content Directory Object Update Notification Message

Message	Field Explanation
M-NOTIFY //IGRS HTTP/1.1	HTTP EXTENDED COMMAND LINE
Host:Target host IP address:Port	Required field
MAN:"http://www.igrs.org/spec1.0";ns=01	Required field
01-IGRSVersion: IGRS/1.0	Required field, IGRS Version number
01-IGRSMessageType: FamsServiceNotifyClient	Required field, content shall be this
01-SourceDeviceId: Source device identifier	Required field, type is URI
01-TargetDeviceId: Target device identifier	Required field, type is URI
01-Sequenceld: Device pipe message sequence Id	Required field, type is 32 bit UnsignedInt (0 is reserved)
Content-Length:message body length	Required field
Content-type:text/xml; charset=utf-8	Required field
MAN:"http://schemas.xmlsoap.org/soap/envelope/";ns=02	Required field
02-SoapAction:"IGRS-FamsServiceToClient-Notify"	Required field
	Shall be empty

Message	Field Explanation
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">	Required field
<SOAP-ENV:Body>	Required field
<Session xmlns="http://www.igrs.org/spec1.0"request="NoResponse">	Required field, "request" attribute indicates that no response is needed
<SourceServiceId> <i>Source Service identifier</i> </SourceServiceId>	Required field, Type is 32 bit UnsignedInt (0 is reserved)
<TargetClientId> <i>Target client identifier</i> </TargetClientId>	Required field, Type is 32 bit UnsignedInt (0 is reserved)
<SequenceId> <i>Invocation request sequence id</i> </SequenceId>	Required field, Type is 32 bit UnsignedInt (0 is reserved)
<!--Here is the specific notification message-->	
<FamsUpdateNotification>	Required field
<SubscribeObjectId> <i>The subscribed file/directory object identifier</i> </SubscribeObjectId>	Required field, refer to the Type_ObjectId data type definition in Annex C
<ParentDirectoryId> <i>ID of parent directory in which the update event occurred</i> </ParentDirectoryId>	Required field, refer to Type_ObjectId data type definition in Annex C
<EventObjectIdList> <i>File/Directory object ID list in which update event occurred</i> </EventObjectIdList>	Optional field, refer to Type_ObjectIdList data type definition in Annex B, the updated object identifier list.
<EventType> <i>File/directory object update event type</i> </EventType>	Required field, Type is String, The options are: "ChildrenAdded", "ChildrenDeleted", "NameChanged", "AccessRightChanged", "ContentChanged", "SelfDeleted"
</FamsUpdateNotification>	Required field
</Session>	Required field
</SOAP-ENV:Body>	Required field
</SOAP-ENV:Envelope>	Required field

690

691 **10.5 Service Attribute Update Notification Message**

692 Service can notify the Client based on session setup with Client when service attribute changes.

693 This message doesn't require response. Message Definition is shown in Table 8:

694

Table 8 - Service Attribute Update Notification Message

Message	Field Explanation
M-NOTIFY //IGRS HTTP/1.1	HTTP EXTENDED COMMAND LINE
Host:Target host IP address:Port	Required field
MAN:"http://www.igrs.org/spec1.0";ns=01	Required field
01-IGRSVersion: IGRS/1.0	Required field, IGRS Version number
01-IGRSMessageType:ServiceNotifyClient	Required field, content shall be this
01-SourceDeviceId: Source Device identifier	Required field, type is URI
01-TargetDeviceId: Target Device Identifier	Required field, type is URI
01-SequenceId: Device pipe message sequence Id	Required field, type is 32 bit

Message	Field Explanation
	UnsignedInt (0 is reserved)
Content-Length:message body length	Required field
Content-type:text/xml; charset=utf-8	Required field
MAN:"http://schemas.xmlsoap.org/soap/envelope/";ns=02	Required field
02-SoapAction:"IGRS-ServiceToClient-Notify"	Required field
	Shall be empty
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">	Required field
<SOAP-ENV:Body>	Required field
<Session xmlns="http://www.igrs.org/spec1.0"request="NoResponse">	Required field, "request" attribute indicates that no response is needed
<SourceServiceId> <i>Source Service Identifier</i> </SourceServiceId>	Required field, type is 32 bit UnsignedInt (0 is reserved)
<TargetClientId> <i>Target Client Identifier</i> </TargetClientId>	Required field, type is 32 bit UnsignedInt (0 is reserved)
<SequenceId> <i>Invocation request sequence Id</i> </SequenceId>	Required field, type is 32 bit UnsignedInt (0 is reserved)
<!--Here is the specific notification message-->	
<ServiceAttributeUpdateNotification>	Required field
<SubscriptionId> <i>Subscription Identifier</i> </SubscriptionId>	Required field, String, Subscription Identifier
<ServiceAttributeName> <i>The updating Service attribute name</i> </ServiceAttributeName>	Required field, String, the updating Service attribute name
<ServiceAttributeValue > <i>Value of service attribute in which update event occurred</i> </ ServiceAttributeValue >	Required field. String.
</ServiceAttributeUpdateNotification>	Required field
</Session>	Required field
</SOAP-ENV:Body>	Required field
</SOAP-ENV:Envelope>	Required field

696
697
698

Annex A (Normative Annex) Specific Description of Data Type Generation Rules

699 **A.1 Type_ObjectId**700 **A.1.1 Overview**

701 File/directory object identifier Type_ObjectId is a global unique ID. Each file/directory object
702 identifier may uniquely correspond to a directory or file.

703 **A.1.2 IGRS File/Directory Object Identifier Format**

704 <IGRSObjectIDURN> ::= "urn:" <IGRSDeviceNS> ":" <IGRSObjectNS> ":" <IGRSObjectIDVAL>

705 Notes:

706 <IGRSDeviceNS> ::= Device ID (GUID)

707 <IGRSObjectNS> ::= <"Directory"> | <"File">

708 <IGRSObjectIDVAL> ::= Object path name, such as "/a/b/c", and is case sensitive

709 **A.2 ObjectType**

710 Identified Object Type. The legal values: FILE|DIRECTORY

711 Notes:

712 If the object is file, the value of ObjectType shall be FILE.

713 If the object is directory, the value of ObjectType shall be DIRECTORY.

714 **A.3 Type_FilterRule**715 **A.3.1 Syntactic Definition**

716 < FilterRule > ::= <NILSTRING> | <LogicalExpression>

717 <LogicalExpression> ::= <RelationExpression> | '(<LogicalExpression>)' | 'not'

718 <LogicalExpression> | <LogicalExpression>

719 <BinaryLogicalOperator> <LogicalExpression>

720 <BinaryLogicalOperator> ::= 'and' | 'or'

721 <RelationExpression> ::= <ATTRIBUTENAME> <RelationOperator> <CONSTANTVALUE>

722 <RelationOperator> ::= '=' | '<' | '>' | '<=' | '>=' | '<>' | 'like' | 'not like'

723 **A.3.2 Semantic Description**

724 a) <NILSTRING> denotes it is a null string. Any Item shall satisfy the searching condition
725 indicated by <NILSTRING>;

726 b) <ATTRIBUTENAME> is the attribute name included in the return result of
727 GetSearchCapability;

728 c) <CONSTANTVALUE> is a constant, its type shall match the type of <ATTRIBUTENAME>;

729 d) 'like' | 'not like' is a relation operator of mode matching. They denotes 'matching' | 'not
730 matching' respectively.

731 e) The wild card that appears in matching mode is '%'. For example, 'abc%' means a string
732 started with abc. '%abc' means a string ended with abc. '%abc%' means a string contained
733 abc. 'abc%def' means a string started with abc and ended with def, etc.

734 f) '=' | '<' | '>' | '<=' | '>=' | '<>' are comparison operators. They denote "equal" | "less than" | "greater
735 than" | "greater than or equal" | "less than or equal" | "not equal".

736 g) 'and' | 'or' are binary logic operators.

737 'not' is a unitary logic operators.

738 **A.4 Type_SortRule**739 **A.4.1 Syntactic Definition**

740 <SortRule> ::= <NILSTRING> | <SortExpression>

741 <SortExpression>::= <ATTRIBUTENAME> <SortSwitch> |<ATTRIBUTENAME> <SortSwitch> ','
 742 <SortExpression>
 743 <SortSwitch>::= 'ASC' | 'DESC'

744 **A.4.2 Semantic Description**

- 745 a) <NILSTRING> denotes a null string. It indicates that the result can be processed without
 746 sorting.
 747 b) <ATTRIBUTENAME> is the name of attribute that is contained in the return result of
 748 GetSortCapability;
 749 c) 'ASC'|'DESC' are sorting options. They mean "ascending sort"|"descending sort";
 750 d) The priority of the sorting indicators separated by ',' is from left to right.

751 **A.5 Content Directory Object Update Notification Message**

752 **A.5.1 Definition**

```

753 <element name="FamsUpdateNotification">
754 <complexType>
755 <sequence>
756 <element name="SubscribeObjectId" type="xsd:string"/>
757 <element name="ParentDirectoryId" type="xsd:string"/>
758 <element name="EventObjectIdList" minOccurs="0">
759 <complexType>
760 <sequence>
761 <element name="EventObjectId"
762 type="xsd:string" minOccurs="1" maxOccurs="unbounded"/>
763 </sequence>
764 </complexType>
765 </element>
766 <element name="EventType">
767 <complexType>
768 <choice>
769 <element name="ChildrenAdded"/>
770 <element name="ChildrenDeleted"/>
771 <element name="NameChanged"/>
772 <element name="AccessRightChanged"/>
773 <element name="ContentChanged"/>
774 <element name="SelfDeleted"/>
775 </choice>
776 </complexType>
777 </element>
778 </sequence>
779 </complexType>
780 </element>

```

781 **A.6 Service Attribute Update Notification Message**

782 **A.6.1 Definition**

```

783 <element name="ServiceAttributeUpdateNotification">
784 <complexType>
785 <sequence>
786 <element name="SubscriptionId" type="xsd:string"/>
787 <element name="ServiceAttributeName" type="xsd:string"/>
788 <element name="ServiceAttributeValue" type="xsd:string"/>
789 </sequence>
790 </complexType>
791 </element>

```

792
793
794

Annex B
(Normative Annex)
Information device – IGRS – File Profile Message Format

795 B.1 Overview

796 IGRS File Profile messages adopt XML Schema description. Service invocation message formats
797 of IGRS File Profile shall conform to the service invocation message format specified in ISO/IEC
798 14543-5-1.

799 B.1.1 Service Invocation Request Message

800 The Client can initiate “request/response” service invocation based on the session with the target
801 service. The definition of the service invocation request message is shown in Table B 1:

802 Table B 1 - IGRS File Profile Service Invocation Request Message

Message	Field Explanation
M-POST /IGRS HTTP/1.1	HTTP EXTENDED COMMAND LINE
Host: Target Host IP address:Port	Required field
01-IGRSVersion: IGRS/1.0	Required field, IGRS version number
01-IGRSMessageType:InvokeServiceRequest	Required field, content shall be this
01-SourceDeviceId: requesting device identifier	Required field, and type is URI, defined in 8.1.2 of ISO/IEC 14543-5-1
01-TargetDeviceId: target device identifier	Required field, type is URI, defined in 8.1.2 of ISO/IEC 14543-5-1
Content-Length: Message body length	Required field
Content-type:text/xml; charset=utf-8	Required field
MAN:"http://www.igrs.org/spec1.0";ns=01	Required field
MAN:"http://schemas.xmlsoap.org/soap/envelope/";ns=02	Required field
02-SoapAction:"IGRS-InvokeService-Request"	Required field
	Shall be empty
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">	Required field
<SOAP-ENV:Body>	Required field
<Session xmlns="http://www.igrs.org/spec1.0"request="NeedResponse">	Required field, request attribute indicates that the invocation message needs response
<SourceClientId> <i>Source client identifier</i> </SourceClientId>	Required field, type is 32-bit UnsignedInt
<TargetServiceId> <i>Target service identifier</i> </TargetServiceId>	Required field, type is 32-bit UnsignedInt
<SequenceId> <i>Invocation request sequence Id</i> </SequenceId>	Required field, type is 32-bit UnsignedInt
<!--Here is the start of the specific invocation request message-->	
<Invocation interface name>	Required field
<Input parameter name> <i>Input parameter value</i> </Input parameter name>	The service invocation interface that meets the requirements of the IGRS specification ISO/IEC 14543-5-2
.....	

Message	Field Explanation
<Input parameter name> <i>Input parameter value</i> </Input parameter name>	The service invocation interface that meets the requirements of the IGRS specification ISO/IEC 14543-5-2.
</Invocation interface name>	Required field
<!--Here is the end of the specific invocation request message-->	
</Session>	Required field
</SOAP-ENV:Body>	Required field
</SOAP-ENV:Envelope>	Required field
Note: Italics indicate where content is to be inserted; all other text in message definitions is fixed.	

803

804 **B.1.2 Service Invocation Response Message**

805 When an IGRS service receives an invocation request with response requirement, it should
 806 return invocation response message. Service invocation response message is sent to the client
 807 who is invoking the service based on the same session. Message definition is shown in Table B
 808 2:

809

Table B 2 - Service Invocation Response Message

Message	Field Explanation
HTTP/1.1 200 OK	HTTP Command line
Ext:	Required field
Cache-control:no-cache="Ext"	Required field
MAN:"http://www.igrs.org/spec1.0";ns=01	Required field
01-IGRSVersion: IGRS/1.0	Required field, IGRS Version number
01-IGRSMessageType:InvokeServiceResponse	Required field, content shall be this
01-TargetDeviceId: target device identifier	Required field, type is URI
01-SourceDeviceId: requesting device identifier	Required field, type is URI
01-Acknowledgeld: Service pipe message response sequence Id	Required field, Type is 32 bit UnsignedInt
Content-Length: Message body length	Required field
Content-type:text/xml; charset=utf-8	Required field
MAN:"http://schemas.xmlsoap.org/soap/envelope/";ns=02	Required field
02-SoapAction:"IGRS-InvokeService-Response"	Required field
	Shall be empty
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">	Required field
<SOAP-ENV:Body>	Required field
<Session xmlns="http://www.igrs.org/spec1.0">	Required field
<SourceServiceId> <i>Source service identifier</i> </SourceServiceId>	Required field, Type is 32 bit UnsignedInt
<TargetClientId> <i>Target client identifier</i> </TargetClientId>	Required field, Type is 32 bit UnsignedInt
<Acknowledgeld> <i>Invocation response message sequence Id</i> </Acknowledgeld>	Required field, Type is 32 bit UnsignedInt, It's the same as the invocation request SequenceId of the

Message	Field Explanation
	request message.
<ReturnCode> <i>Invocation response status code</i> </ReturnCode>	Required field, defined in Section 11 of ISO/IEC 14543-5-1.
<!--Here is the specific invocation response message-->	
<FileReturnCode> <i>File universal error return code</i> </FileReturnCode>	Optional field, Indicate the error response message independent of the service invocation, such as the service invocation is not supported. There shall be one and only one return code returned between FileReturnCode and ReturnCode.
<Invocation interface name>	Required field
<ReturnCode> <i>Service invocation response status code</i> </ReturnCode>	Optional field. Conform to service invocation return code definition. There shall be one and only one return code returned between FileReturnCode and ReturnCode.
< Output parameter> <i>Output parameter value</i> </ Output parameter>	Shall meet service interface definition requirement
.....	
< Output parameter> <i>Output parameter value</i> </ Output parameter>	Shall meet service interface definition requirement.
</ Invocation interface name>	Required field
<!--Here is the specific invocation response message-->	
</Session>	Required field
</SOAP-ENV:Body>	Required field
</SOAP-ENV:Envelope>	Required field

810

811 **B.1.3 File/Directory Object Update Notification Message**

812 FileAccessManagement service can notify client based on session setup with client when the
813 file/directory object changes. This message does not require any response. If file/directory object
814 update event is subscribed to a multi-directory structure, the update message should be sent
815 only to the lowest directory object. Message definition is shown in Table B 3:

816

Table B 3 - File/Directory Object Update Notification Message

Message	Field Explanation
M-NOTIFY //IGRS HTTP/1.1	HTTP EXTENDED COMMAND LINE
Host:Target host IP address:Port	Required field
MAN:"http://www.igrs.org/spec1.0";ns=01	Required field
01-IGRSVersion: IGRS/1.0	Required field, IGRS Version number
01-IGRSMessageType: FamsServiceNotifyClient	Required field, content shall be this
01-SourceDeviceId: Source device identifier	Required field, type is URI
01-TargetDeviceId: Target device identifier	Required field, type is URI
01-SequenceId: Device pipe message sequence Id	Required field, type is 32 bit UnsignedInt (0 is reserved)
Content-Length:message body length	Required field
Content-type:text/xml; charset=utf-8	Required field
MAN:"http://schemas.xmlsoap.org/soap/envelope/";ns=02	Required field
02-SoapAction:"IGRS-FamsServiceToClient-Notify"	Required field
	Shall be empty

Message	Field Explanation
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">	Required field
<SOAP-ENV:Body>	Required field
<Session xmlns="http://www.igrs.org/spec1.0"request="NoResponse">	Required field, "request" attribute indicates that no response is needed
<SourceServiceId> <i>Source Service identifier</i> </SourceServiceId>	Required field, Type is 32 bit UnsignedInt (0 is reserved)
<TargetClientId> <i>Target client identifier</i> </TargetClientId>	Required field, Type is 32 bit UnsignedInt (0 is reserved)
<SequenceId> <i>Invocation request sequence id</i> </SequenceId>	Required field, Type is 32 bit UnsignedInt (0 is reserved)
<!--Here is the specific notification message-->	
<FamsUpdateNotification>	Required field
<SubscribeObjectId> <i>The subscribed file/directory object identifier</i> </SubscribeObjectId>	Required field, refer to the Type_ObjectId data type definition in Annex C
<ParentDirectoryId> <i>ID of parent directory in which the update event occurred</i> </ParentDirectoryId>	Required field, refer to Type_ObjectId data type definition in Annex C
<EventObjectIdList> <i>File/Directory object ID list in which update event occurred</i> </EventObjectIdList>	Optional field, refer to Type_ObjectIdList data type definition in Annex B. the updated object identifier list.
<EventType> <i>File/directory object update event type</i> </EventType>	Required field, Type is String, The options are: "ChildrenAdded", "ChildrenDeleted", "NameChanged", "AccessRightChanged", "ContentChanged", "SelfDeleted"
</FamsUpdateNotification>	Required field
</Session>	Required field
</SOAP-ENV:Body>	Required field
</SOAP-ENV:Envelope>	Required field

817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838

Note:

XML schema of specific notification message in file/directory object update notification message is defined as:

```

<element name="FamsUpdateNotification">
  <complexType>
    <sequence>
      <element name="SubscribeObjectId" type="xsd:string"/>
      <element name="ParentDirectoryId" type="xsd:string"/>
      <element name="EventObjectIdList" minOccurs="0">
        <complexType>
          <sequence>
            <element name="EventObjectId"
              type="xsd:string"
              minOccurs="1" maxOccurs="unbounded"/>
          </sequence>
        </complexType>
      </element>
      <element name="EventType">
        <complexType>
          <choice>
            <element name="ChildrenAdded/>

```

```

839                                     <element name="ChildrenDeleted"/>
840                                     <element name="NameChanged"/>
841                                     <element
842           name="AccessRightChanged"/>
843                                     <element name="ContentChanged"/>
844                                     <element name="SelfDeleted"/>
845                                     </choice>
846                                     </complexType>
847                                     </element>
848                                     </sequence>
849           </complexType>
850 </element>
    
```

851 **B.1.4 Service Attribute Update Notification Message**

852 Service can notify the Client based on session setup with Client when service attribute changes.
853 This message does not require response. Message Definition is shown in Table B 4:

854 **Table B 4 - Service Attribute Update Notification Message**

Message	Field Explanation
M-NOTIFY //IGRS HTTP/1.1	HTTP EXTENDED COMMAND LINE
Host:Target host IP address:Port	Required field
MAN:"http://www.igrs.org/spec1.0";ns=01	Required field
01-IGRSVersion: IGRS/1.0	Required field, IGRS Version number
01-IGRSMessageType:ServiceNotifyClient	Required field, content shall be this
01-SourceDeviceId: Source Device identifier	Required field, type is URI
01-TargetDeviceId: Target Device Identifier	Required field, type is URI
01-SequenceId: Device pipe message sequence Id	Required field, type is 32 bit UnsignedInt (0 is reserved)
Content-Length:message body length	Required field
Content-type:text/xml; charset=utf-8	Required field
MAN:"http://schemas.xmlsoap.org/soap/envelope/";ns=02	Required field
02-SoapAction:"IGRS-ServiceToClient-Notify"	Required field
	Shall be empty
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">	Required field
<SOAP-ENV:Body>	Required field
<Session xmlns="http://www.igrs.org/spec1.0"request="NoResponse">	Required field, "request" attribute indicates that no response is needed
<SourceServiceId> <i>Source Service Identifier</i> </SourceServiceId>	Required field, type is 32 bit UnsignedInt (0 is reserved)
<TargetClientId> <i>Target Client Identifier</i> </TargetClientId>	Required field, type is 32 bit UnsignedInt (0 is reserved)
<SequenceId> <i>Invocation request sequence Id</i> </SequenceId>	Required field, type is 32 bit UnsignedInt (0 is reserved)
<!--Here is the specific notification message-->	
<ServiceAttributeUpdateNotification>	Required field
<SubscriptionId> <i>Subscription Identifier</i> </SubscriptionId>	Required field, String, Subscription Identifier
<ServiceAttributeName> <i>The updating Service attribute name</i>	Required field, String, the updating

Message	Field Explanation
</ServiceAttributeName>	Service attribute name
<ServiceAttributeValue > <i>Value of service attribute in which update event occurred</i> </ ServiceAttributeValue >	Required field. String.
</ServiceAttributeUpdateNotification>	Required field
</Session>	Required field
</SOAP-ENV:Body>	Required field
</SOAP-ENV:Envelope>	Required field

855

856

Note:

857

XML schema of specific notification message in service attribute update notification message is defined as:

858

859

```
<element name="ServiceAttributeUpdateNotification">
```

860

```
<complexType>
```

861

```
<sequence>
```

862

```
<element name="SubscriptionId" Type="xsd:string"/>
```

863

```
<element name="ServiceAttributeName" Type="xsd:string"/>
```

864

```
<element name="ServiceAttributeValue" Type="xsd:string"/>
```

865

```
</sequence>
```

866

```
</complexType>
```

867

```
</element>
```

868

B.2 FileAccessManagement Service

869

B.2.1 Service Data Type

870

```
<schema xmlns = "http://www.w3.org/2001/XMLSchema"
```

871

```
xmlns:igrs = "http://www.igrs.org/spec1.0"
```

872

```
targetNamespace = "http://www.igrs.org/spec1.0">
```

```
<element
```

```
name="
```

873

```
Type_AuthenticationKey" type="xsd:string"/>
```

874

```
<element name=" Type_InstanceId" type="xsd:string"/>
```

875

```
<element name=" Type_ObjectId" type="xsd:string"/>
```

876

```
<element name=" Type_SubscriptionId" type="xsd:string"/>
```

877

```
<element name=" Type_FilterRule" type="xsd:string"/>
```

878

```
<element name=" Type_SortRule" type="xsd:string" />
```

879

```
<element name=" Type_ServiceAttributeName" type="xsd:string"/>
```

880

```
<element name=" Type_Count" type="xsd:int"/>
```

881

```
<element name=" Type_SortCapability" type="xsd:string"/>
```

882

```
<element name=" Type_SearchCapability" type="xsd:string" />
```

883

```
<element name=" Type_DeleteMode" type="xsd:string"/>
```

884

```
<element name=" Type_UserAuthenticationInfo"
```

885

```
type="UserAuthenticationInfoType"/>
```

886

```
<element name=" Type_ObjectAttribute" type="ObjectAttributeType"/>
```

887

```
<element name=" Type_ObjectURI" type="xsd:string"/>
```

888

```
<element name=" Type_ObjectIdList">
```

889

```
<complexType>
```

890

```
<sequence>
```

891

```
<element name="ObjectId" type="xsd:string"
```

892

```
minOccurs="1" maxOccurs="unbounded"/>
```

893

```
</sequence>
```

894

```
</complexType>
```

895

```
</element>
```

896

```
<element name="Type_ObjectList">
```

897

```
<complexType>
```

898

```
<sequence>
```

899

```
<element name="Object" type="ObjectAttributeType"
```

900

```
minOccurs="0"/>
```

901

```
</sequence>
```

902

```
</complexType>
```

```

903         </element>
904         <element name="Type_ObjectURITreeList">
905             <complexType>
906                 <sequence>
907                     <element name="ObjectURITree"
908 type="ObjectURITreeType"
909             minOccurs="1" maxOccurs="unbounded"/>
910                 </sequence>
911             </complexType>
912         </element>
913         <complexType name="ObjectURITreeType">
914             <attribute name="ObjectAttribute" type="ObjectAttributeType"/>
915             <attribute name="ObjectURI" type="xsd:string"/>
916             <sequence>
917                 <element name="ObjectURITree"
918 type="ObjectURITreeType"
919             minOccurs="0" maxOccurs="unbounded"/>
920             </sequence>
921         </complexType>
922         <complexType name="UserAuthenticationInfoType">
923             <sequence>
924                 <element name="UserInfo" minOccurs="0">
925                     <complexType>
926                         <sequence>
927                             <element name="UserName"
928 type="xsd:string"/>
929                             <element name="UserPassword"
930 type="xsd:string"/>
931                         </sequence>
932                     </complexType>
933                 </element>
934                 <element name="DeviceInfo" minOccurs="0">
935                     <complexType>
936                         <sequence>
937                             <element name="DeviceId"
938 type="xsd:string"/>
939                             <element name="DeviceName"
940 type="xsd:string"/>
941                         </sequence>
942                     </complexType>
943                 </element>
944                 <element name="3rdPartyAuthen" minOccurs="0">
945                     <complexType>
946                         <sequence>
947                             <element name="3rdPartyAuthenType"
948 type="string"/>
949                         </sequence>
950                     </complexType>
951                 </element>
952             </sequence>
953         </complexType>
954         <complexType name="ObjectAttributeType">
955             <sequence>
956                 <element name="ObjectType" type="xsd:string"/>
957                 <element name="ObjectId" type="xsd:string"/>
958                 <element name="ObjectName" type="xsd:string"/>
959                 <element name="ParentId" type="xsd:string" minOccurs="0"/>
960                 <element name="DeviceId" type="xsd:string" minOccurs="0"/>
961                 <element name="DeviceName" type="xsd:string" minOccurs="0"/>
962                 <element name="AccessRight">

```

```

963         <complexType>
964             <sequence>
965                 <element name="Read" type="xsd:string"
966 minOccurs="0"/>
967                 <element name="Write" type="xsd:string"
968 minOccurs="0"/>
969                 <element name="Hide" type="xsd:string"
970 minOccurs="0"/>
971             </sequence>
972         </complexType>
973     </element>
974     <element name="CreateTime" type="xsd:string" minOccurs="0"/>
975     <element name="LastAccessTime" type="xsd:string"
976 minOccurs="0"/>
977     <element name="LastWriteTime" type="xsd:string"
978 minOccurs="0"/>
979     <element name="Size" type="xsd:string" minOccurs="0"/>
980     <element name="Num_SubDirectories" type="unsignedInt"
981 minOccurs="0"/>
982     <element name="Num_SubFiles" type="unsignedInt"
983 minOccurs="0"/>
984     </sequence>
985 </complexType>
986 </schema>

```

987 B.2.2 Service Invocation Message Format

988 B.2.2.1 GetAuthenticationKey

989 B.2.2.1.1 GetAuthenticationKeyRequest

```

990 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
991 xmlns:igrs = "http://www.igrs.org/spec1.0"
992 targetNamespace = "http://www.igrs.org/spec1.0">
993     <element name="GetAuthenticationKeyRequest">
994         <complexType>
995             <sequence>
996                 <element name="Instanceld" type="string"/>
997                 <element name="UserAuthenticationInfo"
998 type="Type_UserAuthenticationInfo"/>
999             </sequence>
1000         </complexType>
1001     </element>

```

1002 B.2.2.1.2 GetAuthenticationKeyResponse

```

1003 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1004 xmlns:igrs = "http://www.igrs.org/spec1.0"
1005 targetNamespace = "http://www.igrs.org/spec1.0">
1006     <element name="GetAuthenticationKeyResponse">
1007         <complexType>
1008             <sequence>
1009                 <element name="ReturnCode" type="unsignedInt"/>
1010                 <element name="AuthenticationKey" type="string"/>
1011             </sequence>
1012         </complexType>
1013     </element>
1014 </schema>

```

1015 B.2.2.2 GetSortCapability**1016 B.2.2.2.1 GetSortCapabilityRequest**

```

1017 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1018 xmlns:igrs = "http://www.igrs.org/spec1.0"
1019 targetNamespace = "http://www.igrs.org/spec1.0">
1020   <element name="GetSortCapabilityRequest">
1021     <complexType>
1022       <sequence>
1023         <element name=" AuthenticationKey" type="string"/>
1024       </sequence>
1025     </complexType>
1026   </element>

```

1027 B.2.2.2.2 GetSortCapabilityResponse

```

1028 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1029 xmlns:igrs = "http://www.igrs.org/spec1.0"
1030 targetNamespace = "http://www.igrs.org/spec1.0">
1031   <element name="GetSortCapabilityResponse">
1032     <complexType>
1033       <sequence>
1034         <element name="ReturnCode" type="unsignedInt"/>
1035         <element name="SortCaps" ref="igrs: Type_SortCapability"/>
1036       </sequence>
1037     </complexType>
1038   </element>
1039 </schema>

```

1040 B.2.2.3 GetSearchCapability**1041 B.2.2.3.1 GetSearchCapabilityRequest**

```

1042 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1043 xmlns:igrs = "http://www.igrs.org/spec1.0"
1044 targetNamespace = "http://www.igrs.org/spec1.0">
1045   <element name="GetSearchCapabilityRequest">
1046     <complexType>
1047       <sequence>
1048         <element name=" AuthenticationKey" type="string"/>
1049       </sequence>
1050     </complexType>
1051   </element>

```

1052 B.2.2.3.2 GetSearchCapabilityResponse

```

1053 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1054 xmlns:igrs = "http://www.igrs.org/spec1.0"
1055   targetNamespace = "http://www.igrs.org/spec1.0"><element
1056   name="GetType_SearchCapabilityResponse">
1057     <complexType>
1058       <sequence>
1059         <element name="ReturnCode" type="unsignedInt"/>
1060         <element name="SearchCaps" ref="igrs: Type_SearchCapability"/>
1061       </sequence>
1062     </complexType>
1063   </element>

```

1064 B.2.2.4 Browse**1065 B.2.2.4.1 BrowseRequest**

```

1066 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1067 xmlns:igrs = "http://www.igrs.org/spec1.0"

```

```
1068 targetNamespace = "http://www.igrs.org/spec1.0">
1069   <element name="BrowseRequest">
1070     <complexType>
1071       <sequence>
1072         <element name="AuthenticationKey" type="string"/>
1073         <element name="ObjectId" type="Type_ObjectId"/>
1074         <element name="BrowseFilter" type="Type_FilterRule"/>
1075         <element name="StartOffset" type="Type_Count"/>
1076         <element name="RequestedCount" type="Type_Count"/>
1077         <element name="SortRule" type="Type_SortRule"/>
1078       </sequence>
1079     </complexType>
1080   </element>
```

1081 **B.2.2.4.2 BrowseResponse**

```
1082 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1083 xmlns:igrs = "http://www.igrs.org/spec1.0"
1084 targetNamespace = "http://www.igrs.org/spec1.0">
1085   <element name="BrowseResponse">
1086     <complexType>
1087       <sequence>
1088         <element name="ReturnCode" type="unsignedInt"/>
1089         <element name="Result" type="Type_ObjectList"/>
1090         <element name="NumberReturned" type="Type_Count"/>
1091         <element name="NumberTotalMatched" type="Type_Count"/>
1092       </sequence>
1093     </complexType>
1094   </element>
```

1095 **B.2.2.5 GetAttribute**

1096 **B.2.2.5.1 GetAttributeRequest**

```
1097 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1098 xmlns:igrs = "http://www.igrs.org/spec1.0"
1099 targetNamespace = "http://www.igrs.org/spec1.0">
1100   <element name="GetAttributeRequest">
1101     <complexType>
1102       <sequence>
1103         <element name="AuthenticationKey" type="string"/>
1104         <element name="ObjectId" type="Type_ObjectId"/>
1105       </sequence>
1106     </complexType>
1107   </element>
```

1108 **B.2.2.5.2 GetAttributeResponse**

```
1109 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1110 xmlns:igrs = "http://www.igrs.org/spec1.0"
1111 targetNamespace = "http://www.igrs.org/spec1.0">
1112   <element name="GetFileAttributeResponse">
1113     <complexType>
1114       <sequence>
1115         <element name="ReturnCode" type="unsignedInt"/>
1116         <element name="ObjectAttribute" type="Type_ObjectAttribute"/>
1117       </sequence>
1118     </complexType>
1119   </element>
```

1120 **B.2.2.6 SetAttribute**1121 **B.2.2.6.1 SetAttribute Request**

```

1122 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1123 xmlns:igrs = "http://www.igrs.org/spec1.0"
1124 targetNamespace = "http://www.igrs.org/spec1.0">
1125   <element name="SetAttributeRequest">
1126     <complexType>
1127       <sequence>
1128         <element name=" AuthenticationKey" type="string"/>
1129         <element name="ObjectId" type=" Type_ObjectId"/>
1130         <element name="ObjectAttribute" type=" Type_ObjectAttribute"/>
1131       </sequence>
1132     </complexType>
1133   </element>

```

1134 **B.2.2.6.2 SetAttributeResponse**

```

1135 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1136 xmlns:igrs = "http://www.igrs.org/spec1.0"
1137 targetNamespace = "http://www.igrs.org/spec1.0">
1138   <element name=" SetAttributeResponse">
1139     <complexType>
1140       <sequence>
1141         <element name="ReturnCode" type="unsignedInt"/>
1142       </sequence>
1143     </complexType>
1144   </element>

```

1145 **B.2.2.7 Search**1146 **B.2.2.7.1 SearchRequest**

```

1147 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1148 xmlns:igrs = "http://www.igrs.org/spec1.0"
1149 targetNamespace = "http://www.igrs.org/spec1.0">
1150   <element name="SearchRequest">
1151     <complexType>
1152       <sequence>
1153         <element name=" AuthenticationKey" type="string"/>
1154         <element name="ObjectIdList" type=" Type_ObjectIdList"/>
1155         <element name="SearchRule" type=" Type_FilterRule"/>
1156         <element name="StartOffset" type=" Type_Count"/>
1157         <element name="RequestedCount" type=" Type_Count"/>
1158         <element name="SortRule" type=" Type_SortRule"/>
1159       </sequence>
1160     </complexType>
1161   </element>

```

1162 **B.2.2.7.2 SearchResponse**

```

1163 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1164 xmlns:igrs = "http://www.igrs.org/spec1.0"
1165 targetNamespace = "http://www.igrs.org/spec1.0">
1166   <element name="SearchResponse">
1167     <complexType>
1168       <sequence>
1169         <element name="ReturnCode" type="unsignedInt"/>
1170         <element name="Result" type=" Type_ObjectList"/>
1171         <element name="NumberReturned" type=" Type_Count"/>
1172         <element name="NumberTotalMatched" type=" Type_Count"/>
1173       </sequence>
1174     </complexType>

```

1175 </element>

1176 **B.2.2.8 GetBrowseFilter**

1177 **B.2.2.8.1 GetBrowseFilterRequest**

```
1178 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1179 xmlns:igrs = "http://www.igrs.org/spec1.0"
1180 targetNamespace = "http://www.igrs.org/spec1.0">
1181   <element name=" GetBrowseFilterRequest">
1182     <complexType>
1183       <sequence>
1184         <element name=" AuthenticationKey" type="string"/>
1185       </sequence>
1186     </complexType>
1187   </element>
```

1188 **B.2.2.8.2 GetBrowseFilterResponse**

```
1189 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1190 xmlns:igrs = "http://www.igrs.org/spec1.0"
1191 targetNamespace = "http://www.igrs.org/spec1.0">
1192   <element name="GetBrowseFilterResponse">
1193     <complexType>
1194       <sequence>
1195         <element name="ReturnCode" type="unsignedInt"/>
1196         <element name="BrowseFilter" type=" Type_FilterRule"/>
1197       </sequence>
1198     </complexType>
1199   </element>
1200 </schema>
```

1201 **B.2.2.9 SetBrowseFilter**

1202 **B.2.2.9.1 SetBrowseFilterRequest**

```
1203 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1204 xmlns:igrs = "http://www.igrs.org/spec1.0"
1205 targetNamespace = "http://www.igrs.org/spec1.0">
1206   <element name=" SetBrowseFilterRequest">
1207     <complexType>
1208       <sequence>
1209         <element name="AuthenticationKey" type="string"/>
1210         <element name="BrowseFilter" type=" Type_FilterRule"/>
1211       </sequence>
1212     </complexType>
1213   </element>
```

1214 **B.2.2.9.2 SetBrowseFilterResponse**

```
1215 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1216 xmlns:igrs = "http://www.igrs.org/spec1.0"
1217 targetNamespace = "http://www.igrs.org/spec1.0">
1218   <element name="SetBrowseFilterResponse">
1219     <complexType>
1220       <sequence>
1221         <element name="ReturnCode" type="unsignedInt"/>
1222       </sequence>
1223     </complexType>
1224   </element>
1225 </schema>
```

1226 **B.2.2.10 New**

1227 **B.2.2.10.1 NewRequest**

```
1228 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1229 xmlns:igrs = "http://www.igrs.org/spec1.0"
1230 targetNamespace = "http://www.igrs.org/spec1.0">
1231   <element name=" NewRequest">
1232     <complexType>
1233       <sequence>
1234         <element name=" AuthenticationKey" type="string"/>
1235         <element name="ParentId" type=" Type_ObjectId"/>
1236         <element name="ObjectAttribute" type=" Type_ObjectAttribute"/>
1237       </sequence>
1238     </complexType>
1239   </element>
```

1240 **B.2.2.10.2 NewResponse**

```
1241 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1242 xmlns:igrs = "http://www.igrs.org/spec1.0"
1243 targetNamespace = "http://www.igrs.org/spec1.0">
1244   <element name=" NewResponse">
1245     <complexType>
1246       <sequence>
1247         <element name="ReturnCode" type="unsignedInt"/>
1248         <element name="ObjectId" type=" Type_ObjectId"/>
1249       </sequence>
1250     </complexType>
1251   </element>
```

1252 **B.2.2.11 Copy**

1253 **B.2.2.11.1 CopyRequest**

```
1254 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1255 xmlns:igrs = "http://www.igrs.org/spec1.0"
1256 targetNamespace = "http://www.igrs.org/spec1.0">
1257   <element name=" CopyRequest">
1258     <complexType>
1259       <sequence>
1260         <element name=" AuthenticationKey" type="string"/>
1261         <element name="SourceObjectId" type="Type_ObjectId"/>
1262         <element name="DestParentId" type="Type_ObjectId"/>
1263       </sequence>
1264     </complexType>
1265   </element>
```

1266 **B.2.2.11.2 CopyResponse**

```
1267 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1268 xmlns:igrs = "http://www.igrs.org/spec1.0"
1269 targetNamespace = "http://www.igrs.org/spec1.0">
1270   <element name="CopyResponse">
1271     <complexType>
1272       <sequence>
1273         <element name="ReturnCode" type="unsignedInt"/>
1274         <element name="DestObjectId" type="Type_ObjectId"/>
1275         <element name="DestObjectAttribute" type="Type_ObjectAttribute"/>
1276       </sequence>
1277     </complexType>
1278   </element>
```

1279 **B.2.2.12 Move**1280 **B.2.2.12.1 MoveRequest**

```
1281 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1282 xmlns:igrs = "http://www.igrs.org/spec1.0"
1283 targetNamespace = "http://www.igrs.org/spec1.0">
1284   <element name="MoveRequest">
1285     <complexType>
1286       <sequence>
1287         <element name="AuthenticationKey" type="string"/>
1288         <element name="SourceObjectId" type="Type_ObjectId"/>
1289         <element name="DestParentId" type="Type_ObjectId"/>
1290       </sequence>
1291     </complexType>
1292   </element>
```

1293 **B.2.2.12.2 MoveResponse**

```
1294 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1295 xmlns:igrs = "http://www.igrs.org/spec1.0"
1296 targetNamespace = "http://www.igrs.org/spec1.0">
1297   <element name="MoveResponse">
1298     <complexType>
1299       <sequence>
1300         <element name="ReturnCode" type="unsignedInt"/>
1301         <element name="DestObjectId" type="Type_ObjectId"/>
1302         <element name="DestObjectAttribute" type="Type_ObjectAttribute"/>
1303       </sequence>
1304     </complexType>
1305   </element>
```

1306 **B.2.2.13 Delete**1307 **B.2.2.13.1 DeleteRequest**

```
1308 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1309 xmlns:igrs = "http://www.igrs.org/spec1.0"
1310 targetNamespace = "http://www.igrs.org/spec1.0">
1311   <element name="DeleteRequest">
1312     <complexType>
1313       <sequence>
1314         <element name="AuthenticationKey" type="string"/>
1315         <element name="ObjectId" type="Type_ObjectId"/>
1316         <element name="DeleteMode" type="Type_DeleteMode"/>
1317       </sequence>
1318     </complexType>
1319   </element>
```

1320 **B.2.2.13.2 DeleteResponse**

```
1321 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1322 xmlns:igrs = "http://www.igrs.org/spec1.0"
1323 targetNamespace = "http://www.igrs.org/spec1.0">
1324   <element name="DeleteResponse">
1325     <complexType>
1326       <sequence>
1327         <element name="ReturnCode" type="unsignedInt"/>
1328       </sequence>
1329     </complexType>
1330   </element>
```

1331 **B.2.2.14 PrepareforDownload**1332 **B.2.2.14.1 PrepareforDownloadRequest**

```

1333 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1334 xmlns:igrs = "http://www.igrs.org/spec1.0"
1335 targetNamespace = "http://www.igrs.org/spec1.0">
1336   <element name=" PrepareforDownloadRequest">
1337     <complexType>
1338       <sequence>
1339         <element name="AuthenticationKey" type="string">
1340           <element name="SourceObjectIdList" type=" Type_ObjectIdList"/>
1341         </sequence>
1342       </complexType>
1343     </element>

```

1344 **B.2.2.14.2 PrepareforDownloadResponse**

```

1345 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1346 xmlns:igrs = "http://www.igrs.org/spec1.0"
1347 targetNamespace = "http://www.igrs.org/spec1.0">
1348   <element name=" PrepareforDownloadResponse">
1349     <complexType>
1350       <sequence>
1351         <element name="ReturnCode" type="unsignedInt"/>
1352         <element name="SourceObjectURITreeList"
1353           type="Type_ObjectURITreeList"/>
1354       </sequence>
1355     </complexType>
1356   </element>

```

1357 **B.2.2.15 PrepareforUpload**1358 **B.2.2.15.1 PrepareforUploadRequest**

```

1359 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1360 xmlns:igrs = "http://www.igrs.org/spec1.0"
1361 targetNamespace = "http://www.igrs.org/spec1.0">
1362   <element name=" PrepareforUploadRequest">
1363     <complexType>
1364       <sequence>
1365         <element name="AuthenticationKey" type="string"/>
1366         <element name="ObjectAttribute" type="Type_ObjectAttribute"/>
1367         <element name="DestParentId" type="Type_ObjectId"/>
1368       </sequence>
1369     </complexType>
1370   </element>

```

1371 **B.2.2.15.2 PrepareforUploadResponse**

```

1372 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1373 xmlns:igrs = "http://www.igrs.org/spec1.0"
1374 targetNamespace = "http://www.igrs.org/spec1.0">
1375   <element name=" PrepareforUploadResponse">
1376     <complexType>
1377       <sequence>
1378         <element name="ReturnCode" type="unsignedInt"/>
1379         <element name="DestParentURI" type="Type_ObjectURI"/>
1380       </sequence>
1381     </complexType>
1382   </element>

```

1383 B.2.2.16 SubscribeObjectChange**1384 B.2.2.16.1 SubscribeObjectChangeRequest**

```
1385 <schema xmlns = "http://www.w3.org/2001/XMLSchema"  
1386 xmlns:igrs = "http://www.igrs.org/spec1.0"  
1387 targetNamespace = "http://www.igrs.org/spec1.0">  
1388   <element name=" SubscribeObjectChangeRequest">  
1389     <complexType>  
1390       <sequence>  
1391         <element name="AuthenticationKey" type="string"/>  
1392         <element name="ObjectId" type=" Type_ObjectId"/>  
1393       </sequence>  
1394     </complexType>  
1395   </element>
```

1396 B.2.2.16.2 SubscribeObjectChangeResponse

```
1397 <schema xmlns = "http://www.w3.org/2001/XMLSchema"  
1398 xmlns:igrs = "http://www.igrs.org/spec1.0"  
1399 targetNamespace = "http://www.igrs.org/spec1.0">  
1400   <element name=" SubscribeObjectChangeResponse">  
1401     <complexType>  
1402       <sequence>  
1403         <element name="ReturnCode" type="unsignedInt"/>  
1404       </sequence>  
1405     </complexType>  
1406   </element>
```

1407 B.2.2.17 UnSubscribeServiceAttribute**1408 B.2.2.17.1 UnSubscribeServiceAttributeRequest**

```
1409 <schema xmlns = "http://www.w3.org/2001/XMLSchema"  
1410 xmlns:igrs = "http://www.igrs.org/spec1.0"  
1411 targetNamespace = "http://www.igrs.org/spec1.0">  
1412   <element name="UnSubscribeServiceAttributeRequest">  
1413     <complexType>  
1414       <sequence>  
1415         <element name=" AuthenticationKey" type="string"/>  
1416         <element name="ObjectId" type=" Type_ObjectId"/>  
1417       </sequence>  
1418     </complexType>  
1419   </element>
```

1420 B.2.2.17.2 UnSubscribeServiceAttributeResponse

```
1421 <schema xmlns = "http://www.w3.org/2001/XMLSchema"  
1422 xmlns:igrs = "http://www.igrs.org/spec1.0"  
1423 targetNamespace = "http://www.igrs.org/spec1.0">  
1424   <element name="UnSubscribeServiceAttributeResponse">  
1425     <complexType>  
1426       <sequence>  
1427         <element name="ReturnCode" type="unsignedInt"/>  
1428       </sequence>  
1429     </complexType>  
1430   </element>
```

1431 B.2.2.18 SubscribeServiceAttribute**1432 B.2.2.18.1 SubscribeServiceAttributeRequest**

```
1433 <schema xmlns = "http://www.w3.org/2001/XMLSchema"  
1434 xmlns:igrs = "http://www.igrs.org/spec1.0"  
1435 targetNamespace = "http://www.igrs.org/spec1.0">
```

```
1436     <element name="SubscribeServiceAttributeRequest">
1437         <complexType>
1438             <sequence>
1439                 <element name=" AuthenticationKey" type="string"/>
1440                 <element name="ServiceAttributeName" type="string"/>
1441             </sequence>
1442         </complexType>
1443     </element>
```

1444 **B.2.2.18.2 SubscribeServiceAttributeResponse**

```
1445 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1446 xmlns:igrs = "http://www.igrs.org/spec1.0"
1447 targetNamespace = "http://www.igrs.org/spec1.0">
1448     <element name="SubscribeServiceAttributeResponse">
1449         <complexType>
1450             <sequence>
1451                 <element name="ReturnCode" type="unsignedInt"/>
1452                 <element name="SubscriptionId" type="string"/>
1453             </sequence>
1454         </complexType>
1455     </element>
```

1456 **B.2.2.19 UnsubscribeServiceAttribute**

1457 **B.2.2.19.1 UnsubscribeServiceAttributeRequest**

```
1458 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1459 xmlns:igrs = "http://www.igrs.org/spec1.0"
1460 targetNamespace = "http://www.igrs.org/spec1.0">
1461     <element name="UnsubscribeServiceAttributeRequest">
1462         <complexType>
1463             <sequence>
1464                 <element name=" AuthenticationKey" type="string"/>
1465                 <element name="SubscriptionId" type="string"/>
1466             </sequence>
1467         </complexType>
1468     </element>
```

1469 **B.2.2.19.2 UnsubscribeServiceAttributeResponse**

```
1470 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1471 xmlns:igrs = "http://www.igrs.org/spec1.0"
1472 targetNamespace = "http://www.igrs.org/spec1.0">
1473     <element name="UnsubscribeServiceAttributeResponse">
1474         <complexType>
1475             <sequence>
1476                 <element name="ReturnCode" type="unsignedInt"/>
1477             </sequence>
1478         </complexType>
1479     </element>
```

1480 **B.3 FileConnectionManagement Service**

1481 **B.3.1 Service Data Type**

```
1482 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1483 xmlns:igrs = "http://www.igrs.org/spec1.0"
1484 targetNamespace = "http://www.igrs.org/spec1.0">
1485     <complexType base="PortocolInfoListType">
1486         <sequence>
1487             <element name="Type_ProtocolInfo" type="ProtocolInfoType"/>
1488             <element name="Type_ConnectionId" type="xsd: unsignedInt"/>
1489         </sequence>
1490     </complexType>
```

```

1489     <element name="Type_ServiceAttributeName" type="xsd: string" minOccurs="0"/>
1490     <element name="Type_SubscriptionId" type="xsd: string" minOccurs="0"/>
1491     <element name="Type_IPList" minOccurs="0">
1492         <complexType>
1493             <sequence>
1494                 <element name="IP" type="xsd: string"
1495 maxOccurs="unbounded"/>
1496             </sequence>
1497         </complexType>
1498     </element>
1499     <complexType name="PortcollInfoListType">
1500         <sequence>
1501             <element name="ProtocollInfo" type="ProtocollInfoType"
1502 minOccurs="0" maxOccurs="unbounded"/>
1503         </sequence>
1504     </complexType>
1505     <complexType name="ProtocollInfoType">
1506         <sequence>
1507             <element name="TransportProtocol">
1508                 <complexType>
1509                     <sequence>
1510                         <element name="Port" type="xsd: string"
1511 minOccurs="0" maxOccurs="unbounded"/>
1512                     </sequence>
1513                     <attribute name="Name" type="xsd: string"/>
1514                 </complexType>
1515             </element>
1516             <element name="ParameterList" type="xsd: string"
1517 minOccurs="0"/>
1518             <element name="AdditionalInfo" type="xsd: string"
1519 minOccurs="0"/>
1520         </sequence>

```

1521 **B.3.2 Service Invocation Message Format**

1522 **B.3.2.1 GetProtocollInfo**

1523 **B.3.2.1.1 GetProtocollInfoRequest**

```

1524 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1525 xmlns:igrs = "http://www.igrs.org/spec1.0"
1526 targetNamespace = "http://www.igrs.org/spec1.0">
1527     <element name="GetProtocollInfoRequest">

```

1528 **B.3.2.1.2 GetProtocollInfoResponse**

```

1529 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1530 xmlns:igrs = "http://www.igrs.org/spec1.0"
1531 targetNamespace = "http://www.igrs.org/spec1.0">
1532     <element name="GetProtocollInfoResponse">
1533         <complexType>
1534             <sequence>
1535                 <element name="ReturnCode" type="unsignedInt"/>
1536                 <element name="ProtocollInfoList" ref="igrs: Type_ProtocollInfoList"/>
1537                 <element name="IPList" ref="igrs: Type_IPList"/>
1538             </sequence>
1539         </complexType>
1540     </element>

```

1541 B.3.2.2 PrepareforConnection**1542 B.3.2.2.1 PrepareforConnectionRequest**

```
1543 <schema xmlns = "http://www.w3.org/2001/XMLSchema"  
1544 xmlns:igrs = "http://www.igrs.org/spec1.0"  
1545 targetNamespace = "http://www.igrs.org/spec1.0">  
1546   <element name="PrepareforConnectionRequest">  
1547     <complexType>  
1548       <sequence>  
1549         <element name="RemoteProcollInfo" type="Type_ProcollInfo"/>  
1550       </sequence>  
1551     </complexType>  
1552   </element>
```

1553 B.3.2.2.2 PrepareforConnectionResponse

```
1554 <schema xmlns = "http://www.w3.org/2001/XMLSchema"  
1555 xmlns:igrs = "http://www.igrs.org/spec1.0"  
1556 targetNamespace = "http://www.igrs.org/spec1.0">  
1557   <element name="PrepareforConnectionResponse">  
1558     <complexType>  
1559       <sequence>  
1560         <element name="ReturnCode" type="unsignedInt"/>  
1561         <element name="ConnectionId" type="Type_ConnectionId"/>  
1562       </sequence>  
1563     </complexType>  
1564   </element>
```

1565 B.3.2.3 ReleaseConnection**1566 B.3.2.3.1 ReleaseConnectionRequest**

```
1567 <schema xmlns = "http://www.w3.org/2001/XMLSchema"  
1568 xmlns:igrs = "http://www.igrs.org/spec1.0"  
1569 targetNamespace = "http://www.igrs.org/spec1.0">  
1570   <element name="ReleaseConnectionRequest".  
1571     <complexType>  
1572       <sequence>  
1573         <element name="ConnectionId" type="Type_ConnectionId"/>  
1574       </sequence>  
1575     </complexType>  
1576   </element>
```

1577 B.3.2.3.2 ReleaseConnectionResponse

```
1578 <schema xmlns = "http://www.w3.org/2001/XMLSchema"  
1579 xmlns:igrs = "http://www.igrs.org/spec1.0"  
1580 targetNamespace = "http://www.igrs.org/spec1.0">  
1581   <element name="ReleaseConnectionResponse">  
1582     <complexType>  
1583       <sequence>  
1584         <element name="ReturnCode" type="unsignedInt"/>  
1585       </sequence>  
1586     </complexType>  
1587   </element>
```

1588 B.3.2.4 SubscribeServiceAttribute**1589 B.3.2.4.1 SubscribeServiceAttributeRequest**

```
1590 <schema xmlns = "http://www.w3.org/2001/XMLSchema"  
1591 xmlns:igrs = "http://www.igrs.org/spec1.0"  
1592 targetNamespace = "http://www.igrs.org/spec1.0">  
1593   <element name="SubscribeServiceAttributeRequest">
```

```
1594     <complexType>
1595         <sequence>
1596             <element name="ServiceAttributeName" Type="string"/>
1597         </sequence>
1598     </complexType>
1599 </element>
```

1600 **B.3.2.4.2 SubscribeServiceAttributeResponse**

```
1601 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1602 xmlns:igrs = "http://www.igrs.org/spec1.0"
1603 targetNamespace = "http://www.igrs.org/spec1.0">
1604     <element name="SubscribeServiceAttributeResponse">
1605         <complexType>
1606             <sequence>
1607                 <element name="ReturnCode" type="unsignedInt"/>
1608                 <element name="SubscriptionId" type="string"/>
1609             </sequence>
1610         </complexType>
1611     </element>
```

1612 **B.3.2.5 UnsubscribeServiceAttribute**

1613 **B.3.2.5.1 UnsubscribeServiceAttributeRequest**

```
1614 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1615 xmlns:igrs = "http://www.igrs.org/spec1.0"
1616 targetNamespace = "http://www.igrs.org/spec1.0">
1617     <element name="UnsubscribeServiceAttributeRequest">
1618         <complexType>
1619             <sequence>
1620                 <element name="SubscriptionId" type="string"/>
1621             </sequence>
1622         </complexType>
1623     </element>
```

1624 **B.3.2.5.2 UnsubscribeServiceAttributeResponse**

```
1625 <schema xmlns = "http://www.w3.org/2001/XMLSchema"
1626 xmlns:igrs = "http://www.igrs.org/spec1.0"
1627 targetNamespace = "http://www.igrs.org/spec1.0">
1628     <element name="UnsubscribeServiceAttributeResponse">
1629         <complexType>
1630             <sequence>
1631                 <element name="ReturnCode" type="unsignedInt"/>
1632             </sequence>
1633         </complexType>
1634     </element>
1635 </schema>
```

1636
1637
1638

Annex C
(Normative Annex)
File Profile – FileAccessManagement Service Description.wsdl

```

1639 <?xml version="1.0" encoding="UTF-8"?>
1640 <definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
1641 xmlns:http="http://schemas.xmlsoap.org/wsdl/http/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
1642 xmlns:igrsExt="http://www.igrs.org/igrs/ServiceDescription"
1643 xmlns:igrs="http://www.igrs.org/igrs/FileAccessManagement"
1644 targetNamespace="http://www.igrs.org/igrs/FileAccessManagement" name="FileAccessManagement">
1645     <types>
1646         <element name="Type_AuthenticationKey" type="xsd:string"/>
1647         <element name="Type_InstanceId" type="xsd:string"/>
1648         <element name="Type_ObjectId" type="xsd:string"/>
1649         <element name="Type_SubscriptionId" type="xsd:string"/>
1650         <element name="Type_FilterRule" type="xsd:string"/>
1651         <element name="Type_SortRule" type="xsd:string" />
1652         <element name="Type_ServiceAttributeName" type="xsd:string"/>
1653         <element name="Type_Count" type="xsd:int"/>
1654         <element name="Type_SortCapability" type="xsd:string"/>
1655         <element name="Type_SearchCapability" type="xsd:string" />
1656         <element name="Type_DeleteMode" type="xsd:string"/>
1657         <element name="Type_UserAuthenticationInfo" type="UserAuthenticationInfoType"/>
1658         <element name="Type_ObjectAttribute" type="ObjectAttributeType"/>
1659         <element name="Type_ObjectURI" type="xsd:string"/>
1660         <element name="Type_ObjectIdList">
1661             <complexType>
1662                 <sequence>
1663                     <element name="ObjectId" type="xsd:string" minOccurs="1"
1664 maxOccurs="unbounded"/>
1665                 </sequence>
1666             </complexType>
1667         </element>
1668         <element name="Type_ObjectList">
1669             <complexType>
1670                 <sequence>
1671                     <element name="Object" type="ObjectAttributeType"
1672 minOccurs="0"/>
1673                 </sequence>
1674             </complexType>
1675         </element>
1676         <element name="Type_ObjectURITreeList">
1677             <complexType>
1678                 <sequence>
1679                     <element name="ObjectURITree" type="ObjectURITreeType"
1680 minOccurs="1" maxOccurs="unbounded"/>
1681                 </sequence>
1682             </complexType>
1683         </element>
1684         <complexType name="ObjectURITreeType">
1685             <attribute name="ObjectAttribute" type="ObjectAttributeType"/>
1686             <attribute name="ObjectURI" type="xsd:string"/>
1687             <sequence>
1688                 <element name="ObjectURITree" type="ObjectURITreeType"
1689 minOccurs="0" maxOccurs="unbounded"/>
1690             </sequence>
1691         </complexType>
1692         <complexType name="UserAuthenticationInfoType">
1693             <sequence>

```

```

1694         <element name="UserInfo" minOccurs="0">
1695             <complexType>
1696                 <sequence>
1697                     <element name="UserName" type="xsd:string"/>
1698                     <element name="UserPassword" type="xsd:string"/>
1699                 </sequence>
1700             </complexType>
1701         </element>
1702         <element name="DeviceInfo" minOccurs="0">
1703             <complexType>
1704                 <sequence>
1705                     <element name="DeviceId" type="xsd:string"/>
1706                     <element name="DeviceName" type="xsd:string"/>
1707                 </sequence>
1708             </complexType>
1709         </element>
1710         <element name="3rdPartyAuthen" minOccurs="0">
1711             <complexType>
1712                 <sequence>
1713                     <element name="3rdPartyAuthenType"
1714 type="xsd:string"/>
1715                 </sequence>
1716             </complexType>
1717         </element>
1718     </sequence>
1719 </complexType>
1720 <complexType name="ObjectAttributeType">
1721     <sequence>
1722         <element name="ObjectType" type="xsd:string"/>
1723         <element name="ObjectId" type="xsd:string"/>
1724         <element name="ObjectName" type="xsd:string"/>
1725         <element name="ParentId" type="xsd:string" minOccurs="0"/>
1726         <element name="DeviceId" type="xsd:string" minOccurs="0"/>
1727         <element name="DeviceName" type="xsd:string" minOccurs="0"/>
1728         <element name="AccessRight">
1729             <complexType>
1730                 <sequence>
1731                     <element name="Read" type="xsd:string"
1732 minOccurs="0"/>
1733                     <element name="Write" type="xsd:string"
1734 minOccurs="0"/>
1735                     <element name="Hide" type="xsd:string"
1736 minOccurs="0"/>
1737                 </sequence>
1738             </complexType>
1739         </element>
1740         <element name="CreateTime" type="xsd:string" minOccurs="0"/>
1741         <element name="LastAccessTime" type="xsd:string" minOccurs="0"/>
1742         <element name="LastWriteTime" type="xsd:string" minOccurs="0"/>
1743         <element name="Size" type="xsd:string" minOccurs="0"/>
1744         <element name="Num_SubDirectories" type="unsignedInt" minOccurs="0"/>
1745         <element name="Num_SubFiles" type="unsignedInt" minOccurs="0"/>
1746     </sequence>
1747 </complexType>
1748 </types>
1749 <message name="GetAuthenticationKeyRequest">
1750     <part name="InstanceId" type="Type_InstanceId"/>
1751     <part name="UserAuthenticationInfo" type="Type_UserAuthenticationInfo"/>
1752 </message>
1753 <message name="GetAuthenticationKeyResponse">

```

```
1754     <part name="ReturnCode" type="xsd:unsignedInt"/>
1755     <part name="AuthenticationKey" type="xsd:string"/>
1756 </message>
1757 <message name="GetSortCapabilityRequest">
1758     <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1759 </message>
1760 <message name="GetSortCapabilityResponse">
1761     <part name="ReturnCode" type="unsignedInt"/>
1762     <part name="SortCaps" ref="igrs:Type_SortCapability"/>
1763 </message>
1764 <message name="GetSearchCapabilityRequest">
1765     <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1766 </message>
1767 <message name="GetSearchCapabilityResponse">
1768     <part name="ReturnCode" type="unsignedInt"/>
1769     <part name="SearchCaps" ref="igrs:Type_SearchCapability"/>
1770 </message>
1771 <message name="BrowseRequest">
1772     <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1773     <part name="ObjectId" type="Type_ObjectId"/>
1774     <part name="BrowseFilter" type="Type_FilterRule"/>
1775     <part name="StartOffset" type="Type_Count"/>
1776     <part name="RequestedCount" type="Type_Count"/>
1777     <part name="SortRule" type="Type_SortRule"/>
1778 </message>
1779 <message name="BrowseResponse">
1780     <part name="ReturnCode" type="unsignedInt"/>
1781     <part name="Result" type="Type_ObjectList"/>
1782     <part name="NumberReturned" type="Type_Count"/>
1783     <part name="NumberTotalMatched" type="Type_Count"/>
1784 </message>
1785 <message name="GetAttributeRequest">
1786     <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1787     <part name="ObjectId" type="Type_ObjectId"/>
1788 </message>
1789 <message name="GetAttributeResponse">
1790     <part name="ReturnCode" type="unsignedInt"/>
1791     <part name="ObjectAttribute" type="Type_ObjectAttribute"/>
1792 </message>
1793 <message name="SetAttributeRequest">
1794     <part name="AuthenticationKey" tpe="Type_AuthenticationKey"/>
1795     <part name="ObjectId" type="Type_ObjectId"/>
1796     <part name="ObjectAttribute" type="Type_ObjectAttribute"/>
1797 </message>
1798 <message name="SetAttributeResponse">
1799     <part name="ReturnCode" type="unsignedInt"/>
1800 </message>
1801 <message name="SearchRequest">
1802     <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1803     <part name="ObjectIdList" type="Type_ObjectIdList"/>
1804     <part name="SearchRule" type="Type_FilterRule"/>
1805     <part name="StartOffset" type="Type_Count"/>
1806     <part name="RequestedCount" type="Type_Count"/>
1807     <part name="SortRule" ref="igrs:Type_SortRule"/>
1808 </message>
1809 <message name="SearchResponse">
1810     <part name="ReturnCode" type="unsignedInt"/>
1811     <part name="Result" type="Type_ObjectList"/>
1812     <part name="NumberReturned" type="Type_Count"/>
1813     <part name="NumberTotalMatched" type="Type_Count"/>
```

```
1814     </message>
1815     <message name="GetBrowseFilterRequest">
1816         <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1817     </message>
1818     <message name="GetBrowseFilterResponse">
1819         <part name="ReturnCode" type="unsignedInt"/>
1820         <part name="BrowseFilter" type="Type_FilterRule"/>
1821     </message>
1822     <message name="SetBrowseFilterRequest">
1823         <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1824         <part name="BrowseFilter" type="Type_FilterRule"/>
1825     </message>
1826     <message name="SetBrowseFilterResponse">
1827         <part name="ReturnCode" type="unsignedInt"/>
1828     </message>
1829     <message name="NewRequest">
1830         <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1831         <part name="ParentId" type="Type_ObjectId"/>
1832         <part name="ObjectAttribute" type="Type_ObjectAttribute"/>
1833     </message>
1834     <message name="NewResponse">
1835         <part name="ReturnCode" type="unsignedInt"/>
1836         <part name="ObjectId" type="Type_ObjectId"/>
1837     </message>
1838     <message name="CopyRequest">]
1839         <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1840         <part name="SourceObjectId" type="Type_ObjectId"/>
1841         <part name="DestParentId" type="Type_ObjectId"/>
1842     </message>
1843     <message name="CopyResponse">
1844         <part name="ReturnCode" type="unsignedInt"/>
1845         <part name="DestObjectId" type="Type_ObjectId"/>
1846         <part name="DestObjectAttribute" type="Type_ObjectAttribute"/>
1847     </message>
1848     <message name="MoveRequest">
1849         <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1850         <part name="SourceObjectId" type="Type_ObjectId"/>
1851         <part name="DestParentId" type="Type_ObjectId"/>
1852     </message>
1853     <message name="MoveResponse">
1854         <part name="ReturnCode" type="unsignedInt"/>
1855         <part name="DestObjectId" type="Type_ObjectId"/>
1856         <part name="DestObjectAttribute" type="Type_ObjectAttribute"/>
1857     </message>
1858     <message name="DeleteRequest">
1859         <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1860         <part name="ObjectId" type="Type_ObjectId"/>
1861         <part name="DeleteMode" type="Type_DeleteMode"/>
1862     </message>
1863     <message name="DeleteResponse">
1864         <part name="ReturnCode" type="unsignedInt"/>
1865     </message>
1866     <message name="PrepareforDownloadRequest">
1867         <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1868         <part name="SourceObjectIdList" type="Type_ObjectIdList"/>
1869     </message>
1870     <message name="PrepareforDownloadResponse">
1871         <part name="ReturnCode" type="unsignedInt"/>
1872         <part name="SourceObjectURITreeList" type="Type_ObjectURITreeList"/>
1873     </message>
```

```

1874     <message name="PrepareforUploadRequest">
1875         <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1876         <part name="ObjectAttribute" type="Type_ObjectAttribute"/>
1877         <part name="DestParentId" type="Type_ObjectId"/>
1878     </message>
1879     <message name="PrepareforUploadResponse">
1880         <part name="ReturnCode" type="unsignedInt"/>
1881         <part name="DestParentURI" type="Type_ObjectURI"/>
1882     </message>
1883     <message name="SubscribeObjectChangeRequest">
1884         <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1885         <part name="ObjectId" type="Type_ObjectId"/>]
1886     </message>
1887     <message name="SubscribeObjectChangeResponse">
1888         <part name="ReturnCode" type="unsignedInt"/>
1889     </message>
1890     <message name="UnSubscribeObjectChangeRequest">
1891         <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1892         <part name="ObjectId" type="Type_ObjectId"/>
1893     </message>
1894     <message name="UnSubscribeObjectChangeResponse">
1895         <part name="ReturnCode" type="unsignedInt"/>
1896     </message>
1897     <message name="SubscribeServiceAttributeRequest">
1898         <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1899         <part name="ServiceAttributeName" type="Type_ServiceAttributeName"/>
1900     </message>
1901     <message name="SubscribeServiceAttributeResponse">
1902         <part name="ReturnCode" type="unsignedInt"/>
1903         <part name="SubscriptionId" type="Type_SubscriptionId"/>
1904     </message>
1905     <message name="UnSubscribeServiceAttributedRequest">
1906         <part name="AuthenticationKey" type="Type_AuthenticationKey"/>
1907         <part name="SubscriptionId" type="Type_SubscriptionId"/>
1908     </message>
1909     <message name="UnSubscribeServiceAttributeResponse">
1910         <part name="ReturnCode" type="unsignedInt"/>
1911     </message>
1912     <portType name="FileAccessManagementPortType">
1913         <operation name="GetAuthenticationKey">
1914             <input message="igrs:GetAuthenticationKeyRequest"/>
1915             <output message="igrs:GetAuthenticationKeyResponse"/>
1916         </operation>
1917         <operation name="GetSortCapability">
1918             <input message="igrs:GetSortCapabilityRequest"/>
1919             <output message="igrs:GetSortCapabilityResponse"/>
1920         </operation>
1921         <operation name="GetSearchCapability">
1922             <input message="igrs:GetSearchCapabilityRequest"/>
1923             <output message="igrs:GetSearchCapabilityResponse"/>
1924         </operation>
1925         <operation name="Browse">
1926             <input message="igrs:BrowseRequest"/>
1927             <output message="igrs:BrowseResponse"/>
1928         </operation>
1929         <operation name="GetAttribute">
1930             <input message="igrs:GetAttributeRequest"/>
1931             <output message="igrs:GetAttributeResponse"/>
1932         </operation>
1933         <operation name="SetAttribute">

```

```
1934         <input message="igrs:SetAttributeRequest"/>
1935         <output message="igrs:SetAttributeResponse"/>
1936     </operation>
1937
1938     <operation name="Search">
1939         <input message="igrs:SearchRequest"/>
1940         <output message="igrs:SearchResponse"/>
1941     </operation>
1942     <operation name="GetBrowseFilter">
1943         <input message="igrs:GetBrowseFilterRequest"/>
1944         <output message="igrs:GetBrowseFilterResponse"/>
1945     </operation>
1946     <operation name="SetBrowseFilter">
1947         <input message="igrs: SetBrowseFilterRequest"/>
1948         <output message="igrs: SetBrowseFilterResponse"/>
1949     </operation>
1950     <operation name="New">
1951         <input message="igrs: NewRequest"/>
1952         <output message="igrs: NewResponse"/>
1953     </operation>
1954     <operation name="Copy">
1955         <input message="igrs:CopyRequest"/>
1956         <output message="igrs:CopyResponse"/>
1957     </operation>
1958     <operation name="Move">
1959         <input message="igrs:MoveRequest"/>
1960         <output message="igrs:MoveResponse"/>
1961     </operation>
1962     <operation name="Delete">
1963         <input message="igrs:DeleteRequest"/>
1964         <output message="igrs:DeleteResponse"/>
1965     </operation>
1966     <operation name="PrepareforDownload">
1967         <input message="igrs:PrepareforDownloadRequest"/>
1968         <output message="igrs:PrepareforDownloadResponse"/>
1969     </operation>
1970     <operation name="PrepareforUpload">
1971         <input message="igrs:PrepareforUploadRequest"/>
1972         <output message="igrs:PrepareforUploadResponse"/>
1973     </operation>
1974     <operation name="SubscribeObjectChange">
1975         <input message="igrs:SubscribeObjectChangeRequest"/>
1976         <output message="igrs:SubscribeObjectChangeResponse"/>
1977     </operation>
1978     <operation name="UnsubscribeObjectChange">
1979         <input message="igrs:UnsubscribeObjectChangeRequest"/>
1980         <output message="igrs:UnsubscribeObjectChangeResponse"/>
1981     </operation>
1982     <operation name="SubscribeServiceAttribute">
1983         <input message="igrs:SubscribeServiceAttributeRequest"/>
1984         <output message="igrs:SubscribeServiceAttributeResponse"/>
1985     </operation>
1986     <operation name="UnSubscribeServiceAttribute">
1987         <input message="igrs:UnSubscribeServiceAttributeRequest"/>
1988         <output message="igrs:UnSubscribeServiceAttributeResponse"/>
1989     </operation>
1990     <igrs:serviceAttribute name="SortCaps" type="Type_SortCapability" minOccurs="0"
1991     notifiable="true"/>
1992     <igrs:serviceAttribute name="SearchCaps" type="Type_SearchCapability" minOccurs="0"
1993     notifiable="true"/>
```

```
1994     </portType>
1995     <binding name="FileAccessManagementServiceIGRSPipe" type="igrs:FileAccessManagementPortType">
1996         <igrs:binding transport="http://www.igrs.org/igrs/igrspipe" style="rpc"/>
1997         <operation name="GetAuthenticationKey">
1998             <input>
1999                 <soap:body use="encoded"
2000 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2001             </input>
2002             <output>
2003                 <soap:body use="encoded"
2004 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2005             </output>
2006         </operation>
2007         <operation name="GetSortCapability">
2008             <input>
2009                 <soap:body use="encoded"
2010 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2011             </input>
2012             <output>
2013                 <soap:body use="encoded"
2014 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2015             </output>
2016         </operation>
2017         <operation name="GetSearchCapability">
2018             <input>
2019                 <soap:body use="encoded"
2020 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2021             </input>
2022             <output>
2023                 <soap:body use="encoded"
2024 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2025             </output>
2026         </operation>
2027         <operation name="Browse">
2028             <input>
2029                 <soap:body use="encoded"
2030 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2031             </input>
2032             <output>
2033                 <soap:body use="encoded"
2034 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2035             </output>
2036         </operation>
2037         <operation name="GetAttribute">
2038             <input>
2039                 <soap:body use="encoded"
2040 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2041             </input>
2042             <output>
2043                 <soap:body use="encoded"
2044 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2045             </output>
2046         </operation>
2047         <operation name="SetAttribute">
2048             <input>
2049                 <soap:body use="encoded"
2050 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2051             </input>
2052             <output>
2053                 <soap:body use="encoded"
```

```
2054 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2055     </output>
2056     </operation>
2057     <operation name="Search">
2058         <input>
2059             <soap:body use="encoded"
2060 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2061             </input>
2062             <output>
2063                 <soap:body use="encoded"
2064 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2065                 </output>
2066             </operation>
2067             <operation name="GetBrowseFilter">
2068                 <input>
2069                     <soap:body use="encoded"
2070 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2071                     </input>
2072                     <output>
2073                         <soap:body use="encoded"
2074 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2075                         </output>
2076                     </operation>
2077                     <operation name="SetBrowseFilter">
2078                         <input>
2079                             <soap:body use="encoded"
2080 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2081                             </input>
2082                             <output>
2083                                 <soap:body use="encoded"
2084 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2085                                 </output>
2086                             </operation>
2087                             <operation name="New">
2088                                 <input>
2089                                     <soap:body use="encoded"
2090 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2091                                     </input>
2092                                     <output>
2093                                         <soap:body use="encoded"
2094 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2095                                         </output>
2096                                     </operation>
2097                                     <operation name="Copy">
2098                                         <input>
2099                                             <soap:body use="encoded"
2100 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2101                                             </input>
2102                                             <output>
2103                                                 <soap:body use="encoded"
2104 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2105                                                 </output>
2106                                             </operation>
2107                                             <operation name="Move">
2108                                                 <input>
2109                                                     <soap:body use="encoded"
2110 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2111                                                     </input>
2112                                                     <output>
2113                                                         <soap:body use="encoded"
```

```
2114 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2115     </output>
2116     </operation>
2117     <operation name="Delete">
2118         <input>
2119             <soap:body use="encoded"
2120 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2121                 </input>
2122                 <output>
2123                     <soap:body use="encoded"
2124 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2125                         </output>
2126                 </operation>
2127                 <operation name="PrepareforDownload">
2128                     <input>
2129                         <soap:body use="encoded"
2130 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2131                             </input>
2132                             <output>
2133                                 <soap:body use="encoded"
2134 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2135                                     </output>
2136                                 </operation>
2137                                 <operation name="PrepareforUpload">
2138                                     <input>
2139                                         <soap:body use="encoded"
2140 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2141                                             </input>
2142                                             <output>
2143                                                 <soap:body use="encoded"
2144 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2145                                                     </output>
2146                                                 </operation>
2147                                                 <operation name="SubscribeObjectChange">
2148                                                     <input>
2149                                                         <soap:body use="encoded"
2150 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2151                                                             </input>
2152                                                             <output>
2153                                                                 <soap:body use="encoded"
2154 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2155                                                                     </output>
2156                                                                 </operation>
2157                                                                 <operation name="UnsubscribeObjectChange">
2158                                                                     <input>
2159                                                                         <soap:body use="encoded"
2160 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2161                                                                             </input>
2162                                                                             <output>
2163                                                                                 <soap:body use="encoded"
2164 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2165                                                                                     </output>
2166                                                                                 </operation>
2167                                                                                 <operation name="SubscribeServiceAttribute">
2168                                                                                     <input>
2169                                                                                         <soap:body use="encoded"
2170 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2171                                                                                             </input>
2172                                                                                             <output>
2173                                                                                                 </output>
```

```
2174             <soap:body use="encoded"
2175 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2176             </output>
2177         </operation>
2178         <operation name="UnsubscribeServiceAttribute">
2179             <input>
2180                 <soap:body use="encoded"
2181 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2182                 </input>
2183                 <output>
2184                     <soap:body use="encoded"
2185 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2186                     </output>
2187                 </operation>
2188             </binding>
2189             <service name="FileAccessManagementService">
2190                 <port name="FileAccessManagementService"
2191 binding="igrs:FileAccessManagementServiceIGRSPipe">
2192                     <igrs:address location="http://www.igrs.org/igrs/igrspipe/FileAccessManagement"/>
2193                 </port>
2194             </service>
2195 </definitions>
```

2196
2197
2198

Annex D
(Normative Annex)
File Profile – FileConnectionManagement Service Description.wsd

```

2199 <?xml version="1.0" encoding="UTF-8"?>
2200 <definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
2201 xmlns:http="http://schemas.xmlsoap.org/wsdl/http/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2202 xmlns:igrsExt="http://www.igrs.org/igrs/ServiceDescription"
2203 xmlns:igrs="http://www.igrs.org/igrs/FileConnectionManagement"
2204 targetNamespace="http://www.igrs.org/igrs/FileConnectionManagement" name="FileConnectionManagement">
2205   <types>
2206     <element name="Type_ProtocolInfoList" type="PortocolInfoListType"/>
2207     <element name="Type_ProtocolInfo" type="ProtocolInfoType"/>
2208     <element name="Type_ConnectionId" type="xsd:unsignedInt"/>
2209     <element name="Type_ServiceAttributeName" type="xsd:string" minOccurs="0"/>
2210     <element name="Type_SubscriptionId" type="xsd:string" minOccurs="0"/>
2211     <element name="Type_IPList" minOccurs="0">
2212       <complexType>
2213         <sequence>
2214           <element name="IP" type="xsd:string" maxOccurs="unbounded"/>
2215         </sequence>
2216       </complexType>
2217     </element>
2218     <complexType name="PortocolInfoListType">
2219       <sequence>
2220         <element name="ProtocolInfo" type="ProtocolInfoType" minOccurs="0"
2221 maxOccurs="unbounded"/>
2222       </sequence>
2223     </complexType>
2224     <complexType name="ProtocolInfoType">
2225       <sequence>
2226         <element name="TransportProtocol">
2227           <complexType>
2228             <sequence>
2229               <element name="Port" type="xsd:string"
2230 minOccurs="0" maxOccurs="unbounded"/>
2231             </sequence>
2232             <attribute name="Name" type="xsd:string"/>
2233           </complexType>
2234         </element>
2235         <element name="ParameterList" type="xsd:string" minOccurs="0"/>
2236         <element name="AdditionalInfo" type="xsd:string" minOccurs="0"/>
2237       </sequence>
2238     </complexType>
2239   </types>
2240   <message name="GetProtocolInfoRequest"/>
2241   <message name="GetProtocolInfoResponse">
2242     <part name="ReturnCode" type="xsd:unsignedInt"/>
2243     <part name="ProtocolInfoList" ref="igrs:Type_ProtocolInfoList"/>
2244     <part name="IPList" ref="igrs:Type_IPList"/>
2245   </message>
2246   <message name="PrepareforConnectionRequest">
2247     <part name="RemoteProtocolInfo" type="Type_ProtocolInfo"/>
2248   </message>
2249   <message name="PrepareforConnectionResponse">
2250     <part name="ReturnCode" type="xsd:unsignedInt"/>
2251     <part name="ConnectionId" type="Type_ConnectionId"/>
2252   </message>
2253   <message name="ReleaseConnectionRequest">

```

```

2254         <part name="ConnectionId" type="Type_ConnectionId"/>
2255     </message>
2256     <message name="ReleaseConnectionResponse">
2257         <part name="ReturnCode" type="xsd:unsignedInt"/>
2258     </message>
2259     <message name="SubscribeServiceAttributeRequest">
2260         <part name="ServiceAttributeName" type="Type_ServiceAttributeName"/>
2261     </message>
2262     <message name="SubscribeServiceAttributeResponse">
2263         <part name="ReturnCode" type="xsd:unsignedInt"/>
2264         <part name="SubscribeId" type="Type_SubscriptionId"/>
2265     </message>
2266     <message name="UnsubscribeServiceAttributeRequest">
2267         <part name="SubscriptionId" type="Type_SubscriptionId"/>
2268     </message>
2269     <message name="UnsubscribeServiceAttributeResponse">
2270         <part name="ReturnCode" type="xsd:unsignedInt"/>
2271     </message>
2272
2273     <portType name="FileConnectionManagementPortType">
2274         <operation name="GetProtocolInfo">
2275             <input message="igrs:GetProtocolInfoRequest"/>
2276             <output message="igrs:GetProtocolInfoResponse"/>
2277         </operation>
2278         <operation name="PrepareforConnection">
2279             <input message="igrs:PrepareforConnectionRequest"/>
2280             <output message="igrs:PrepareforConnectionResponse"/>
2281         </operation>
2282         <operation name="ReleaseConnection">
2283             <input message="igrs:ReleaseConnectionRequest"/>
2284             <output message="igrs:ReleaseConnectionResponse"/>
2285         </operation>
2286         <operation name="SubscribeServiceAttribute">
2287             <input message="igrs:SubscribeServiceAttributeRequest"/>
2288             <output message="igrs:SubscribeServiceAttributeResponse"/>
2289         </operation>
2290         <operation name="UnsubscribeServiceAttribute">
2291             <input message="igrs:UnsubscribeServiceAttributeRequest"/>
2292             <output message="igrs:UnsubscribeServiceAttributeResponse"/>
2293         </operation>
2294
2295         <igrs:serviceAttribute name="ProtocolInfoList" type="Type_ProtocolInfoList" notifiable="true"/>
2296         <igrs:serviceAttribute name="IPList" type="Type_IPList" notifiable="true"/>
2297
2298     </portType>
2299
2300     <binding name="FileConnectionManagementServiceIGRSPipe"
2301 type="igrs:FileConnectionManagementPortType">
2302         <igrs:binding transport="http://www.igrs.org/igrs/igrspipe" style="rpc"/>
2303         <operation name="GetProtocolInfo">
2304             <input>
2305                 <soap:body use="encoded"
2306 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2307             </input>
2308             <output>
2309                 <soap:body use="encoded"
2310 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2311             </output>
2312         </operation>
2313         <operation name="PrepareforConnection">

```

```
2314         <input>
2315         <soap:body use="encoded"
2316 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2317         </input>
2318         <output>
2319         <soap:body use="encoded"
2320 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2321         </output>
2322     </operation>
2323     <operation name="ReleaseConnection">
2324         <input>
2325         <soap:body use="encoded"
2326 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2327         </input>
2328         <output>
2329         <soap:body use="encoded"
2330 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2331         </output>
2332     </operation>
2333     <operation name="SubscribeServiceAttribute">
2334         <input>
2335         <soap:body use="encoded"
2336 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2337         </input>
2338         <output>
2339         <soap:body use="encoded"
2340 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2341         </output>
2342     </operation>
2343     <operation name="UnsubscribeServiceAttribute">
2344         <input>
2345         <soap:body use="encoded"
2346 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2347         </input>
2348         <output>
2349         <soap:body use="encoded"
2350 encodingStyle="http://www.igrs.org/igrs/igrspipe/encoding"/>
2351         </output>
2352     </operation>
2353 </binding>
2354 <service name="FileConnectionManagementService">
2355     <port name="FileConnectionManagementService"
2356 binding="igrs:FileConnectionManagementServiceIGRSPipe">
2357         <igrs:address location="http://www.igrs.org/igrs/igrspipe/FileConnectionManagement"/>
2358     </port>
2359 </service>
2360 </definitions>
```

2361

References

2362

[1] W3C WSDL 1.1 □ Web Service Description Language Version 1.1

2363

[2] IETF RFC 2234 □ Extend BNF normal form used as grammar regulation