

Modus: Alle Zeilen
Linker Ursprungsordner: W:\BCAST\OMA\CRs\own\OMA-IOP-TTCN-2008-xxxx-CR_Flute_TSI\orig
Rechter Ursprungsordner: W:\BCAST\OMA\CRs\own\OMA-IOP-TTCN-2008-xxxx-CR_Flute_TSI\modified TTCN-3 sources

Datei: AtsBCast_ServiceGuide_Functions.ttcn

1	/**	=	1	/**
2	*		2	*
3	* @author		3	* @author
4	* ETSI CTI BCAST CON Project		4	* ETSI CTI BCAST CON Project
5	* @version		5	* @version
6	* \$Id\$		6	* \$Id\$
7	* @desc		7	* @desc
8	* This module specifies some function for the service guide.		8	* This module specifies some function for the service guide.
9	*/		9	*/
10			10	
11	module AtsBCast_ServiceGuide_Functions {		11	module AtsBCast_ServiceGuide_Functions {
12	import from AtsBCast_TestSystem all;		12	import from AtsBCast_TestSystem all;
13	import from AtsBCast_ModuleParameters { modulepar PX_SIMULATE_TC };		13	import from AtsBCast_ModuleParameters { modulepar PX_SIMULATE_TC };
14	import from LibBCast_UpperTesterPrimitives_TypesAndValues all;		14	import from LibBCast_UpperTesterPrimitives_TypesAndValues all;
15	import from LibBCast_ServiceGuide_Templates all;		15	import from LibBCast_ServiceGuide_Templates all;
16	import from LibBCast_ServiceGuide_TypesAndValues all;		16	import from LibBCast_ServiceGuide_TypesAndValues all;
17	import from LibBCast_Common_TypesAndValues all;		17	import from LibBCast_Common_TypesAndValues all;
18	import from LibCommon_DataStrings all;		18	import from LibCommon_DataStrings all;
19	import from LibCommon_BasicTypesAndValues all;		19	import from LibCommon_BasicTypesAndValues all;
20	import from LibBCast_ServicePrimitives_TypesAndValues all;		20	import from LibBCast_ServicePrimitives_TypesAndValues all;
21			21	
22	group FDT {		22	group FDT {
23			23	
24	// change 6 (WK18): dynamic setting of TOI number		24	// change 6 (WK18): dynamic setting of TOI number
25	/**		25	/**
26	*		26	*
27	* @desc		27	* @desc
28	* This functions create a FDT for a ServiceGuideDeliveryUnit		28	* This functions create a FDT for a ServiceGuideDeliveryUnit
29	* @param		29	* @param
30	* p_SGDU_ID The Id of the ServiceGuideDeliveryUnit		30	* p_SGDU_ID The Id of the ServiceGuideDeliveryUnit
31	* @param		31	* @param
32	* p_ContentEncoding The encoding of the		32	* p_ContentEncoding The encoding of the
33	* ServiceGuideDeliveryUnit		33	* ServiceGuideDeliveryUnit
34	* @param		34	* @param
35	* p_Expires The expire time how long the FDT is vaild.		35	* p_Expires The expire time how long the FDT is vaild.
36	* @param		36	* @param
37	* p_SGDU_TOI the TOI number		37	* p_SGDU_TOI the TOI number
38	* @return		38	* @return
39	* The File Delivery Table Insstance		39	* The File Delivery Table Insstance
40	*/		40	*/
41	function f_createFDT4SGDU(41	function f_createFDT4SGDU(
42	AnyUri p_SGDU_Id,		42	AnyUri p_SGDU_Id,
43	template charstring p_ContentEncoding,		43	template charstring p_ContentEncoding,
44	charstring p_Expires,		44	charstring p_Expires,
45	UInt p_SGDU_TOI) return FDT_InstanceType {		45	UInt p_SGDU_TOI) return FDT_InstanceType {
46	var LibBCast_ServicePrimitives_TypesAndValues.FileList v_FileList;		46	var LibBCast_ServicePrimitives_TypesAndValues.FileList v_FileList;
47			47	
48	//note: we assume for now, that all fragments are included in one file		48	//note: we assume for now, that all fragments are included in one file
49	var LibBCast_ServicePrimitives_TypesAndValues.FileType v_File :=		49	var LibBCast_ServicePrimitives_TypesAndValues.FileType v_File :=
50	valueof(m_def_FileType(50	valueof(m_def_FileType(
51	p_SGDU_Id,		51	p_SGDU_Id,
52	p_SGDU_TOI,		52	p_SGDU_TOI,
53	"application/vnd.oma.bcast.sgdu",		53	"application/vnd.oma.bcast.sgdu",
54	p_ContentEncoding));		54	p_ContentEncoding));
55			55	
56	v_FileList[0] := v_File;		56	v_FileList[0] := v_File;
57	var FDT_InstanceType v_FDT :=		57	var FDT_InstanceType v_FDT :=
58	valueof(m_def_FDT_Instance(v_FileList, p_Expires, omit, 0));		58	valueof(m_def_FDT_Instance(v_FileList, p_Expires, omit, 0));
59			59	
60	return v_FDT;		60	return v_FDT;
61	}		61	}
62			62	
63	// change 6 (WK18): dynamic setting of TOI number		63	// change 6 (WK18): dynamic setting of TOI number
64	/**		64	/**
65	*		65	*
66	* @desc		66	* @desc

Datei: AtsBCast_ServiceGuide_Functions.ttcn (Fortsetzung)

67	* Creates a File Dilivery Table Instance (FDT) for a Service	67	* Creates a File Dilivery Table Instance (FDT) for a Service
68	* Guide Delivery Descriptor (SGDD).	68	* Guide Delivery Descriptor (SGDD).
69	* @param	69	* @param
70	* p_SGDD_Id the Service Guide Delivery Descriptor id.	70	* p_SGDD_Id the Service Guide Delivery Descriptor id.
71	* @param	71	* @param
72	* p_ContentEncoding the encoding of the SGDD	72	* p_ContentEncoding the encoding of the SGDD
73	* @param	73	* @param
74	* p_Expires the expire time how long this FDT is vaild.	74	* p_Expires the expire time how long this FDT is vaild.
75	* @param	75	* @param
76	* p_SGDD_TOI the TOI number	76	* p_SGDD_TOI the TOI number
77	* @return	77	* @return
78	* the File Delivery Table Instance (FDT)	78	* the File Delivery Table Instance (FDT)
79	*/	79	*/
80	function f_createFDT4SGDD(80	function f_createFDT4SGDD(
81	AnyUri p_SGDD_Id,	81	AnyUri p_SGDD_Id,
82	template charstring p_ContentEncoding,	82	template charstring p_ContentEncoding,
83	charstring p_Expires,	83	charstring p_Expires,
84	UInt p_SGDD_TOI) return FDT_InstanceType {	84	UInt p_SGDD_TOI) return FDT_InstanceType {
85	var LibBCast_ServicePrimitives_TypesAndValues.FileList v_FileList;	85	var LibBCast_ServicePrimitives_TypesAndValues.FileList v_FileList;
86		86	
87	var LibBCast_ServicePrimitives_TypesAndValues.FileType v_File :=	87	var LibBCast_ServicePrimitives_TypesAndValues.FileType v_File :=
88	valueof(m_def_FileType(88	valueof(m_def_FileType(
89	p_SGDD_Id,	89	p_SGDD_Id,
90	p_SGDD_TOI,	90	p_SGDD_TOI,
91	"application/vnd.oma.bcast.sgdd+xml",	91	"application/vnd.oma.bcast.sgdd+xml",
92	p_ContentEncoding));	92	p_ContentEncoding));
93		93	
94	v_FileList[0] := v_File;	94	v_FileList[0] := v_File;
95	var FDT_InstanceType v_FDT :=	95	var FDT_InstanceType v_FDT :=
96	valueof(m_def_FDT_Instance(v_FileList, p_Expires, omit, 0));	96	valueof(m_def_FDT_Instance(v_FileList, p_Expires, omit, 0));
97		97	
98	return v_FDT;	98	return v_FDT;
99	}	99	}
100		100	
101	// change 04 (WK 6): creates FDT instance for DVB-H bootstrapping	101	// change 04 (WK 6): creates FDT instance for DVB-H bootstrapping
102	/**	102	/**
103	*	103	*
104	* @desc	104	* @desc
105	* Creates a File Dilivery Table Instance (FDT) for the	105	* Creates a File Dilivery Table Instance (FDT) for the
106	* DVB-H bootstrap.	106	* DVB-H bootstrap.
107	* @param	107	* @param
108	* p_Expires the expire time how long this FDT is vaild.	108	* p_Expires the expire time how long this FDT is vaild.
109	* @return	109	* @return
110	* the File Delivery Table Instance (FDT)	110	* the File Delivery Table Instance (FDT)
111	*/	111	*/
112	function f_createFDT4DVBH(charstring p_Expires) return FDT_InstanceType {	112	function f_createFDT4DVBH(charstring p_Expires) return FDT_InstanceType {
113	var UInt v_TOI := 1;	113	var UInt v_TOI := 1;
114	var LibBCast_ServicePrimitives_TypesAndValues.FileList v_FileList;	114	var LibBCast_ServicePrimitives_TypesAndValues.FileList v_FileList;
115		115	
116	var LibBCast_ServicePrimitives_TypesAndValues.FileType v_FileDiscoveryXML :=	116	var LibBCast_ServicePrimitives_TypesAndValues.FileType v_FileDiscoveryXML :=
117	valueof(m_def_FileType("esg-discovery.xml", v_TOI, "text/xml", omit	117	valueof(m_def_FileType("esg-discovery.xml", v_TOI, "text/xml", omit
118		118	
119	var LibBCast_ServicePrimitives_TypesAndValues.FileType v_DiscoveryBIN :=	119	var LibBCast_ServicePrimitives_TypesAndValues.FileType v_DiscoveryBIN :=
120	valueof(m_def_FileType("esg-discovery.bin", v_TOI + 1, "application	120	valueof(m_def_FileType("esg-discovery.bin", v_TOI + 1, "application
121		121	
122	v_FileList[0] := v_FileDiscoveryXML;	122	v_FileList[0] := v_FileDiscoveryXML;
123	v_FileList[1] := v_DiscoveryBIN;	123	v_FileList[1] := v_DiscoveryBIN;
124		124	
125	var FDT_InstanceType v_FDT :=	125	var FDT_InstanceType v_FDT :=
126	valueof(m_def_FDT_Instance(v_FileList, p_Expires, c	126	valueof(m_def_FDT_Instance(v_FileList, p_Expires, c
127		127	
128	return v_FDT;	128	return v_FDT;
129	}	129	}
130	}	130	}
131		131	
132	group SGDU {	132	group SGDU {
133	// CR 26/8: generic function for adding XML fragments	133	// CR 26/8: generic function for adding XML fragments
134	// change 7 (WK18): improved source code for CR 2007-0026 change 8	134	// change 7 (WK18): improved source code for CR 2007-0026 change 8
135	/**	135	/**
136	* @desc This function adds a XML fragment at the last element of a SGDU.	136	* @desc This function adds a XML fragment at the last element of a SGDU.
137	* @param	137	* @param

Datei: AtsBCast_ServiceGuide_Functions.ttcn (Fortsetzung)

138	* p_SGDU The SGDU which shoud add the service fragment.	138	* p_SGDU The SGDU which shoud add the service fragment.
139	* @param	139	* @param
140	* p_xmlFragment The XML fragment to add	140	* p_xmlFragment The XML fragment to add
141	*/	141	*/
142	function f_addXMLFragment(inout ServiceGuideDeliveryUnit p_SGDU, XmlFragment p_xmlFragment)	142	function f_addXMLFragment(inout ServiceGuideDeliveryUnit p_SGDU, XmlFragment p_xmlFragment)
143	var integer v_size := sizeof (p_SGDU);	143	var integer v_size := sizeof (p_SGDU);
144	var UInt v_version := 0;	144	var UInt v_version := 0;
145		145	
146	if (ischosen(p_xmlFragment.Service)) {	146	if (ischosen(p_xmlFragment.Service)) {
147	v_version := p_xmlFragment.Service.version;	147	v_version := p_xmlFragment.Service.version;
148	}	148	}
149	else if (ischosen(p_xmlFragment.Content)) {	149	else if (ischosen(p_xmlFragment.Content)) {
150	v_version := p_xmlFragment.Content.version;	150	v_version := p_xmlFragment.Content.version;
151	}	151	}
152	else if (ischosen(p_xmlFragment.Schedule)) {	152	else if (ischosen(p_xmlFragment.Schedule)) {
153	v_version := p_xmlFragment.Schedule.version;	153	v_version := p_xmlFragment.Schedule.version;
154	}	154	}
155	else if (ischosen(p_xmlFragment.Access)) {	155	else if (ischosen(p_xmlFragment.Access)) {
156	v_version := p_xmlFragment.Access.version;	156	v_version := p_xmlFragment.Access.version;
157	}	157	}
158	else if (ischosen(p_xmlFragment.PurchaseItem)) {	158	else if (ischosen(p_xmlFragment.PurchaseItem)) {
159	v_version := p_xmlFragment.PurchaseItem.version;	159	v_version := p_xmlFragment.PurchaseItem.version;
160	}	160	}
161	else if (ischosen(p_xmlFragment.PurchaseData)) {	161	else if (ischosen(p_xmlFragment.PurchaseData)) {
162	v_version := p_xmlFragment.PurchaseData.version;	162	v_version := p_xmlFragment.PurchaseData.version;
163	}	163	}
164	else if (ischosen(p_xmlFragment.PurchaseChannel)) {	164	else if (ischosen(p_xmlFragment.PurchaseChannel)) {
165	v_version := p_xmlFragment.PurchaseChannel.version;	165	v_version := p_xmlFragment.PurchaseChannel.version;
166	}	166	}
167	else if (ischosen(p_xmlFragment.InteractivityData)) {	167	else if (ischosen(p_xmlFragment.InteractivityData)) {
168	v_version := p_xmlFragment.InteractivityData.version;	168	v_version := p_xmlFragment.InteractivityData.version;
169	}	169	}
170	else if (ischosen(p_xmlFragment.PreviewData)) {	170	else if (ischosen(p_xmlFragment.PreviewData)) {
171	v_version := p_xmlFragment.PreviewData.version;	171	v_version := p_xmlFragment.PreviewData.version;
172	}	172	}
173		173	
174	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(174	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(
175	vc_BCASTTransportID,	175	vc_BCASTTransportID,
176	v_version,	176	v_version,
177	p_xmlFragment)	177	p_xmlFragment)
178);	178);
179		179	
180	vc_BCASTTransportID := vc_BCASTTransportID + 1;	180	vc_BCASTTransportID := vc_BCASTTransportID + 1;
181	}	181	}
182		182	
183	// CR 26/8: generic function for adding XML fragments	183	// CR 26/8: generic function for adding XML fragments
184	/**	184	/**
185	*	185	*
186	* @desc	186	* @desc
187	* This function adds a service fragment at the last element of	187	* This function adds a service fragment at the last element of
188	* a SGDU.	188	* a SGDU.
189	* @param	189	* @param
190	* p_SGDU The SGDU which shoud add the service fragment.	190	* p_SGDU The SGDU which shoud add the service fragment.
191	* @param	191	* @param
192	* p_Service The service fragment to add	192	* p_Service The service fragment to add
193	*/	193	*/
194	function f_addServiceFragment(194	function f_addServiceFragment(
195	inout ServiceGuideDeliveryUnit p_SGDU,	195	inout ServiceGuideDeliveryUnit p_SGDU,
196	ServiceType p_Service) runs on BCastMainComponent {	196	ServiceType p_Service) runs on BCastMainComponent {
197	var integer v_size := sizeof (p_SGDU);	197	var integer v_size := sizeof (p_SGDU);
198	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Service	198	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Service
199	vc_BCASTTransportID := vc_BCASTTransportID + 1;	199	vc_BCASTTransportID := vc_BCASTTransportID + 1;
200	}	200	}
201		201	
202	// CR 26/8: generic function for adding XML fragments	202	// CR 26/8: generic function for adding XML fragments
203	/**	203	/**
204	*	204	*
205	* @desc	205	* @desc
206	* This function adds a content fragment at the last element of	206	* This function adds a content fragment at the last element of
207	* a SGDU.	207	* a SGDU.
208	* @param	208	* @param

Datei: AtsBCast_ServiceGuide_Functions.ttcn (Fortsetzung)

209	//	* p_SGDU The SGDU which shoud add the content fragment.	209	//	* p_SGDU The SGDU which shoud add the content fragment.
210	//	* @param	210	//	* @param
211	//	* p_Content The content fragment to add	211	//	* p_Content The content fragment to add
212	//	*/	212	//	*/
213	//	function f_addContentFragment(213	//	function f_addContentFragment(
214	//	inout ServiceGuideDeliveryUnit p_SGDU,	214	//	inout ServiceGuideDeliveryUnit p_SGDU,
215	//	ContentType p_Content) runs on BCastMainComponent {	215	//	ContentType p_Content) runs on BCastMainComponent {
216	//	var integer v_size := sizeof (p_SGDU);	216	//	var integer v_size := sizeof (p_SGDU);
217	//	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Content	217	//	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Content
218	//	Content := p_Content	218	//	Content := p_Content
219	//	}});	219	//	}});
220	//	vc_BCASTTransportID := vc_BCASTTransportID + 1;	220	//	vc_BCASTTransportID := vc_BCASTTransportID + 1;
221	//	}	221	//	}
222			222		
223		// CR 26/8: generic function for adding XML fragments	223		// CR 26/8: generic function for adding XML fragments
224	//	/**	224	//	/**
225	//	*	225	//	*
226	//	* @desc	226	//	* @desc
227	//	* This function adds a schedule fragment at the last element of	227	//	* This function adds a schedule fragment at the last element of
228	//	* a SGDU.	228	//	* a SGDU.
229	//	* @param	229	//	* @param
230	//	* p_SGDU The SGDU which shoud add the schedule fragment.	230	//	* p_SGDU The SGDU which shoud add the schedule fragment.
231	//	* @param	231	//	* @param
232	//	* p_Schedule The schedule fragment to add	232	//	* p_Schedule The schedule fragment to add
233	//	*/	233	//	*/
234	//	function f_addScheduleFragment(234	//	function f_addScheduleFragment(
235	//	inout ServiceGuideDeliveryUnit p_SGDU,	235	//	inout ServiceGuideDeliveryUnit p_SGDU,
236	//	ScheduleType p_Schedule) runs on BCastMainComponent {	236	//	ScheduleType p_Schedule) runs on BCastMainComponent {
237	//	var integer v_size := sizeof (p_SGDU);	237	//	var integer v_size := sizeof (p_SGDU);
238	//	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Schedu	238	//	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Schedu
239	//	Schedule := p_Schedule	239	//	Schedule := p_Schedule
240	//	}});	240	//	}});
241	//	vc_BCASTTransportID := vc_BCASTTransportID + 1;	241	//	vc_BCASTTransportID := vc_BCASTTransportID + 1;
242	//	}	242	//	}
243			243		
244		// CR 26/8: generic function for adding XML fragments	244		// CR 26/8: generic function for adding XML fragments
245	//	/**	245	//	/**
246	//	*	246	//	*
247	//	* @desc	247	//	* @desc
248	//	* This function adds a access fragment at the last element of a	248	//	* This function adds a access fragment at the last element of a
249	//	* SGDU.	249	//	* SGDU.
250	//	* @param	250	//	* @param
251	//	* p_SGDU The SGDU which shoud add the access fragment.	251	//	* p_SGDU The SGDU which shoud add the access fragment.
252	//	* @param	252	//	* @param
253	//	* p_Access The access fragment to add	253	//	* p_Access The access fragment to add
254	//	*/	254	//	*/
255	//	function f_addAccessFragment(255	//	function f_addAccessFragment(
256	//	inout ServiceGuideDeliveryUnit p_SGDU,	256	//	inout ServiceGuideDeliveryUnit p_SGDU,
257	//	AccessType p_Access) runs on BCastMainComponent {	257	//	AccessType p_Access) runs on BCastMainComponent {
258	//	var integer v_size := sizeof (p_SGDU);	258	//	var integer v_size := sizeof (p_SGDU);
259	//	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Access	259	//	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Access
260	//	Access := p_Access	260	//	Access := p_Access
261	//	}});	261	//	}});
262	//	vc_BCASTTransportID := vc_BCASTTransportID + 1;	262	//	vc_BCASTTransportID := vc_BCASTTransportID + 1;
263	//	}	263	//	}
264			264		
265		// CR 26/8: generic function for adding XML fragments	265		// CR 26/8: generic function for adding XML fragments
266	//	/**	266	//	/**
267	//	*	267	//	*
268	//	* @desc	268	//	* @desc
269	//	* This function adds a PreviewData fragment at the last element of a	269	//	* This function adds a PreviewData fragment at the last element of a
270	//	* SGDU.	270	//	* SGDU.
271	//	* @param	271	//	* @param
272	//	* p_SGDU The SGDU which shoud add the PreviewData fragment.	272	//	* p_SGDU The SGDU which shoud add the PreviewData fragment.
273	//	* @param	273	//	* @param
274	//	* p_PreviewData The PreviewData fragment to add	274	//	* p_PreviewData The PreviewData fragment to add
275	//	*/	275	//	*/
276	//	function f_addPreviewDataFragment(276	//	function f_addPreviewDataFragment(
277	//	inout ServiceGuideDeliveryUnit p_SGDU,	277	//	inout ServiceGuideDeliveryUnit p_SGDU,
278	//	PreviewDataType p_PreviewData) runs on BCastMainComponent {	278	//	PreviewDataType p_PreviewData) runs on BCastMainComponent {
279	//	var integer v_size := sizeof (p_SGDU);	279	//	var integer v_size := sizeof (p_SGDU);

Datei: AtsBCast_ServiceGuide_Functions.ttcn (Fortsetzung)

280 //	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Preview	280 //	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Preview
281 //	PreviewData := p_PreviewData	281 //	PreviewData := p_PreviewData
282 //	}});	282 //	}});
283 //	vc_BCASTTransportID := vc_BCASTTransportID + 1;	283 //	vc_BCASTTransportID := vc_BCASTTransportID + 1;
284 //	}	284 //	}
285		285	
286	// CR 26/8: generic function for adding XML fragments	286	// CR 26/8: generic function for adding XML fragments
287 //	/**	287 //	/**
288 //	*	288 //	*
289 //	* @desc	289 //	* @desc
290 //	* This function adds a InteractivityData fragment at the last element of a	290 //	* This function adds a InteractivityData fragment at the last element of a
291 //	* SGDU.	291 //	* SGDU.
292 //	* @param	292 //	* @param
293 //	* p_SGDU The SGDU which shoud add the InteractivityData fragment.	293 //	* p_SGDU The SGDU which shoud add the InteractivityData fragment.
294 //	* @param	294 //	* @param
295 //	* p_InteractivityData The InteractivityData fragment to add	295 //	* p_InteractivityData The InteractivityData fragment to add
296 //	*/	296 //	*/
297 //	function f_addInteractivityDataFragment(297 //	function f_addInteractivityDataFragment(
298 //	inout ServiceGuideDeliveryUnit p_SGDU,	298 //	inout ServiceGuideDeliveryUnit p_SGDU,
299 //	InteractivityDataType p_InteractivityData) runs on BCastMainComponent {	299 //	InteractivityDataType p_InteractivityData) runs on BCastMainComponent {
300 //	var integer v_size := sizeof (p_SGDU);	300 //	var integer v_size := sizeof (p_SGDU);
301 //	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Interac	301 //	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Interac
302 //	InteractivityData := p_InteractivityData	302 //	InteractivityData := p_InteractivityData
303 //	}});	303 //	}});
304 //	vc_BCASTTransportID := vc_BCASTTransportID + 1;	304 //	vc_BCASTTransportID := vc_BCASTTransportID + 1;
305 //	}	305 //	}
306		306	
307	// CR 26/8: generic function for adding XML fragments	307	// CR 26/8: generic function for adding XML fragments
308 //	/**	308 //	/**
309 //	*	309 //	*
310 //	* @desc	310 //	* @desc
311 //	* This function adds a PurchaseItem fragment at the last element of a	311 //	* This function adds a PurchaseItem fragment at the last element of a
312 //	* SGDU.	312 //	* SGDU.
313 //	* @param	313 //	* @param
314 //	* p_SGDU The SGDU which shoud add the PurchaseItem fragment.	314 //	* p_SGDU The SGDU which shoud add the PurchaseItem fragment.
315 //	* @param	315 //	* @param
316 //	* p_PurchaseItem The PurchaseItem fragment to add	316 //	* p_PurchaseItem The PurchaseItem fragment to add
317 //	*/	317 //	*/
318 //	function f_addPurchaseItemFragment(318 //	function f_addPurchaseItemFragment(
319 //	inout ServiceGuideDeliveryUnit p_SGDU,	319 //	inout ServiceGuideDeliveryUnit p_SGDU,
320 //	PurchaseItemType p_PurchaseItem) runs on BCastMainComponent {	320 //	PurchaseItemType p_PurchaseItem) runs on BCastMainComponent {
321 //	var integer v_size := sizeof (p_SGDU);	321 //	var integer v_size := sizeof (p_SGDU);
322 //	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Purchas	322 //	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Purchas
323 //	PurchaseItem := p_PurchaseItem	323 //	PurchaseItem := p_PurchaseItem
324 //	}});	324 //	}});
325 //	vc_BCASTTransportID := vc_BCASTTransportID + 1;	325 //	vc_BCASTTransportID := vc_BCASTTransportID + 1;
326 //	}	326 //	}
327		327	
328	// CR 26/8: generic function for adding XML fragments	328	// CR 26/8: generic function for adding XML fragments
329 //	/**	329 //	/**
330 //	*	330 //	*
331 //	* @desc	331 //	* @desc
332 //	* This function adds a PurchaseData fragment at the last element of a	332 //	* This function adds a PurchaseData fragment at the last element of a
333 //	* SGDU.	333 //	* SGDU.
334 //	* @param	334 //	* @param
335 //	* p_SGDU The SGDU which shoud add the PurchaseData fragment.	335 //	* p_SGDU The SGDU which shoud add the PurchaseData fragment.
336 //	* @param	336 //	* @param
337 //	* p_PurchaseData The PurchaseData fragment to add	337 //	* p_PurchaseData The PurchaseData fragment to add
338 //	*/	338 //	*/
339 //	function f_addPurchaseDataFragment(339 //	function f_addPurchaseDataFragment(
340 //	inout ServiceGuideDeliveryUnit p_SGDU,	340 //	inout ServiceGuideDeliveryUnit p_SGDU,
341 //	PurchaseDataType p_PurchaseData) runs on BCastMainComponent {	341 //	PurchaseDataType p_PurchaseData) runs on BCastMainComponent {
342 //	var integer v_size := sizeof (p_SGDU);	342 //	var integer v_size := sizeof (p_SGDU);
343 //	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Purchas	343 //	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_Purchas
344 //	PurchaseData := p_PurchaseData	344 //	PurchaseData := p_PurchaseData
345 //	}});	345 //	}});
346 //	vc_BCASTTransportID := vc_BCASTTransportID + 1;	346 //	vc_BCASTTransportID := vc_BCASTTransportID + 1;
347 //	}	347 //	}
348		348	
349	// CR 26/8: generic function for adding XML fragments	349	// CR 26/8: generic function for adding XML fragments
350 //	/**	350 //	/**

Datei: AtsBCast_ServiceGuide_Functions.ttcn (Fortsetzung)

351	//	*	351	//	*
352	//	* @desc	352	//	* @desc
353	//	* This function adds a PurchaseChannel fragment at the last element of a	353	//	* This function adds a PurchaseChannel fragment at the last element of a
354	//	* SGDU.	354	//	* SGDU.
355	//	* @param	355	//	* @param
356	//	* p_SGDU The SGDU which shoud add the PurchaseChannel fragment.	356	//	* p_SGDU The SGDU which shoud add the PurchaseChannel fragment.
357	//	* @param	357	//	* @param
358	//	* p_PurchaseChannel The PurchaseChannel fragment to add	358	//	* p_PurchaseChannel The PurchaseChannel fragment to add
359	//	*/	359	//	*/
360	//	function f_addPurchaseChannelFragment(360	//	function f_addPurchaseChannelFragment(
361	//	inout ServiceGuideDeliveryUnit p_SGDU,	361	//	inout ServiceGuideDeliveryUnit p_SGDU,
362	//	PurchaseChannelType p_PurchaseChannel) runs on BCastMainComponent {	362	//	PurchaseChannelType p_PurchaseChannel) runs on BCastMainComponent {
363	//	var integer v_size := sizeof (p_SGDU);	363	//	var integer v_size := sizeof (p_SGDU);
364	//	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_PurchaseChannel := p_PurchaseChannel	364	//	p_SGDU[v_size] := valueof(m_xml_ServiceGuideFragment(vc_BCASTTransportID, p_PurchaseChannel := p_PurchaseChannel
365	//	PurchaseChannel := p_PurchaseChannel	365	//	PurchaseChannel := p_PurchaseChannel
366	//));	366	//));
367	//	vc_BCASTTransportID := vc_BCASTTransportID + 1;	367	//	vc_BCASTTransportID := vc_BCASTTransportID + 1;
368	//	}	368	//	}
369			369		
370		/**	370		/**
371		*	371		*
372		* @desc	372		* @desc
373		* This function adds a sdp fragment at the last element of a	373		* This function adds a sdp fragment at the last element of a
374		* SGDU.	374		* SGDU.
375		* @param	375		* @param
376		* p_SGDU The SGDU which shoud add the sdp fragment.	376		* p_SGDU The SGDU which shoud add the sdp fragment.
377		* @param	377		* @param
378		* p_version The fragment version	378		* p_version The fragment version
379		* @param	379		* @param
380		* p_SDP The sdp fragment to add	380		* p_SDP The sdp fragment to add
381		*/	381		*/
382		function f_addSdpFragment(382		function f_addSdpFragment(
383		inout ServiceGuideDeliveryUnit p_SGDU,	383		inout ServiceGuideDeliveryUnit p_SGDU,
384		UInt32 p_version,	384		UInt32 p_version,
385		SdpFragment p_SDP) runs on BCastMainComponent {	385		SdpFragment p_SDP) runs on BCastMainComponent {
386		var integer v_size := sizeof (p_SGDU);	386		var integer v_size := sizeof (p_SGDU);
387		p_SGDU[v_size] :=	387		p_SGDU[v_size] :=
388		valueof(m_sdp_ServiceGuideFragment(vc_BCASTTransportID, p_version, p_SDP));	388		valueof(m_sdp_ServiceGuideFragment(vc_BCASTTransportID, p_version, p_SDP));
389		vc_BCASTTransportID := vc_BCASTTransportID + 1;	389		vc_BCASTTransportID := vc_BCASTTransportID + 1;
390		}	390		}
391		}	391		}
392			392		
393		group SGDD {	393		group SGDD {
394		/**	394		/**
395		*	395		*
396		* @desc	396		* @desc
397		* This function creates a SGDD fragment from a	397		* This function creates a SGDD fragment from a
398		* ServiceGuideFragment.	398		* ServiceGuideFragment.
399		* @param	399		* @param
400		* p_ServiceGuideFragment The ServiceGuideFragment which	400		* p_ServiceGuideFragment The ServiceGuideFragment which
401		* includes the informations about the Fragment	401		* includes the informations about the Fragment
402		* @param	402		* @param
403		* p_SGDU_Id The SGDU id	403		* p_SGDU_Id The SGDU id
404		* @return	404		* @return
405		* The SGDD Fragment	405		* The SGDD Fragment
406		*/	406		*/
407		function f_createSGDDFragment(407		function f_createSGDDFragment(
408		ServiceGuideFragment p_ServiceGuideFragment,	408		ServiceGuideFragment p_ServiceGuideFragment,
409		charstring p_SGDU_Id) return FragmentType {	409		charstring p_SGDU_Id) return FragmentType {
410		var ServiceGuideFragmentType v_Type :=	410		var ServiceGuideFragmentType v_Type :=
411		p_ServiceGuideFragment.fragmentType;	411		p_ServiceGuideFragment.fragmentType;
412		var FragmentType v_Fragment;	412		var FragmentType v_Fragment;
413			413		
414		if (ischosen (v_Type.xmlFragment)) {	414		if (ischosen (v_Type.xmlFragment)) {
415		var XmlFragment v_xml := v_Type.xmlFragment;	415		var XmlFragment v_xml := v_Type.xmlFragment;
416		var UInt8 v_FragmentType := f_getXmlFragmentType(v_xml);	416		var UInt8 v_FragmentType := f_getXmlFragmentType(v_xml);
417			417		
418		v_Fragment :=	418		v_Fragment :=
419		valueof(m_def_Fragment(419		valueof(m_def_Fragment(
420		p_ServiceGuideFragment.fragmentTransportId,	420		p_ServiceGuideFragment.fragmentTransportId,
421		p_ServiceGuideFragment.fragmentVersion,	421		p_ServiceGuideFragment.fragmentVersion,

Datei: AtsBCast_ServiceGuide_Functions.ttcn (Fortsetzung)

```
422         f_getXmlFragmentId(v_xml),
423         c_XML_BCAST_SG_fragment,
424         v_FragmentType));
425
426     } else if (ischosen (v_Type.sdpFragment)) {
427         v_Fragment :=
428             valueof(m_def_Fragment(
429                 p_ServiceGuideFragment.fragmentTransportId,
430                 p_ServiceGuideFragment.fragmentVersion,
431                 v_Type.sdpFragment.fragmentId,
432                 c_SDP_fragment,
433                 omit));
434     } else if (ischosen (v_Type.mbmsUsbdFragment)) {
435         v_Fragment :=
436             valueof(m_def_Fragment(
437                 p_ServiceGuideFragment.fragmentTransportId,
438                 p_ServiceGuideFragment.fragmentVersion,
439                 v_Type.mbmsUsbdFragment.fragmentId,
440                 c_MBMS_USD_fragment,
441                 omit));
442     } else if (ischosen (v_Type.adpFragment)) {
443         v_Fragment :=
444             valueof(m_def_Fragment(
445                 p_ServiceGuideFragment.fragmentTransportId,
446                 p_ServiceGuideFragment.fragmentVersion,
447                 v_Type.adpFragment.fragmentId,
448                 c_XML_ADP_fragment,
449                 omit));
450     }
451
452     return v_Fragment;
453 }
454
455 /**
456  *
457  * @desc
458  *   this functions returns the id of a xml fragment
459  * @param
460  *   p_XmlFragment The xml fragment
461  * @return
462  *   The id of the xml fragment
463  */
464 function f_getXmlFragmentId(XmlFragment p_XmlFragment) return charstring {
465     var charstring v_id;
466
467     if (ischosen (p_XmlFragment.Service)) {
468         v_id := p_XmlFragment.Service.id;
469     } else if (ischosen (p_XmlFragment.Content)) {
470         v_id := p_XmlFragment.Content.id;
471     } else if (ischosen (p_XmlFragment.Schedule)) {
472         v_id := p_XmlFragment.Schedule.id;
473     } else if (ischosen (p_XmlFragment.Access)) {
474         v_id := p_XmlFragment.Access.id;
475     } else if (ischosen (p_XmlFragment.PreviewData)) {
476         v_id := p_XmlFragment.PreviewData.id;
477     } else if (ischosen (p_XmlFragment.PurchaseItem)) {
478         v_id := p_XmlFragment.PurchaseItem.id;
479     } else if (ischosen (p_XmlFragment.PurchaseData)) {
480         v_id := p_XmlFragment.PurchaseData.id;
481     } else if (ischosen (p_XmlFragment.PurchaseChannel)) {
482         v_id := p_XmlFragment.PurchaseChannel.id;
483     } else if (ischosen (p_XmlFragment.InteractivityData)) {
484         v_id := p_XmlFragment.InteractivityData.id;
485     }
486
487     return v_id;
488 }
489
490 /**
491  *
492  * @desc
```

```
422         f_getXmlFragmentId(v_xml),
423         c_XML_BCAST_SG_fragment,
424         v_FragmentType));
425
426     } else if (ischosen (v_Type.sdpFragment)) {
427         v_Fragment :=
428             valueof(m_def_Fragment(
429                 p_ServiceGuideFragment.fragmentTransportId,
430                 p_ServiceGuideFragment.fragmentVersion,
431                 v_Type.sdpFragment.fragmentId,
432                 c_SDP_fragment,
433                 omit));
434     } else if (ischosen (v_Type.mbmsUsbdFragment)) {
435         v_Fragment :=
436             valueof(m_def_Fragment(
437                 p_ServiceGuideFragment.fragmentTransportId,
438                 p_ServiceGuideFragment.fragmentVersion,
439                 v_Type.mbmsUsbdFragment.fragmentId,
440                 c_MBMS_USD_fragment,
441                 omit));
442     } else if (ischosen (v_Type.adpFragment)) {
443         v_Fragment :=
444             valueof(m_def_Fragment(
445                 p_ServiceGuideFragment.fragmentTransportId,
446                 p_ServiceGuideFragment.fragmentVersion,
447                 v_Type.adpFragment.fragmentId,
448                 c_XML_ADP_fragment,
449                 omit));
450     }
451
452     return v_Fragment;
453 }
454
455 /**
456  *
457  * @desc
458  *   this functions returns the id of a xml fragment
459  * @param
460  *   p_XmlFragment The xml fragment
461  * @return
462  *   The id of the xml fragment
463  */
464 function f_getXmlFragmentId(XmlFragment p_XmlFragment) return charstring {
465     var charstring v_id;
466
467     if (ischosen (p_XmlFragment.Service)) {
468         v_id := p_XmlFragment.Service.id;
469     } else if (ischosen (p_XmlFragment.Content)) {
470         v_id := p_XmlFragment.Content.id;
471     } else if (ischosen (p_XmlFragment.Schedule)) {
472         v_id := p_XmlFragment.Schedule.id;
473     } else if (ischosen (p_XmlFragment.Access)) {
474         v_id := p_XmlFragment.Access.id;
475     } else if (ischosen (p_XmlFragment.PreviewData)) {
476         v_id := p_XmlFragment.PreviewData.id;
477     } else if (ischosen (p_XmlFragment.PurchaseItem)) {
478         v_id := p_XmlFragment.PurchaseItem.id;
479     } else if (ischosen (p_XmlFragment.PurchaseData)) {
480         v_id := p_XmlFragment.PurchaseData.id;
481     } else if (ischosen (p_XmlFragment.PurchaseChannel)) {
482         v_id := p_XmlFragment.PurchaseChannel.id;
483     } else if (ischosen (p_XmlFragment.InteractivityData)) {
484         v_id := p_XmlFragment.InteractivityData.id;
485     }
486
487     return v_id;
488 }
489
490 /**
491  *
492  * @desc
```

Datei: AtsBCast_ServiceGuide_Functions.ttcn (Fortsetzung)

493	* This function returns the fragment type for a xml fragment.		493	* This function returns the fragment type for a xml fragment.
494	* @param		494	* @param
495	* p_XmlFragment The xml fragment.		495	* p_XmlFragment The xml fragment.
496	* @return		496	* @return
497	* the fragment type of the xml fragment		497	* the fragment type of the xml fragment
498	*/		498	*/
499	function f_getXmlFragmentType(XmlFragment p_XmlFragment) return UInt8 {		499	function f_getXmlFragmentType(XmlFragment p_XmlFragment) return UInt8 {
500	var UInt8 v_FragmentType;		500	var UInt8 v_FragmentType;
501			501	
502	if (ischosen (p_XmlFragment.Service)) {		502	if (ischosen (p_XmlFragment.Service)) {
503	v_FragmentType := c_SG_Service_Fragment;		503	v_FragmentType := c_SG_Service_Fragment;
504	} else if (ischosen (p_XmlFragment.Content)) {		504	} else if (ischosen (p_XmlFragment.Content)) {
505	v_FragmentType := c_SG_Content_Fragment;		505	v_FragmentType := c_SG_Content_Fragment;
506	} else if (ischosen (p_XmlFragment.Schedule)) {		506	} else if (ischosen (p_XmlFragment.Schedule)) {
507	v_FragmentType := c_SG_Schedule_Fragment;		507	v_FragmentType := c_SG_Schedule_Fragment;
508	} else if (ischosen (p_XmlFragment.Access)) {		508	} else if (ischosen (p_XmlFragment.Access)) {
509	v_FragmentType := c_SG_Access_Fragment;		509	v_FragmentType := c_SG_Access_Fragment;
510	} else if (ischosen (p_XmlFragment.PreviewData)) {		510	} else if (ischosen (p_XmlFragment.PreviewData)) {
511	v_FragmentType := c_SG_PreviewData_Fragment;		511	v_FragmentType := c_SG_PreviewData_Fragment;
512	} else if (ischosen (p_XmlFragment.PurchaseItem)) {		512	} else if (ischosen (p_XmlFragment.PurchaseItem)) {
513	v_FragmentType := c_SG_PurchaseItem_Fragment;		513	v_FragmentType := c_SG_PurchaseItem_Fragment;
514	} else if (ischosen (p_XmlFragment.PurchaseData)) {		514	} else if (ischosen (p_XmlFragment.PurchaseData)) {
515	v_FragmentType := c_SG_PurchaseData_Fragment;		515	v_FragmentType := c_SG_PurchaseData_Fragment;
516	} else if (ischosen (p_XmlFragment.PurchaseChannel)) {		516	} else if (ischosen (p_XmlFragment.PurchaseChannel)) {
517	v_FragmentType := c_SG_PurchaseChannel_Fragment;		517	v_FragmentType := c_SG_PurchaseChannel_Fragment;
518	} else if (ischosen (p_XmlFragment.InteractivityData)) {		518	} else if (ischosen (p_XmlFragment.InteractivityData)) {
519	v_FragmentType := c_SG_InteractivityData_Fragment;		519	v_FragmentType := c_SG_InteractivityData_Fragment;
520	}		520	}
521			521	
522	return v_FragmentType;		522	return v_FragmentType;
523	}		523	}
524			524	
525	// change 6 (WK18): dynamic setting of TOI number		525	// change 6 (WK18): dynamic setting of TOI number
		+-	526	// change 1 (WK48/08): TSI Value correction - ESG payout
526	/**	=	527	/**
527	*		528	*
528	* @desc		529	* @desc
529	* This function create a SGDD for a SGDU.		530	* This function create a SGDD for a SGDU.
530	* @param		531	* @param
531	* p_SGDU The SGDU		532	* p_SGDU The SGDU
532	* @param		533	* @param
533	* p_SGDU_ID The id of the SGDU		534	* p_SGDU_ID The id of the SGDU
534	* @param		535	* @param
535	* p_SGDD_ID The id of the SGDD		536	* p_SGDD_ID The id of the SGDD
536	* @param		537	* @param
537	* p_TOI4SGDU The TOI number		538	* p_TOI4SGDU The TOI number
		+-	539	* @param
			540	* p_TSI_SGDU The TSI of the SGDU (for reverencing the appropriate SGDU FLUTE session)
			541	* @param
			542	* p_startTime The start time of the SGDU session
			543	* @param
			544	* p_stopTime The stop time of the SGDU session
538	* @return	=	545	* @return
539	* The SGDD		546	* The SGDD
540	*/		547	*/
541	function f_createSGDD4SGDU(548	function f_createSGDD4SGDU(
542	ServiceGuideDeliveryUnit p_SGDU,		549	ServiceGuideDeliveryUnit p_SGDU,
543	AnyUri p_SGDU_ID,		550	AnyUri p_SGDU_ID,
544	AnyUri p_SGDD_ID,		551	AnyUri p_SGDD_ID,
545	UInt p_TOI4SGDU,		552	UInt p_TOI4SGDU,
		<>	553	UInt16 p_TSI_SGDU,
546	UInt p_startTime,		554	UInt p_startTime,
547	UInt p_endTime) return ServiceGuideDeliveryDescriptorType {		555	UInt p_endTime) return ServiceGuideDeliveryDescriptorType {
548	var FragmentList v_list;	=	556	var FragmentList v_list;
549	var FragmentType v_Fragment;		557	var FragmentType v_Fragment;
550	var integer v_SGDU_idx;		558	var integer v_SGDU_idx;
551			559	
552	// go through the ServiceGuideDeliveryUnit to create the FragmentList		560	// go through the ServiceGuideDeliveryUnit to create the FragmentList
553	for (var integer i := 0; i < sizeof (p_SGDU); i := i + 1) {		561	for (var integer i := 0; i < sizeof (p_SGDU); i := i + 1) {
554	var ServiceGuideFragment v_ServiceGuideFragment := p_SGDU[i];		562	var ServiceGuideFragment v_ServiceGuideFragment := p_SGDU[i];
555	v_Fragment :=		563	v_Fragment :=

Datei: AtsBCast_ServiceGuide_Functions.ttcn (Fortsetzung)

556	f_createSGDDFragment(v_ServiceGuideFragment, p_SGDU_ID);	564	f_createSGDDFragment(v_ServiceGuideFragment, p_SGDU_ID);
557	v_list[i] := v_Fragment;	565	v_list[i] := v_Fragment;
558	}	566	}
559		567	
560	var ServiceGuideDeliveryUnitType v_SGDU_field :=	568	var ServiceGuideDeliveryUnitType v_SGDU_field :=
561	valueof(m_def_ServiceGuideDeliveryUnitField(p_TOI4SGDU,	569	valueof(m_def_ServiceGuideDeliveryUnitField(p_TOI4SGDU,
562	p_SGDU_ID, // location	570	p_SGDU_ID, // location
563	v_list)); // fragment list	571	v_list)); // fragment list
564	var ServiceGuideDeliveryDescriptorType v_SGDD :=	572	var ServiceGuideDeliveryDescriptorType v_SGDD :=
565	valueof(m_def_ServiceGuideDeliveryDescriptor(p_SGDD_ID, v_SGDU_field,p_sta	573	valueof(m_def_ServiceGuideDeliveryDescriptor(p_SGDD_ID, v_SGDU_field, p_TS
566		574	
567	return v_SGDD;	575	return v_SGDD;
568	}	576	}
569	}	577	}
570		578	
571	group misc {	579	group misc {
572	/**	580	/**
573	*	581	*
574	* @desc	582	* @desc
575	* This function increase the fragment version.	583	* This function increase the fragment version.
576	* @param	584	* @param
577	* p_version the fragment version to increase	585	* p_version the fragment version to increase
578	* @return	586	* @return
579	* The increased fragment version	587	* The increased fragment version
580	*/	588	*/
581	function f_IncFragmentVersion(in UInt32 p_version) return UInt32 {	589	function f_IncFragmentVersion(in UInt32 p_version) return UInt32 {
582	var UInt32 v_rc := p_version;	590	var UInt32 v_rc := p_version;
583	if (v_rc < c_uInt32Max and v_rc >= 0) {	591	if (v_rc < c_uInt32Max and v_rc >= 0) {
584	v_rc := v_rc + 1;	592	v_rc := v_rc + 1;
585	} else {	593	} else {
586	log ("It is not possible to increase the Fragment Version");	594	log ("It is not possible to increase the Fragment Version");
587	}	595	}
588		596	
589	return v_rc;	597	return v_rc;
590	}	598	}
591		599	
592	/**	600	/**
593	*	601	*
594	* @desc	602	* @desc
595	* This function returns the access tpye for a specific	603	* This function returns the access tpye for a specific
596	* broadcast network bearer.	604	* broadcast network bearer.
597	* @param	605	* @param
598	* p_type The BroadcastBearerType enumeration which contains the	606	* p_type The BroadcastBearerType enumeration which contains the
599	* access type.	607	* access type.
600	* @return	608	* @return
601	* The access type used in access fragment	609	* The access type used in access fragment
602	*/	610	*/
603	function f_getAccessType(BroadcastBearerType p_type) return UInt8 {	611	function f_getAccessType(BroadcastBearerType p_type) return UInt8 {
604	var UInt8 v_rc;	612	var UInt8 v_rc;
605		613	
606	if (p_type == e_dvb_h) {	614	if (p_type == e_dvb_h) {
607	v_rc := c_ipdcOverDvbH_bcType;	615	v_rc := c_ipdcOverDvbH_bcType;
608	} else if (p_type == e_mbms) {	616	} else if (p_type == e_mbms) {
609	v_rc := c_3gppMbms_bcType;	617	v_rc := c_3gppMbms_bcType;
610	} else if (p_type == e_bmcms) {	618	} else if (p_type == e_bmcms) {
611	v_rc := c_3gpp2Bcmcs_bcType;	619	v_rc := c_3gpp2Bcmcs_bcType;
612	}	620	}
613		621	
614	return v_rc;	622	return v_rc;
615	}	623	}
616		624	
617	/**	625	/**
618	*	626	*
619	* @desc This external function returns the 32 Bits integer part of the current NTP timesta	627	* @desc This external function returns the 32 Bits integer part of the current NTP timesta
620	* @return The current NTP timestamp	628	* @return The current NTP timestamp
621	*/	629	*/
622	external function fx_getNTPTime() return UInt;	630	external function fx_getNTPTime() return UInt;
623		631	
624	/**	632	/**
625	*	633	*
626	* @desc This external function returns the DateTime representation for a NTP timestamp.	634	* @desc This external function returns the DateTime representation for a NTP timestamp.

Datei: AtsBCast_ServiceGuide_Functions.ttcn (Fortsetzung)

627	* @param p_NTPTIME The NTP timestamp which should be converted to the DateTime format.	635	* @param p_NTPTIME The NTP timestamp which should be converted to the DateTime format.
628	* @return The DateTime representation of the NTP timestamp	636	* @return The DateTime representation of the NTP timestamp
629	*/	637	*/
630	external function fx_getDateTime(UInt p_NTPTIME) return DateTime;	638	external function fx_getDateTime(UInt p_NTPTIME) return DateTime;
631		639	
632		640	
633	/**	641	/**
634	*	642	*
635	* @desc	643	* @desc
636	* This function returns the current NTP time as integer.	644	* This function returns the current NTP time as integer.
637	* @return	645	* @return
638	* The current NTP time.	646	* The current NTP time.
639	*/	647	*/
640	function f_getNTPTIME() return UInt {	648	function f_getNTPTIME() return UInt {
641	if(PX_SIMULATE_TC) {	649	if(PX_SIMULATE_TC) {
642	return 0;	650	return 0;
643	}	651	}
644		652	
645	return fx_getNTPTIME();	653	return fx_getNTPTIME();
646	}	654	}
647		655	
648	/**	656	/**
649	*	657	*
650	* @desc	658	* @desc
651	* This function returns the DateTime for a NTP time.	659	* This function returns the DateTime for a NTP time.
652	* @param	660	* @param
653	* p_NTPTIME The NTP time which should be converted	661	* p_NTPTIME The NTP time which should be converted
654	* @return	662	* @return
655	* The DateTime	663	* The DateTime
656	*/	664	*/
657	function f_getDateTime(UInt p_NTPTIME) return DateTime {	665	function f_getDateTime(UInt p_NTPTIME) return DateTime {
658	if(PX_SIMULATE_TC) {	666	if(PX_SIMULATE_TC) {
659	return "Day:Month:Year hh:mm:ss";	667	return "Day:Month:Year hh:mm:ss";
660	}	668	}
661		669	
662	return fx_getDateTime(p_NTPTIME);	670	return fx_getDateTime(p_NTPTIME);
663	}	671	}
664		672	
665	// change 37 (WK18): used for file distribution TC > delivered SDP parameter in the Access	673	// change 37 (WK18): used for file distribution TC > delivered SDP parameter in the Access
666	/**	674	/**
667	* @desc Converts an TTCN-3 string into a base64 encoded TTCN-3 string.	675	* @desc Converts an TTCN-3 string into a base64 encoded TTCN-3 string.
668	* @param p_charstring The charstring which should be base64 encoded.	676	* @param p_charstring The charstring which should be base64 encoded.
669	* @return A base64 representation of the input charstring.	677	* @return A base64 representation of the input charstring.
670	*/	678	*/
671	external function fx_charstring2Base64(in charstring p_charstring) return charstring;	679	external function fx_charstring2Base64(in charstring p_charstring) return charstring;
672	}	680	}
673	}	681	}

Datei: LibBCast_BCastNetworkControl_Functions.ttcn

1	/**	=	1	/**
2	*		2	*
3	* @author		3	* @author
4	* ETSI CTI BCAST CON Project		4	* ETSI CTI BCAST CON Project
5	* @version		5	* @version
6	* \$Id\$		6	* \$Id\$
7	* @desc		7	* @desc
8	* This module specifies some function which can be used by the BCastNetworkContol component.		8	* This module specifies some function which can be used by the BCastNetworkContol component.
9	*/		9	*/
10	module LibBCast_BCastNetworkControl_Functions {		10	module LibBCast_BCastNetworkControl_Functions {
11	import from LibBCast_Common_TypesAndValues all;		11	import from LibBCast_Common_TypesAndValues all;
12	import from LibBCast_ServiceGuide_TypesAndValues all;		12	import from LibBCast_ServiceGuide_TypesAndValues all;
13	import from AtsBCast_ServiceGuide_Functions all;		13	import from AtsBCast_ServiceGuide_Functions all;
14	import from LibBCast_Common_Templates all;		14	import from LibBCast_Common_Templates all;
15	import from LibBCast_ServicePrimitives_TypesAndValues all;		15	import from LibBCast_ServicePrimitives_TypesAndValues all;
16	import from LibBCast_UpperTester_Functions all;		16	import from LibBCast_UpperTester_Functions all;
17	import from LibBCast_Interface all;		17	import from LibBCast_Interface all;
18	import from LibBCast_ModuleParameters all;		18	import from LibBCast_ModuleParameters all;
19	import from LibCommon_Time all;		19	import from LibCommon_Time all;
20	import from LibCommon_BasicTypesAndValues all;		20	import from LibCommon_BasicTypesAndValues all;
21	import from AtsBCast_ModuleParameters all; // change 16 (WK 6): usage of PIXIT param		21	import from AtsBCast_ModuleParameters all; // change 16 (WK 6): usage of PIXIT param
22	import from LibBCast_ServiceGuide_Templates { // change 13 (WK 6): send DVB-H bootstrap		22	import from LibBCast_ServiceGuide_Templates { // change 13 (WK 6): send DVB-H bootstrap
23	group DVBH_Bootstrap;		23	group DVBH_Bootstrap;
24	};		24	};
25			25	
26	group subFunctions {		26	group subFunctions {
27	group behaviorFunctions {		27	group behaviorFunctions {
28			28	
29	/**		29	/**
30	*		30	*
31	* @desc		31	* @desc
32	* This function waits for a http subscription indicaton.		32	* This function waits for a http subscription indicaton.
33	* @param		33	* @param
34	* serviceName the name of the service		34	* serviceName the name of the service
35	* @verdict		35	* @verdict
36	* pass The substription was successfull received for the		36	* pass The substription was successfull received for the
37	* given service.		37	* given service.
38	* @verdict		38	* @verdict
39	* fail A message was received which is not expected.		39	* fail A message was received which is not expected.
40	* @verdict		40	* @verdict
41	* inconc A guard timer expires.		41	* inconc A guard timer expires.
42	*/		42	*/
43	function f_ctrl_awaitingHttpSubscriptionIndication(charstring serviceName) runs on		43	function f_ctrl_awaitingHttpSubscriptionIndication(charstring serviceName) runs on
44	t_wait.start(PX_TWAIT);		44	t_wait.start(PX_TWAIT);
45			45	
46	alt {		46	alt {
47	[] ic.receive (HttpSubscriptionRequestIndication:{?, serviceName})		47	[] ic.receive (HttpSubscriptionRequestIndication:{?, serviceName})
48	t_wait.stop;		48	t_wait.stop;
49	setverdict(pass);		49	setverdict(pass);
50	}		50	}
51	[] ic.receive {		51	[] ic.receive {
52	t_wait.stop;		52	t_wait.stop;
53	setverdict(fail);		53	setverdict(fail);
54	log("***f_ctrl_awaitingHttpSubscriptionIndication*** wrong		54	log("***f_ctrl_awaitingHttpSubscriptionIndication*** wrong
55	}		55	}
56	[] t_wait.timeout {		56	[] t_wait.timeout {
57	setverdict(inconc);		57	setverdict(inconc);
58	log("***f_ctrl_awaitingHttpSubscriptionIndication*** timer		58	log("***f_ctrl_awaitingHttpSubscriptionIndication*** timer
59	}		59	}
60	}		60	}
61	}		61	}
62			62	
63			63	
64	/**		64	/**
65	*		65	*
66	* @desc		66	* @desc
67	* This function waits for a mms indication.		67	* This function waits for a mms indication.
68	* @param		68	* @param
69	* content the content which should be inside of the mms.		69	* content the content which should be inside of the mms.
70	* @verdict		70	* @verdict
71	* pass The indication was successfull received for the		71	* pass The indication was successfull received for the

Datei: LibBCast_BCastNetworkControl_Functions.ttcn
(Fortsetzung)

72	* given service.	72	* given service.
73	* @verdict	73	* @verdict
74	* fail A message was received which is not expected.	74	* fail A message was received which is not expected.
75	* @verdict	75	* @verdict
76	* inconc A guard timer expires.	76	* inconc A guard timer expires.
77	*/	77	*/
78	function f_ctrl_awaitingMMS(charstring content) runs on BCastNetworkControlComponen	78	function f_ctrl_awaitingMMS(charstring content) runs on BCastNetworkControlComponen
79	t_wait.start(PX_TWAIT);	79	t_wait.start(PX_TWAIT);
80		80	
81	alt {	81	alt {
82	[] mms.receive(MmsGetIndication:{content}) {	82	[] mms.receive(MmsGetIndication:{content}) {
83	t_wait.stop;	83	t_wait.stop;
84	setverdict(pass);	84	setverdict(pass);
85	}	85	}
86	[] mms.receive {	86	[] mms.receive {
87	t_wait.stop;	87	t_wait.stop;
88	setverdict(fail);	88	setverdict(fail);
89	log("***f_ctrl_awaitingMMS*** wrong message received");	89	log("***f_ctrl_awaitingMMS*** wrong message received");
90	}	90	}
91	[] t_wait.timeout {	91	[] t_wait.timeout {
92	setverdict(inconc);	92	setverdict(inconc);
93	log("***f_ctrl_awaitingMMS*** time expired");	93	log("***f_ctrl_awaitingMMS*** time expired");
94	}	94	}
95	}	95	}
96	}	96	}
97		97	
98	/**	98	/**
99	*	99	*
100	* @desc	100	* @desc
101	* This function waits for a sms indication.	101	* This function waits for a sms indication.
102	* @param	102	* @param
103	* content the content which should be inside of the mms.	103	* content the content which should be inside of the mms.
104	* @verdict	104	* @verdict
105	* pass The indication was successfull received for the	105	* pass The indication was successfull received for the
106	* given service.	106	* given service.
107	* @verdict	107	* @verdict
108	* fail A message was received which is not expected.	108	* fail A message was received which is not expected.
109	* @verdict	109	* @verdict
110	* inconc A guard timer expires.	110	* inconc A guard timer expires.
111	*/	111	*/
112	function f_ctrl_awaitingSMS(charstring content) runs on BCastNetworkControlComponen	112	function f_ctrl_awaitingSMS(charstring content) runs on BCastNetworkControlComponen
113	t_wait.start(PX_TWAIT);	113	t_wait.start(PX_TWAIT);
114		114	
115	alt {	115	alt {
116	[] sms.receive(SmsGetIndication:{content}) {	116	[] sms.receive(SmsGetIndication:{content}) {
117	t_wait.stop;	117	t_wait.stop;
118	setverdict(pass);	118	setverdict(pass);
119	}	119	}
120	[] sms.receive(SmsGetIndication:{*}) { // change 38 (WK18): removes	120	[] sms.receive(SmsGetIndication:{*}) { // change 38 (WK18): removes
121	t_wait.stop;	121	t_wait.stop;
122	setverdict(fail);	122	setverdict(fail);
123	log("***f_ctrl_awaitingSMS*** wrong message received");	123	log("***f_ctrl_awaitingSMS*** wrong message received");
124	}	124	}
125	[] t_wait.timeout {	125	[] t_wait.timeout {
126	setverdict(inconc);	126	setverdict(inconc);
127	log("***f_ctrl_awaitingSMS*** time expired");	127	log("***f_ctrl_awaitingSMS*** time expired");
128	}	128	}
129	}	129	}
130	}	130	}
131		131	
132	// change 26 (WK18): (additional to 17) clearing of HTTP request/responses for Inte	132	// change 26 (WK18): (additional to 17) clearing of HTTP request/responses for Inte
133	/**	133	/**
134	*	134	*
135	* @desc	135	* @desc
136	* This function waits for a http post request and send the	136	* This function waits for a http post request and send the
137	* related response including the service guide.	137	* related response including the service guide.
138	* @param	138	* @param
139	* p_SGDU The ServiceGuideDeliveryUnit to send	139	* p_SGDU The ServiceGuideDeliveryUnit to send
140	* @param	140	* @param
141	* p_SGDU_Id The ServiceGuideDeliveryUnit id	141	* p_SGDU_Id The ServiceGuideDeliveryUnit id

Datei: LibBCast_BCastNetworkControl_Functions.ttcn
(Fortsetzung)

142	* @param	142	* @param
143	* p_SGDD_Id The SGDD id	143	* p_SGDD_Id The SGDD id
144	* @verdict	144	* @verdict
145	* pass in case of that the service guide could successful	145	* pass in case of that the service guide could successful
146	* send	146	* send
147	* @verdict	147	* @verdict
148	* fail in case of the service guide could not successful	148	* fail in case of the service guide could not successful
149	* send	149	* send
150	* @verdict	150	* @verdict
151	* inconc a guard timer runs out of time	151	* inconc a guard timer runs out of time
152	*/	152	*/
153	function f_ctrl_ServiceGuideViaHttp(153	function f_ctrl_ServiceGuideViaHttp(
154	ServiceGuideDeliveryUnit p_SGDU,	154	ServiceGuideDeliveryUnit p_SGDU,
155	charstring p_SGDU_Id,	155	charstring p_SGDU_Id,
156	charstring p_SGDD_Id) runs on BCastNetworkControlComponent {	156	charstring p_SGDD_Id) runs on BCastNetworkControlComponent {
157	var HttpPostIndication v_PostRequest;	157	var HttpPostIndication v_PostRequest;
158		158	
159	// awaiting HTTP POST request	159	// awaiting HTTP POST request
160	t_wait.start(PX_TWAIT);	160	t_wait.start(PX_TWAIT);
161	alt {	161	alt {
162	[] ic.receive(mw_HttpPostIndication_unspecific_sgdd_sgdu) -> value	162	[] ic.receive(mw_HttpPostIndication_unspecific_sgdd_sgdu) -> value
163	t_wait.stop;	163	t_wait.stop;
164	setverdict(pass);	164	setverdict(pass);
165	log("Interaction channel: request for SGDD and SGDU detected	165	log("Interaction channel: request for SGDD and SGDU detected
166	f_sendServiceGuideHttpResponse(166	f_sendServiceGuideHttpResponse(
167	p_SGDU,	167	p_SGDU,
168	PX_SGDU_ID,	168	PX_SGDU_ID,
169	PX_SGDD_ID,	169	PX_SGDD_ID,
170	e_sgdd_sgdu	170	e_sgdd_sgdu
171);	171);
172	}	172	}
173	[] ic.receive(mw_HttpPostIndication_empty) -> value v_PostRequest {	173	[] ic.receive(mw_HttpPostIndication_empty) -> value v_PostRequest {
174	t_wait.stop;	174	t_wait.stop;
175	setverdict(pass);	175	setverdict(pass);
176	log("Interaction channel: unspecific request.");	176	log("Interaction channel: unspecific request.");
177	f_sendServiceGuideHttpResponse(177	f_sendServiceGuideHttpResponse(
178	p_SGDU,	178	p_SGDU,
179	PX_SGDU_ID,	179	PX_SGDU_ID,
180	PX_SGDD_ID,	180	PX_SGDD_ID,
181	e_sgdd_sgdu	181	e_sgdd_sgdu
182);	182);
183	}	183	}
184	[] ic.receive(mw_HttpPostIndication_unspecific_sgdd) -> value v_Pos	184	[] ic.receive(mw_HttpPostIndication_unspecific_sgdd) -> value v_Pos
185	t_wait.stop;	185	t_wait.stop;
186	setverdict(pass);	186	setverdict(pass);
187	log("Interaction channel: unspecific request for SGDD.");	187	log("Interaction channel: unspecific request for SGDD.");
188	f_sendServiceGuideHttpResponse(188	f_sendServiceGuideHttpResponse(
189	p_SGDU,	189	p_SGDU,
190	PX_SGDU_ID,	190	PX_SGDU_ID,
191	PX_SGDD_ID,	191	PX_SGDD_ID,
192	e_sgdd	192	e_sgdd
193);	193);
194	alt {	194	alt {
195	[] ic.receive(mw_HttpPostIndication_unspecific_sgdu	195	[] ic.receive(mw_HttpPostIndication_unspecific_sgdu
196	t_wait.stop;	196	t_wait.stop;
197	setverdict(pass);	197	setverdict(pass);
198	log("Interaction channel: unspecific request	198	log("Interaction channel: unspecific request
199	f_sendServiceGuideHttpResponse(199	f_sendServiceGuideHttpResponse(
200	p_SGDU,	200	p_SGDU,
201	PX_SGDU_ID,	201	PX_SGDU_ID,
202	PX_SGDD_ID,	202	PX_SGDD_ID,
203	e_sgdu	203	e_sgdu
204);	204);
205	}	205	}
206	[] t_wait.timeout {	206	[] t_wait.timeout {
207	setverdict (inconc);	207	setverdict (inconc);
208	}	208	}
209	}	209	}
210	}	210	}
211	[] ic.receive(mw_HttpPostIndication_any) -> value v_PostRequest {	211	[] ic.receive(mw_HttpPostIndication_any) -> value v_PostRequest {

Datei: LibBCast_BCastNetworkControl_Functions.ttcn
(Fortsetzung)

212	t_wait.stop;		212	t_wait.stop;
213	setverdict(fail);		213	setverdict(fail);
214	log("Unexpected HTTP POST request!");		214	log("Unexpected HTTP POST request!");
215	}		215	}
216	[] t_wait.timeout {		216	[] t_wait.timeout {
217	setverdict (inconc);		217	setverdict (inconc);
218	}		218	}
219	}		219	}
220	}		220	}
221			221	
222			222	
223	// CR (WK 34/2008): Flute Update Signalling		223	// CR (WK 34/2008): Flute Update Signalling
224	/**		224	/**
225	*		225	*
226	* @desc		226	* @desc
227	* This function broadcast a ervice guide over the flute		227	* This function broadcast a ervice guide over the flute
228	* protocol		228	* protocol
229	* @param		229	* @param
230	* p_SGDU The ServiceGuideDeliveryUnit		230	* p_SGDU The ServiceGuideDeliveryUnit
231	* @param		231	* @param
232	* p_SGDU_Id The ServiceGuideDeliveryUnit id		232	* p_SGDU_Id The ServiceGuideDeliveryUnit id
233	* @param		233	* @param
234	* p_SGDD_Id The id of the ServiceGuideDeliveryDescriptor		234	* p_SGDD_Id The id of the ServiceGuideDeliveryDescriptor
235	* @param		235	* @param
236	* p_Encoding The encoding type for compression in case that		236	* p_Encoding The encoding type for compression in case that
237	* the service guide should be compressed otherwise this		237	* the service guide should be compressed otherwise this
238	* parameter should be omitted.		238	* parameter should be omitted.
239	* @param		239	* @param
240	* p_FluteUpdateID The Flute Instance ID of the Flute Session.		240	* p_FluteUpdateID The Flute Instance ID of the Flute Session.
241	* @verdict		241	* @verdict
242	* pass in case the broadcast of the		242	* pass in case the broadcast of the
243	* ServiceGuideDeliveryUnit was successful		243	* ServiceGuideDeliveryUnit was successful
244	* @verdict		244	* @verdict
245	* fail in case that the broadcast was not successful		245	* fail in case that the broadcast was not successful
246	* @verdict		246	* @verdict
247	* inconc in case of the timer runs out of time		247	* inconc in case of the timer runs out of time
248	*/		248	*/
249	function f_ctrl_broadcastServiceGuide(249	function f_ctrl_broadcastServiceGuide(
250	ServiceGuideDeliveryUnit p_SGDU,		250	ServiceGuideDeliveryUnit p_SGDU,
251	charstring p_SGDU_Id,		251	charstring p_SGDU_Id,
252	charstring p_SGDD_Id,		252	charstring p_SGDD_Id,
253	template charstring p_Encoding,		253	template charstring p_Encoding,
254	UInt32 p_FluteUpdateID) runs on BCastNetworkControlComponent {		254	UInt32 p_FluteUpdateID) runs on BCastNetworkControlComponent {
		++	255	
			256	
			257	// change 1 (WK48/08): TSI Value correction - ESG payout
			258	/* values to define the respecitve TSI values of the FLUTE sessions.
			259	* DVB-H bootstrap contains reference to the SGDD and therwith also the TS
			260	* while the SGDD refereneces to the SGDU and consequently contains the TSI
			261	*/
			262	var UInt16 v_TSI_DVBH := 0;
			263	var UInt16 v_TSI_SGDD := 1;
			264	var UInt16 v_TSI_SGDU := 2;
			265	
255	// create SGDD	=	266	// create SGDD
256	var ServiceGuideDeliveryDescriptorType v_SGDD :=		267	var ServiceGuideDeliveryDescriptorType v_SGDD :=
257	f_createSGDD4SGDU(p_SGDU, p_SGDU_Id, p_SGDD_Id, v_TOI4SGDU, v_start	<>	268	f_createSGDD4SGDU(p_SGDU, p_SGDU_Id, p_SGDD_Id, v_TOI4SGDU, v_TSI_
258		=	269	
259	if (PX_BROADCAST_NETWORK_BEARER == e_dvb_h) {		270	if (PX_BROADCAST_NETWORK_BEARER == e_dvb_h) {
260	// DVB-H ESG Bootstrapping		271	// DVB-H ESG Bootstrapping
261	f_broadcast_dvbh_bootstrap(0, p_FluteUpdateID);	<>	272	f_broadcast_dvbh_bootstrap(v_TSI_DVBH, v_TSI_SGDD, p_FluteUpdateID)
262	}	=	273	}
263			274	
264	// broadcast SGDD		275	// broadcast SGDD
265	f_broadcast_sgdd(v_SGDD, 1, omit, p_FluteUpdateID);	<>	276	f_broadcast_sgdd(v_SGDD, v_TSI_SGDD, omit, p_FluteUpdateID);
266		=	277	
267	// broadcast ServiceGuideDeliveryUnit		278	// broadcast ServiceGuideDeliveryUnit
268	f_broadcast_sgdu(p_SGDU, p_SGDU_Id, 2, p_Encoding, p_FluteUpdateID);	<>	279	f_broadcast_sgdu(p_SGDU, p_SGDU_Id, v_TSI_SGDU, p_Encoding, p_FluteUpdateID)
269	}	=	280	}
270			281	
271				
272				
273				
274				
275				
276				
277				
278				
279				
280				
281				
282				
283				
284				
285				
286				
287				
288				
289				
290				
291				
292				
293				
294				
295				
296				
297				
298				
299				
300				
301				
302				
303				
304				
305				
306				
307				
308				
309				
310				
311				
312				
313				
314				
315				
316				
317				
318				
319				
320				
321				
322				
323				
324				
325				
326				
327				
328				
329				
330				
331				
332				
333				
334				
335				
336				
337				
338				
339				
340				
341				
342				
343				
344				
345				
346				
347				
348				
349				
350				
351				
352				
353				
354				
355				
356				
357				
358				
359				
360				
361				
362				
363				
364				
365				
366				
367				
368				
369				
370				
371				
372				
373				
374				
375				
376				
377				
378				
379				
380				
381				
382				
383				
384				
385				
386				
387				
388				
389				
390				
391				
392				
393				
394				
395				
396				
397				
398				
399				
400				
401				
402				
403				
404				
405				
406				
407				
408				
409				
410				
411				
412				
413				
414				
415				
416				
417				
418				
419				
420				
421				
422				
423				
424				
425				
426				
427				
428				
429				
430				
431				
432				
433				
434				
435				
436				
437				
438				
439				
440				
441				
442				
443				
444				
445				
446				
447				
448				
449				
450				
451				
452				
453				
454				
455				
456				
457				
458				
459				
460				
461				
462				
463				
464				
465				
466				
467				
468				
469				
470				
471				
472				
473				
474				
475				
476				
477				
478				
479				
480				
481				
482				
483				
484				
485				
486				
487				
488				
489				
490				
491				
492				
493				
494				
495				
496				
497				
498				
499				
500				
501				
502				
503				
504				
505				
506				
507				
508				
509				
510				
511				
512				
513				
514				
515				
516				
517				
518				
519				
520				
521				
522				
523				
524				
525				
526				
527				
528				
529				
530				
531				
532				
533				
534				
535				
536				
537				
538				
539				
540				
541				
542				
543				
544				
545				
546				
547				
548				
549				
550				
551				
552				
553				
554				
555				
556				
557				
558				
559				
560				
561				
562				
563				
564				
565				
566				
567				
568				
569				
570				
571				
572				
573				
574				
575				
576				
577				
578				
579				
580				
581				
582				
583				
584				
585				
586				
587	</			

Datei: LibBCast_BCastNetworkControl_Functions.ttcn
(Fortsetzung)

271	}		282	}
272			283	
273	group commonFunctions {		284	group commonFunctions {
274			285	
275	// change 14 (WK 6): moved function from 'AtsBCast_ServiceGuide_Funtions'		286	// change 14 (WK 6): moved function from 'AtsBCast_ServiceGuide_Funtions'
276	/**		287	/**
277	*		288	*
278	* @desc		289	* @desc
279	* initialize the start and end time of the BCastNetworkControlComponent.		290	* initialize the start and end time of the BCastNetworkControlComponent.
280	* @param		291	* @param
281	* p_startTime The Start Time which is one hour after test execution		292	* p_startTime The Start Time which is one hour after test execution
282	* @param		293	* @param
283	* p_endTime The End Time which is set to a value one our after the start time		294	* p_endTime The End Time which is set to a value one our after the start time
284	*/		295	*/
285	function f_ctrl_InitStartEndTime(UInt p_startTime, UInt p_endTime) runs on BCastNetworkCont		296	function f_ctrl_InitStartEndTime(UInt p_startTime, UInt p_endTime) runs on BCastNetworkCont
286	v_startTime := p_startTime;		297	v_startTime := p_startTime;
287	v_endTime := p_endTime;		298	v_endTime := p_endTime;
288	}		299	}
289			300	
290	// change 26 (WK18): (additional to 17) clearing of HTTP request/responses for Interaction		301	// change 26 (WK18): (additional to 17) clearing of HTTP request/responses for Interaction
291	/**		302	/**
292	*		303	*
293	* @desc		304	* @desc
294	* This function sends a http response including the service		305	* This function sends a http response including the service
295	* guide.		306	* guide.
296	* @param		307	* @param
297	* p_SGDU The ServiceGuideDeliveryUnit to send		308	* p_SGDU The ServiceGuideDeliveryUnit to send
298	* @param		309	* @param
299	* p_SGDU_Id The ServiceGuideDeliveryUnit id		310	* p_SGDU_Id The ServiceGuideDeliveryUnit id
300	* @param		311	* @param
301	* p_SGDD_Id The SGDD id		312	* p_SGDD_Id The SGDD id
302	* @param		313	* @param
303	* p_ConnectionId The http connection id		314	* p_ConnectionId The http connection id
304	* @param		315	* @param
305	* p_type The http request type		316	* p_type The http request type
306	* @verdict		317	* @verdict
307	* pass in case of that the response was successful send		318	* pass in case of that the response was successful send
308	* @verdict		319	* @verdict
309	* fail in case of the response was not successfule send		320	* fail in case of the response was not successfule send
310	* @verdict		321	* @verdict
311	* inconc a guard timer runs out of time		322	* inconc a guard timer runs out of time
312	*/		323	*/
313	function f_sendServiceGuideHttpResponse(ServiceGuideDeliveryUnit p_SGDU, charstring p_SGDU_Id, charstring p_SGDD_Id, template ServiceGuideRequestType p_type) runs on BCastNetworkControlComponent { // create SGDD var ServiceGuideDeliveryDescriptorType v_SGDD :=		324	function f_sendServiceGuideHttpResponse(ServiceGuideDeliveryUnit p_SGDU, charstring p_SGDU_Id, charstring p_SGDD_Id, charstring p_SGDD_Id, template ServiceGuideRequestType p_type) runs on BCastNetworkControlComponent { // create SGDD var ServiceGuideDeliveryDescriptorType v_SGDD :=
320	f_createSGDD4SGDU(p_SGDU, p_SGDU_Id, p_SGDD_Id, v_TOI4SGDU, v_startTime, v	<>	331	f_createSGDD4SGDU(p_SGDU, p_SGDU_Id, p_SGDD_Id, v_TOI4SGDU, 1, v_startTime,
321		=	332	
322	var ServiceGuidePayload v_Payload;		333	var ServiceGuidePayload v_Payload;
323			334	
324	if(valueof(p_type) == e_sgdd) {		335	if(valueof(p_type) == e_sgdd) {
325	v_Payload := valueof(m_def_ServiceGuidePayload_SGDD_SGDU({v_SGDD}, omit));		336	v_Payload := valueof(m_def_ServiceGuidePayload_SGDD_SGDU({v_SGDD}, omit));
326	}		337	}
327	else if (valueof(p_type) == e_sgdu) {		338	else if (valueof(p_type) == e_sgdu) {
328	v_Payload := valueof(m_def_ServiceGuidePayload_SGDD_SGDU(omit, p_SGDU));		339	v_Payload := valueof(m_def_ServiceGuidePayload_SGDD_SGDU(omit, p_SGDU));
329	}		340	}
330	else {		341	else {
331	v_Payload := valueof(m_def_ServiceGuidePayload_SGDD_SGDU({v_SGDD}, p_SGDU));		342	v_Payload := valueof(m_def_ServiceGuidePayload_SGDD_SGDU({v_SGDD}, p_SGDU));
332	}		343	}
333			344	
334	ic.send(m_def_HttpResponseRequest(v_Payload));		345	ic.send(m_def_HttpResponseRequest(v_Payload));
335	}		346	}
336			347	
		+	348	// change 1 (WK48/08): TSI Value correction - ESG payout
337	// change 13 (WK 6): send DVB-H bootstrap files via FLUTE	=	349	// change 13 (WK 6): send DVB-H bootstrap files via FLUTE
338	// CR (WK 34/2008): Flute Update Signalling		350	// CR (WK 34/2008): Flute Update Signalling
339	/**		351	/**

Datei: LibBCast_BCastNetworkControl_Functions.ttcn
(Fortsetzung)

340	*		352	*
341	* @desc		353	* @desc
342	* this function sends the DVB-H bootstrapping over the		354	* this function sends the DVB-H bootstrapping over the
343	* flute protocol and expects a successful flute response back.		355	* flute protocol and expects a successful flute response back.
344	* @param		356	* @param
345	* p_Id The Flute Session ID		357	* p_Id The Flute Session ID
346	* @param		358	* @param
		-+	359	* p_TSI_SGDD The TSI of the SGDD (for reverencing the appropriate SGDD FLUTE session)
			360	* @param
347	* p_FluteUpdateID The Flute Instance ID of the Flute Session	=	361	* p_FluteUpdateID The Flute Instance ID of the Flute Session
348	* @verdict		362	* @verdict
349	* pass in case the DVB-H bootstrap could be successful send		363	* pass in case the DVB-H bootstrap could be successful send
350	* @verdict		364	* @verdict
351	* fail in case the DVB-H bootstrap could not be successful send		365	* fail in case the DVB-H bootstrap could not be successful send
352	* @verdict		366	* @verdict
353	* inconc in case a guard timer expires		367	* inconc in case a guard timer expires
354	*/		368	*/
355	function f_broadcast_dvbh_bootstrap(UInt p_Id, UInt32 p_FluteUpdateID) runs on BCastNetwork	<>	369	function f_broadcast_dvbh_bootstrap(UInt p_Id, UInt16 p_TSI_SGDD, UInt32 p_FluteUpdateID)
356	var StartFluteSessionRequest v_SessionRequest;	=	370	var StartFluteSessionRequest v_SessionRequest;
357			371	
358	var ESGProviderDiscoveryType v_epdd := valueof(m_def_esg_prov_disc(372	var ESGProviderDiscoveryType v_epdd := valueof(m_def_esg_prov_disc(
359	PX_DVBH_ESG_PROV_DISC_URI,		373	PX_DVBH_ESG_PROV_DISC_URI,
360	PX_DVBH_ESG_PROV_DISC_NAME,		374	PX_DVBH_ESG_PROV_DISC_NAME,
361	PX_DVBH_ESG_PROV_DISC_ID));		375	PX_DVBH_ESG_PROV_DISC_ID));
362	var ESGAccessType v_ead := valueof(m_def_esg_access_ipv4(376	var ESGAccessType v_ead := valueof(m_def_esg_access_ipv4(
363	PX_SRC_IP,		377	PX_SRC_IP,
364	PX_SGDD_FLUTE_SESSION_IP,		378	PX_SGDD_FLUTE_SESSION_IP,
365	PX_SGDD_FLUTE_SESSION_PORT));	<>	379	PX_SGDD_FLUTE_SESSION_PORT,
			380	p_TSI_SGDD));
366		=	381	
367	var FDT_InstanceType v_fdt := f_createFDT4DVBH(int2str(v_endTime));		382	var FDT_InstanceType v_fdt := f_createFDT4DVBH(int2str(v_endTime));
368			383	
369	v_SessionRequest := valueof(m_def_StartFluteSessionRequest(384	v_SessionRequest := valueof(m_def_StartFluteSessionRequest(
370	p_Id,		385	p_Id,
371	PX_DVB_H_FLUTE_SESSION_IP,		386	PX_DVB_H_FLUTE_SESSION_IP,
372	PX_DVB_H_FLUTE_SESSION_PORT,		387	PX_DVB_H_FLUTE_SESSION_PORT,
373	p_FluteUpdateID,	//	388	p_FluteUpdateID,
374	v_fdt,		389	v_fdt,
375	m_DVBH_ESG_Descriptor_FluteSessionPayload(v_epdd, v_ead));		390	m_DVBH_ESG_Descriptor_FluteSessionPayload(v_epdd, v_ead));
376			391	
377	ac.send(v_SessionRequest);		392	ac.send(v_SessionRequest);
378	t_wait.start(PX_TWAIT);		393	t_wait.start(PX_TWAIT);
379	alt {		394	alt {
380	[] ac.receive(StartFluteSessionResponse: {0, *}) {		395	[] ac.receive(StartFluteSessionResponse: {0, *}) {
381	t_wait.stop;		396	t_wait.stop;
382	setverdict(pass);		397	setverdict(pass);
383	}		398	}
384	[] ac.receive {		399	[] ac.receive {
385	t_wait.stop;		400	t_wait.stop;
386	setverdict(fail);		401	setverdict(fail);
387	}		402	}
388	[] t_wait.timeout {		403	[] t_wait.timeout {
389	setverdict(inconc);		404	setverdict(inconc);
390	}		405	}
391	}		406	}
392	}		407	}
393			408	
394	// CR (WK 34/2008): Flute Update Signalling		409	// CR (WK 34/2008): Flute Update Signalling
395	/**		410	/**
396	*		411	*
397	* @desc		412	* @desc
398	* this function sends a ServiceGuideDeliveryDescriptor over the		413	* this function sends a ServiceGuideDeliveryDescriptor over the
399	* flute protocol and expects a successful flute response back.		414	* flute protocol and expects a successful flute response back.
400	* @param		415	* @param
401	* p_SGDD The SGDD which should be send		416	* p_SGDD The SGDD which should be send
402	* @param		417	* @param
403	* p_Id The Flute session id		418	* p_Id The Flute session id
404	* @param		419	* @param
405	* p_Encoding The encoding attribute of the		420	* p_Encoding The encoding attribute of the
406	* ServiceGuideDeliveryDescriptor		421	* ServiceGuideDeliveryDescriptor

Datei: LibBCast_BCastNetworkControl_Functions.ttcn
(Fortsetzung)

407	* @param	422	* @param
408	* p_FluteUpdateID The Flute Instance ID of the Flute Session.	423	* p_FluteUpdateID The Flute Instance ID of the Flute Session.
409	* @verdict	424	* @verdict
410	* pass in case the SGDD could be successful send	425	* pass in case the SGDD could be successful send
411	* @verdict	426	* @verdict
412	* fail in case the SGDD could not be successful send	427	* fail in case the SGDD could not be successful send
413	* @verdict	428	* @verdict
414	* inconc in case a guard timer expires	429	* inconc in case a guard timer expires
415	*/	430	*/
416	function f_broadcast_sgdd(431	function f_broadcast_sgdd(
417	ServiceGuideDeliveryDescriptorType p_SGDD,	432	ServiceGuideDeliveryDescriptorType p_SGDD,
418	UInt p_Id,	433	UInt p_Id,
419	template charstring p_Encoding,	434	template charstring p_Encoding,
420	UInt32 p_FluteUpdateID) runs on BCastNetworkControlComponent {	435	UInt32 p_FluteUpdateID) runs on BCastNetworkControlComponent {
421	var StartFluteSessionRequest v_SessionRequest;	436	var StartFluteSessionRequest v_SessionRequest;
422		437	
423	var FDT_InstanceType v_fdt := f_createFDT4SGDD("urn:oma:bcast:sgdd:"&int2str(p_Id),	438	var FDT_InstanceType v_fdt := f_createFDT4SGDD("urn:oma:bcast:sgdd:"&int2str(p_Id),
424	p_Encoding,	439	p_Encoding,
425	int2str(v_endTime),	440	int2str(v_endTime),
426	v_TOI4SGDD := v_TOI4SGDD + 1;	441	v_TOI4SGDD := v_TOI4SGDD + 1;
427		442	
428		443	
429	v_SessionRequest :=	444	v_SessionRequest :=
430	valueof(m_def_StartFluteSessionRequest(445	valueof(m_def_StartFluteSessionRequest(
431	p_Id,	446	p_Id,
432	PX_SGDD_FLUTE_SESSION_IP, // change 16 (WK 6)	447	PX_SGDD_FLUTE_SESSION_IP, // change 16 (WK 6)
433	PX_SGDD_FLUTE_SESSION_PORT, // change 16 (WK 6)	448	PX_SGDD_FLUTE_SESSION_PORT, // change 16 (WK 6)
434	p_FluteUpdateID, // CR (WK 34/2008):	449	p_FluteUpdateID, // CR (WK 34/2008):
435	v_fdt,	450	v_fdt,
436	m_SGDD_FluteSessionPayload(p_SGDD));	451	m_SGDD_FluteSessionPayload(p_SGDD));
437		452	
438	ac.send (v_SessionRequest);	453	ac.send (v_SessionRequest);
439	t_wait.start(PX_TWAIT);	454	t_wait.start(PX_TWAIT);
440	alt {	455	alt {
441	[] ac.receive (StartFluteSessionResponse: {0, *}) {	456	[] ac.receive (StartFluteSessionResponse: {0, *}) {
442	t_wait.stop;	457	t_wait.stop;
443	setverdict (pass);	458	setverdict (pass);
444	}	459	}
445	[] ac.receive {	460	[] ac.receive {
446	t_wait.stop;	461	t_wait.stop;
447	setverdict (fail);	462	setverdict (fail);
448	}	463	}
449	[] t_wait.timeout {	464	[] t_wait.timeout {
450	setverdict (inconc);	465	setverdict (inconc);
451	}	466	}
452	}	467	}
453	}	468	}
454		469	
455	// change 6 (WK18): dynamic setting of TOI number	470	// change 6 (WK18): dynamic setting of TOI number
456	// CR (WK 34/2008): Flute Update Signalling	471	// CR (WK 34/2008): Flute Update Signalling
457	/**	472	/**
458	*	473	*
459	* @desc	474	* @desc
460	* this function sends a ServiceGuideDeliveryUnit over the flute	475	* this function sends a ServiceGuideDeliveryUnit over the flute
461	* protocoal and expect a successful flute response back.	476	* protocoal and expect a successful flute response back.
462	* @param	477	* @param
463	* p_SGDU The ServiceGuideDeliveryUnit which should be send	478	* p_SGDU The ServiceGuideDeliveryUnit which should be send
464	* @param	479	* @param
465	* p_SGDU_Id the id of the ServiceGuideDeliveryUnit	480	* p_SGDU_Id the id of the ServiceGuideDeliveryUnit
466	* @param	481	* @param
467	* p_Id The flute session id	482	* p_Id The flute session id
468	* @param	483	* @param
469	* p_Encoding the encoding of the ServiceGuideDeliveryUnit	484	* p_Encoding the encoding of the ServiceGuideDeliveryUnit
470	* @param	485	* @param
471	* p_FluteUpdateID The Flute Instance ID of the Flute Session.	486	* p_FluteUpdateID The Flute Instance ID of the Flute Session.
472	* @verdict	487	* @verdict
473	* pass in case the SGDD could be successful send	488	* pass in case the SGDD could be successful send
474	* @verdict	489	* @verdict
475	* fail in case the SGDD could not be successful	490	* fail in case the SGDD could not be successful
476	* @verdict	491	* @verdict

Datei: LibBCast_BCastNetworkControl_Functions.ttcn
(Fortsetzung)

477	* inconc in case a guard timer expires	492	* inconc in case a guard timer expires
478	*/	493	*/
479	function f_broadcast_sgdu(494	function f_broadcast_sgdu(
480	ServiceGuideDeliveryUnit p_SGDU,	495	ServiceGuideDeliveryUnit p_SGDU,
481	AnyUri p_SGDU_Id,	496	AnyUri p_SGDU_Id,
482	UInt p_Id,	497	UInt p_Id,
483	template charstring p_Encoding,	498	template charstring p_Encoding,
484	UInt32 p_FluteUpdateID) runs on BCastNetworkControlComponent {	499	UInt32 p_FluteUpdateID) runs on BCastNetworkControlComponent {
485		500	
486	var FDT_InstanceType v_fdt := f_createFDT4SGDU("urn:oma:bcast:sgdu:"&int2str(p_Id),	501	var FDT_InstanceType v_fdt := f_createFDT4SGDU("urn:oma:bcast:sgdu:"&int2str(p_Id),
487	p_Encoding, // change 15	502	p_Encoding, // change 15
488	int2str(v_endTime),	503	int2str(v_endTime),
489	v_TOI4SGDU);	504	v_TOI4SGDU);
490	v_TOI4SGDU := v_TOI4SGDU + 1;	505	v_TOI4SGDU := v_TOI4SGDU + 1;
491		506	
492	var StartFluteSessionRequest v_SessionRequest;	507	var StartFluteSessionRequest v_SessionRequest;
493	v_SessionRequest :=	508	v_SessionRequest :=
494	valueof(m_def_StartFluteSessionRequest(509	valueof(m_def_StartFluteSessionRequest(
495	p_Id,	510	p_Id,
496	PX_SGDU_FLUTE_SESSION_IP, // change 16 (WK 6): usage	511	PX_SGDU_FLUTE_SESSION_IP, // change 16 (WK 6): usage
497	PX_SGDU_FLUTE_SESSION_PORT, // change 16 (WK 6): usage	512	PX_SGDU_FLUTE_SESSION_PORT, // change 16 (WK 6): usage
498	p_FluteUpdateID, // CR (WK 34/2008): Flute U	513	p_FluteUpdateID, // CR (WK 34/2008): Flute U
499	v_fdt,	514	v_fdt,
500	m_SGDU_FluteSessionPayload(p_SGDU));	515	m_SGDU_FluteSessionPayload(p_SGDU));
501		516	
502	dc.send (v_SessionRequest);	517	dc.send (v_SessionRequest);
503	t_wait.start(PX_TWAIT);	518	t_wait.start(PX_TWAIT);
504	alt {	519	alt {
505	[] dc.receive (StartFluteSessionResponse: {0, *}) {	520	[] dc.receive (StartFluteSessionResponse: {0, *}) {
506	t_wait.stop;	521	t_wait.stop;
507	setverdict (pass);	522	setverdict (pass);
508	}	523	}
509	[] dc.receive {	524	[] dc.receive {
510	t_wait.stop;	525	t_wait.stop;
511	setverdict (fail);	526	setverdict (fail);
512	}	527	}
513	[] t_wait.timeout {	528	[] t_wait.timeout {
514	setverdict (inconc);	529	setverdict (inconc);
515	}	530	}
516	}	531	}
517	}	532	}
518	}	533	}
519	}	534	}

Datei: LibBCast_ServiceGuide_Templates.ttcn

1	/**	=	1	/**
2	*		2	*
3	* @author		3	* @author
4	* ETSI CTI OMA BCAST CON Project		4	* ETSI CTI OMA BCAST CON Project
5	* @version		5	* @version
6	* \$Id\$		6	* \$Id\$
7	* @desc		7	* @desc
8	* This module provides service guide specific templates.		8	* This module provides service guide specific templates.
9	*/		9	*/
10	module LibBCast_ServiceGuide_Templates {		10	module LibBCast_ServiceGuide_Templates {
11	import from LibBCast_ServiceGuide_TypesAndValues all;		11	import from LibBCast_ServiceGuide_TypesAndValues all;
12	import from LibBCast_Common_TypesAndValues all;		12	import from LibBCast_Common_TypesAndValues all;
13	import from LibCommon_BasicTypesAndValues all;		13	import from LibCommon_BasicTypesAndValues all;
14	import from LibBCast_ServicePrimitives_TypesAndValues all;		14	import from LibBCast_ServicePrimitives_TypesAndValues all;
15	import from AtsBCast_ModuleParameters all;		15	import from AtsBCast_ModuleParameters all;
16	import from LibBCast_ModuleParameters {		16	import from LibBCast_ModuleParameters {
17	group DvvhBootstrapConfiguration		17	group DvvhBootstrapConfiguration
18	};		18	};
19			19	
20	group FDT {		20	group FDT {
21	/**		21	/**
22	*		22	*
23	* @desc		23	* @desc
24	* This template creates a file elment which is used in File		24	* This template creates a file elment which is used in File
25	* Delivery Table Instance (FDT).		25	* Delivery Table Instance (FDT).
26	* @param		26	* @param
27	* p_ContentLocation The contnent location for the file element		27	* p_ContentLocation The contnent location for the file element
28	* @param		28	* @param
29	* p_TOI The index of the file element in the FDT		29	* p_TOI The index of the file element in the FDT
30	* @see		30	* @see
31	* LibBCast_Common_TypesAndValues.FileDeliveryTableInstance		31	* LibBCast_Common_TypesAndValues.FileDeliveryTableInstance
32	* @see		32	* @see
33	* LibBCast_Common_TypesAndValues.FileElement		33	* LibBCast_Common_TypesAndValues.FileElement
34	*/		34	*/
35			35	
36	template LibBCast_ServicePrimitives_TypesAndValues.FileType m_def_FileType(36	template LibBCast_ServicePrimitives_TypesAndValues.FileType m_def_FileType(
37	AnyUri p_Content_Location,		37	AnyUri p_Content_Location,
38	UInt p_TOI,		38	UInt p_TOI,
39	template charstring p_ContentType,		39	template charstring p_ContentType,
40	template charstring p_ContentEncoding) := {		40	template charstring p_ContentEncoding) := {
41	MBMS_Session_Identitys := omit,		41	MBMS_Session_Identitys := omit,
42	Content_Location := p_Content_Location,		42	Content_Location := p_Content_Location,
43	TOI := p_TOI,		43	TOI := p_TOI,
44	Content_Length := omit,		44	Content_Length := omit,
45	Transfer_Length := omit,		45	Transfer_Length := omit,
46	Content_Type := p_ContentType,		46	Content_Type := p_ContentType,
47	Content_Encoding := p_ContentEncoding,		47	Content_Encoding := p_ContentEncoding,
48	Content_MD5 := omit,		48	Content_MD5 := omit,
49	FEC_OTI_FEC_Encoding_ID := omit,		49	FEC_OTI_FEC_Encoding_ID := omit,
50	FEC_OTI_FEC_Instance_ID := omit,		50	FEC_OTI_FEC_Instance_ID := omit,
51	FEC_OTI_Maximum_Source_Block_Length := omit,		51	FEC_OTI_Maximum_Source_Block_Length := omit,
52	FEC_OTI_Encoding_Symbol_Length := omit,		52	FEC_OTI_Encoding_Symbol_Length := omit,
53	FEC_OTI_Max_Number_of_Encoding_Symbols := omit,		53	FEC_OTI_Max_Number_of_Encoding_Symbols := omit,
54	FEC_OTI_Scheme_Specific_Info := omit,		54	FEC_OTI_Scheme_Specific_Info := omit,
55	Version_ID_Length := omit		55	Version_ID_Length := omit
56	}		56	}
57			57	
58	/**		58	/**
59	*		59	*
60	* @desc This template creates a File Delivery Table Instance (FDT)		60	* @desc This template creates a File Delivery Table Instance (FDT)
61	* @param p_FileList The file list wich should be included		61	* @param p_FileList The file list wich should be included
62	* @param p_Expires The expires time for the FDT		62	* @param p_Expires The expires time for the FDT
63	*/		63	*/
64			64	
65	template FDT_InstanceType m_def_FDT_Instance(65	template FDT_InstanceType m_def_FDT_Instance(
66	LibBCast_ServicePrimitives_TypesAndValues.FileList p_FileList,		66	LibBCast_ServicePrimitives_TypesAndValues.FileList p_FileList,
67	charstring p_Expires,		67	charstring p_Expires,
68	template charstring p_Content_Type,		68	template charstring p_Content_Type,
69	template UInt8 p_FEC_OTI_FEC_Encoding_ID) := {		69	template UInt8 p_FEC_OTI_FEC_Encoding_ID) := {
70	Files := p_FileList,		70	Files := p_FileList,
71	MBMS_Session_Identity_Expirys := omit,		71	MBMS_Session_Identity_Expirys := omit,

Datei: LibBCast_ServiceGuide_Templates.ttcn (Fortsetzung)

72	Expires := p_Expires,	72	Expires := p_Expires,
73	Complete := omit,	73	Complete := omit,
74	Content_Type := p_Content_Type, // necessary for xsd: FDT-Inst	74	Content_Type := p_Content_Type, // necessary for xsd: FDT-Inst
75	Content_Encoding := omit,	75	Content_Encoding := omit,
76	FEC_OTI_FEC_Encoding_ID := p_FEC_OTI_FEC_Encoding_ID, // necessary for xsd: FDT-Inst	76	FEC_OTI_FEC_Encoding_ID := p_FEC_OTI_FEC_Encoding_ID, // necessary for xsd: FDT-Inst
77	FEC_OTI_FEC_Instance_ID := omit,	77	FEC_OTI_FEC_Instance_ID := omit,
78	FEC_OTI_Maximum_Source_Block_Length := omit,	78	FEC_OTI_Maximum_Source_Block_Length := omit,
79	FEC_OTI_Encoding_Symbol_Length := omit,	79	FEC_OTI_Encoding_Symbol_Length := omit,
80	FEC_OTI_Max_Number_of_Encoding_Symbols := omit,	80	FEC_OTI_Max_Number_of_Encoding_Symbols := omit,
81	FEC_OTI_Scheme_Specific_Info:= omit,	81	FEC_OTI_Scheme_Specific_Info:= omit,
82	FullFDT := omit,	82	FullFDT := omit,
83	Version_ID_Length := omit	83	Version_ID_Length := omit
84	}	84	}
85		85	
86	}	86	}
87		87	
88	group SGDD {	88	group SGDD {
89	/**	89	/**
90	*	90	*
91	* @desc	91	* @desc
92	* This templates creates a Service Guide Delivery Descriptor	92	* This templates creates a Service Guide Delivery Descriptor
93	* (SGDD) Fragment.	93	* (SGDD) Fragment.
94	* @param	94	* @param
95	* p_TOI The fragment index in the FDT	95	* p_TOI The fragment index in the FDT
96	* @param	96	* @param
97	* p_Id The fragment id	97	* p_Id The fragment id
98	* @param	98	* @param
99	* p_Encoding The encoding of the fragment	99	* p_Encoding The encoding of the fragment
100	* @param	100	* @param
101	* p_Type The fragment type	101	* p_Type The fragment type
102	* @see	102	* @see
103	* @see	103	* @see
104	* LibBCast_ServiceGuide_TypesAndValues.Fragment	104	* LibBCast_ServiceGuide_TypesAndValues.Fragment
105	*/	105	*/
106	template FragmentType m_def_Fragment(106	template FragmentType m_def_Fragment(
107	UInt p_transportObjectId,	107	UInt p_transportObjectId,
108	UInt p_fragmentVersion, // change 06 (WK 6): dynamic setting of fr	108	UInt p_fragmentVersion, // change 06 (WK 6): dynamic setting of fr
109	AnyUri p_Id,	109	AnyUri p_Id,
110	UInt8 p_Encoding,	110	UInt8 p_Encoding,
111	template UInt8 p_Type) := {	111	template UInt8 p_Type) := {
112	transportID := p_transportObjectId,	112	transportID := p_transportObjectId,
113	id := p_Id,	113	id := p_Id,
114	version := p_fragmentVersion, // change 06 (WK 6): dynamic setting of fr	114	version := p_fragmentVersion, // change 06 (WK 6): dynamic setting of fr
115	validFrom := omit,	115	validFrom := omit,
116	validTo := omit,	116	validTo := omit,
117	fragmentEncoding := p_Encoding,	117	fragmentEncoding := p_Encoding,
118	fragmentType := p_Type,	118	fragmentType := p_Type,
119	GroupingCriteria := omit	119	GroupingCriteria := omit
120	}	120	}
121		121	
122	/**	122	/**
123	*	123	*
124	* @desc	124	* @desc
125	* This template creates a ServiceGuideDeliveryUnitField which	125	* This template creates a ServiceGuideDeliveryUnitField which
126	* is used by the Service Guide Delivery Descriptor (SGDD).	126	* is used by the Service Guide Delivery Descriptor (SGDD).
127	* @param	127	* @param
128	* p_TOI The Service Guide Delivery Unit (SGDU) index in the FDT	128	* p_TOI The Service Guide Delivery Unit (SGDU) index in the FDT
129	* @param	129	* @param
130	* p_location the location of the Service Guide Delivery Unit	130	* p_location the location of the Service Guide Delivery Unit
131	* (SGDU)	131	* (SGDU)
132	* @param	132	* @param
133	* p_FragmentList The fragment list of the	133	* p_FragmentList The fragment list of the
134	* ServiceGuideDeliveryUnitField (SGDU)	134	* ServiceGuideDeliveryUnitField (SGDU)
135	*/	135	*/
136	template ServiceGuideDeliveryUnitType m_def_ServiceGuideDeliveryUnitField(136	template ServiceGuideDeliveryUnitType m_def_ServiceGuideDeliveryUnitField(
137	UInt p_TOI,	137	UInt p_TOI,
138	AnyUri p_location,	138	AnyUri p_location,
139	FragmentList p_FragmentList) := {	139	FragmentList p_FragmentList) := {
140	transportObjectID := p_TOI,	140	transportObjectID := p_TOI,
141	contentLocation := p_location,	141	contentLocation := p_location,
142	validFrom := omit,	142	validFrom := omit,

Datei: LibBCast_ServiceGuide_Templates.ttcn (Fortsetzung)

143	validTo := omit,		143	validTo := omit,
144	Fragments := p_FragmentList		144	Fragments := p_FragmentList
145	}		145	}
146			146	
		-+	147	// change 1 (WK48/08): TSI Value correction - ESG payout
147	/**	=	148	/**
148	*		149	*
149	* @desc		150	* @desc
150	* This template creates a ServiceGuideDeliveryDescriptor		151	* This template creates a ServiceGuideDeliveryDescriptor
151	* (SGDD).		152	* (SGDD).
152	* @param		153	* @param
153	* p_Id The id of the SGDD		154	* p_Id The id of the SGDD
154	* @param		155	* @param
155	* p_ServiceGuideDeliveryUnitField The SGDU field of the SGDD		156	* p_ServiceGuideDeliveryUnitField The SGDU field of the SGDD
		-+	157	* @param
			158	* p_TSI_SGDU The TSI of the SGDU (for reverencing the appropriate SGDU FLUTE session)
			159	* @param
			160	* p_startTime The start time of the SGDU session
			161	* @param
			162	* p_stopTime The stop time of the SGDU session
156	*/	=	163	*/
157	template ServiceGuideDeliveryDescriptorType m_def_ServiceGuideDeliveryDescriptor(164	template ServiceGuideDeliveryDescriptorType m_def_ServiceGuideDeliveryDescriptor(
158	AnyUri p_Id,		165	AnyUri p_Id,
159	ServiceGuideDeliveryUnitType p_ServiceGuideDeliveryUnitField,		166	ServiceGuideDeliveryUnitType p_ServiceGuideDeliveryUnitField,
		-+	167	UInt16 p_TSI_SGDU, // change xxx (WK48/08): TSI Valu
160	UInt p_startTime, // change 01 (WK 6): global time	=	168	UInt p_startTime, // change 01 (WK 6): global time
161	UInt p_endTime) := { // change 01 (WK 6): global time		169	UInt p_endTime) := { // change 01 (WK 6): global time
162	id := p_Id,		170	id := p_Id,
163	version := 1,		171	version := 1,
164	NotificationReception := omit,		172	NotificationReception := omit,
165	BSMList := omit,		173	BSMList := omit,
166	DescriptorEntrys := {		174	DescriptorEntrys := {
167	{		175	{
168	GroupingCriteria := { // change 2		176	GroupingCriteria := { // change 2
169	TimeGroupingCriteria := {p_startTime,p_endTime},		177	TimeGroupingCriteria := {p_startTime,p_endTime},
170	GenreGroupingCriteria := omit,		178	GenreGroupingCriteria := omit,
171	BSMSelectors := omit,		179	BSMSelectors := omit,
172	ServiceCriteria := omit },		180	ServiceCriteria := omit },
173	Transport := {		181	Transport := {
174	PX_SGDU_FLUTE_SESSION_IP,PX_SGDU_FLUTE_SESSION_PORT, // char		182	PX_SGDU_FLUTE_SESSION_IP,PX_SGDU_FLUTE_SESSION_PORT, // char
175	PX_SRC_IP, 1, omit, omit },	<>	183	PX_SRC_IP, p_TSI_SGDU, omit, omit }, // char
176	AlternativeAccessURLs := omit,	=	184	AlternativeAccessURLs := omit,
177	ServiceGuideDeliveryUnits := {		185	ServiceGuideDeliveryUnits := {
178	p_ServiceGuideDeliveryUnitField }		186	p_ServiceGuideDeliveryUnitField }
179	}		187	}
180	},		188	},
181	PrivateExt := omit		189	PrivateExt := omit
182	}		190	}
183	}		191	}
184			192	
185	group SGDU {		193	group SGDU {
186			194	
187	group sdpFragments {		195	group sdpFragments {
188			196	
189	/**		197	/**
190	*		198	*
191	* @desc This is a default template for the SdpFragment.		199	* @desc This is a default template for the SdpFragment.
192	* @param p_fragmentId The id of the SDP fragment		200	* @param p_fragmentId The id of the SDP fragment
193	* @param p_content The content of the SDP fragment		201	* @param p_content The content of the SDP fragment
194	*/		202	*/
195	template SdpFragment m_def_SdpFragment(203	template SdpFragment m_def_SdpFragment(
196	charstring p_fragmentId,		204	charstring p_fragmentId,
197	charstring p_content) := {		205	charstring p_content) := {
198	validFrom := 0,		206	validFrom := 0,
199	validTo := 0,		207	validTo := 0,
200	fragmentId := p_fragmentId,		208	fragmentId := p_fragmentId,
201	sdpFragment := p_content		209	sdpFragment := p_content
202	}		210	}
203	}		211	}
204			212	
205	group xmlFragments {		213	group xmlFragments {

Datei: LibBCast_ServiceGuide_Templates.ttcn (Fortsetzung)

206		214	
207	// change 23 (WK18): missing parameter description	215	// change 23 (WK18): missing parameter description
208	/**	216	/**
209	*	217	*
210	* @desc	218	* @desc
211	* This template create a default PreviewData which includes a	219	* This template create a default PreviewData which includes a
212	* picture reference.	220	* picture reference.
213	* @param	221	* @param
214	* p_Id The Id of the PreviewData	222	* p_Id The Id of the PreviewData
215	* @param	223	* @param
216	* p_Version The version of the PreviewData	224	* p_Version The version of the PreviewData
217	* @param	225	* @param
218	* p_PictureId The picture reference id	226	* p_PictureId The picture reference id
219	* @param	227	* @param
220	* p_txt The preview data element description.	228	* p_txt The preview data element description.
221	* @param	229	* @param
222	* p_mimeType The preview data element MIME type	230	* p_mimeType The preview data element MIME type
223	*/	231	*/
224	template PreviewDataType m_def_picture_PreviewData(232	template PreviewDataType m_def_picture_PreviewData(
225	AnyUri p_Id,	233	AnyUri p_Id,
226	UInt p_Version,	234	UInt p_Version,
227	AnyUri p_PictureId,	235	AnyUri p_PictureId,
228	LanguageString p_txt,	236	LanguageString p_txt,
229	charstring p_mimeType) := {	237	charstring p_mimeType) := {
230	id := p_Id,	238	id := p_Id,
231	version := p_Version,	239	version := p_Version,
232	validFrom := omit,	240	validFrom := omit,
233	validTo := omit,	241	validTo := omit,
234	SMIL := omit,	242	SMIL := omit,
235	Video := omit,	243	Video := omit,
236	Audio := omit,	244	Audio := omit,
237	Picture := {	245	Picture := {
238	choice := { PictureURI := {p_PictureId} },	246	choice := { PictureURI := {p_PictureId} },
239	MIMEType := {omit, p_mimeType},// change 2 (WK18): file distributio	247	MIMEType := {omit, p_mimeType},// change 2 (WK18): file distributio
240	AlternativeTexts := {p_txt} },	248	AlternativeTexts := {p_txt} },
241	Texts := omit,	249	Texts := omit,
242	AccessReference := {idRef := PX_SGDU_ACCESS_ID_PROG_1, usage := 0}, // char	250	AccessReference := {idRef := PX_SGDU_ACCESS_ID_PROG_1, usage := 0}, // char
243	PrivateExt := omit	251	PrivateExt := omit
244	}	252	}
245		253	
246	/**	254	/**
247	*	255	*
248	* @desc This template create a Service Fragment which is used in a SGDU.	256	* @desc This template create a Service Fragment which is used in a SGDU.
249	* @param p_Id The service id	257	* @param p_Id The service id
250	* @param p_Name The service name	258	* @param p_Name The service name
251	* @param p_version The version of the service fragment	259	* @param p_version The version of the service fragment
252	* @param p_ServiceType The service type of the service	260	* @param p_ServiceType The service type of the service
253	*/	261	*/
254	template ServiceType m_def_Service(262	template ServiceType m_def_Service(
255	AnyUri p_Id,	263	AnyUri p_Id,
256	UInt p_version,	264	UInt p_version,
257	charstring p_Name,	265	charstring p_Name,
258	XmlElementUnsignedByteValue p_ServiceType) := {	266	XmlElementUnsignedByteValue p_ServiceType) := {
259	id := p_Id,	267	id := p_Id,
260	version := p_version,	268	version := p_version,
261	validFrom := omit,	269	validFrom := omit,
262	validTo := omit,	270	validTo := omit,
263	globalServiceID := omit,	271	globalServiceID := omit,
264	weight := omit,	272	weight := omit,
265	//serviceContentProtection := omit,	273	//serviceContentProtection := omit,
266	baseCID := omit,	274	baseCID := omit,
267	emergency := omit,	275	emergency := omit,
268	ProtectionKeyIDs := omit,	276	ProtectionKeyIDs := omit,
269	ServiceTypes := { {text_ := p_ServiceType} },	277	ServiceTypes := { {text_ := p_ServiceType} },
270	Names := { {lang := omit, text_:=p_Name} },	278	Names := { {lang := omit, text_:=p_Name} },
271	Descriptions := omit,	279	Descriptions := omit,
272	AudioLanguages := omit,	280	AudioLanguages := omit,
273	TextLanguages := omit,	281	TextLanguages := omit,
274	ParentalRatings := omit,	282	ParentalRatings := omit,
275	TargetUserProfiles := omit,	283	TargetUserProfiles := omit,
276	Genres := omit,	284	Genres := omit,

Datei: LibBCast_ServiceGuide_Templates.ttcn (Fortsetzung)

```
277         Extensions := omit,
278         PreviewDataReferences := omit,
279         BroadcastArea := omit,
280         TermsOfUses := omit,
281         PrivateExt := omit
282     }
283
284     /**
285      *
286      * @desc
287      *   This template create a Content Fragment which is used in a
288      *   SGDU.
289      * @param
290      *   p_Id the content id
291      * @param
292      *   p_Name the content name
293      * @param
294      *   p_ServiceId the referenced service id
295      */
296     template ContentType m_def_Content(
297         AnyUri p_Id,
298         UInt p_version,
299         charstring p_Name,
300         AnyUri p_ServiceId,
301         DateTime p_startTime,
302         DateTime p_stopTime) := {
303         id := p_Id,
304         version := p_version,
305         validFrom := omit,
306         validTo := omit,
307         globalContentID := omit,
308         emergency := omit,
309         //serviceContentProtection := omit,
310         baseCID := omit,
311         ServiceReferences := { { idRef := p_ServiceId, weight := omit } },
312         ProtectionKeyIDs := omit,
313         Names := { {lang := omit, text_:=p_Name} },
314         Descriptions := omit,
315         StartTime := { text_ := p_startTime },
316         EndTime := { text_ := p_stopTime },
317         AudioLanguages := omit,
318         TextLanguages := omit,
319         ParentalRatings := omit,
320         TargetUserProfiles := omit,
321         Genres := omit,
322         Extensions := omit,
323         PreviewDataReferences := omit,
324         BroadcastArea := omit,
325         TermsOfUses := omit,
326         PrivateExt := omit
327     }
328
329     /**
330      * @desc This template creates a SGDU Schedule Fragment
331      * @param p_Id the schedule id
332      * @param p_ServiceId the referenced service id
333      * @param p_ContentId the referenced content id
334      * @param p_StartTime the start time
335      * @param p_EndTime the end time
336      */
337     template ScheduleType m_def_Schedule(
338         AnyUri p_Id,
339         UInt p_version,
340         AnyUri p_ServiceId,
341         AnyUri p_ContentId,
342         UInt p_StartTime,
343         UInt p_EndTime) := {
344         id := p_Id,
345         version := p_version,
346         defaultSchedule := omit,
347         onDemand := omit,
```

```
285         Extensions := omit,
286         PreviewDataReferences := omit,
287         BroadcastArea := omit,
288         TermsOfUses := omit,
289         PrivateExt := omit
290     }
291
292     /**
293      *
294      * @desc
295      *   This template create a Content Fragment which is used in a
296      *   SGDU.
297      * @param
298      *   p_Id the content id
299      * @param
300      *   p_Name the content name
301      * @param
302      *   p_ServiceId the referenced service id
303      */
304     template ContentType m_def_Content(
305         AnyUri p_Id,
306         UInt p_version,
307         charstring p_Name,
308         AnyUri p_ServiceId,
309         DateTime p_startTime,
310         DateTime p_stopTime) := {
311         id := p_Id,
312         version := p_version,
313         validFrom := omit,
314         validTo := omit,
315         globalContentID := omit,
316         emergency := omit,
317         //serviceContentProtection := omit,
318         baseCID := omit,
319         ServiceReferences := { { idRef := p_ServiceId, weight := omit } },
320         ProtectionKeyIDs := omit,
321         Names := { {lang := omit, text_:=p_Name} },
322         Descriptions := omit,
323         StartTime := { text_ := p_startTime },
324         EndTime := { text_ := p_stopTime },
325         AudioLanguages := omit,
326         TextLanguages := omit,
327         ParentalRatings := omit,
328         TargetUserProfiles := omit,
329         Genres := omit,
330         Extensions := omit,
331         PreviewDataReferences := omit,
332         BroadcastArea := omit,
333         TermsOfUses := omit,
334         PrivateExt := omit
335     }
336
337     /**
338      * @desc This template creates a SGDU Schedule Fragment
339      * @param p_Id the schedule id
340      * @param p_ServiceId the referenced service id
341      * @param p_ContentId the referenced content id
342      * @param p_StartTime the start time
343      * @param p_EndTime the end time
344      */
345     template ScheduleType m_def_Schedule(
346         AnyUri p_Id,
347         UInt p_version,
348         AnyUri p_ServiceId,
349         AnyUri p_ContentId,
350         UInt p_StartTime,
351         UInt p_EndTime) := {
352         id := p_Id,
353         version := p_version,
354         defaultSchedule := omit,
355         onDemand := omit,
```

Datei: LibBCast_ServiceGuide_Templates.ttcn (Fortsetzung)

348	validFrom := omit,	356	validFrom := omit,
349	validTo := omit,	357	validTo := omit,
350	ServiceReference := {	358	ServiceReference := {
351	idRef := p_ServiceId,	359	idRef := p_ServiceId,
352	audioLanguageIDRef := omit,	360	audioLanguageIDRef := omit,
353	textLanguageIDRef := omit },	361	textLanguageIDRef := omit },
354	InteractivityDataReferences := omit,	362	InteractivityDataReferences := omit,
355	ContentReferences := {	363	ContentReferences := {
356	{	364	{
357	idRef := p_ContentId,	365	idRef := p_ContentId,
358	//contentLocation := p_ContentId, // Change No. 11 from OMA-IOP-7	366	//contentLocation := p_ContentId, // Change No. 11 from OMA-IOP-7
359	contentLocation := omit,	367	contentLocation := omit,
360	audioLanguageIDRef := omit,	368	audioLanguageIDRef := omit,
361	textLanguageIDRef := omit,	369	textLanguageIDRef := omit,
362	repeatPlayback := omit,	370	repeatPlayback := omit,
363	AutoStarts := omit,	371	AutoStarts := omit,
364	DistributionWindows := omit,	372	DistributionWindows := omit,
365	PresentationWindows := {	373	PresentationWindows := {
366	{	374	{
367	startTime := p_StartTime,	375	startTime := p_StartTime,
368	endTime := p_EndTime,	376	endTime := p_EndTime,
369	duration := omit,	377	duration := omit,
370	id := omit }	378	id := omit }
371	}	379	}
372	}	380	}
373	},	381	},
374	PreviewDataReferences := omit,	382	PreviewDataReferences := omit,
375	TermsOfUses := omit,	383	TermsOfUses := omit,
376	PrivateExt := omit	384	PrivateExt := omit
377	}	385	}
378		386	
379	/**	387	/**
380	*	388	*
381	* @desc This is a default template for a Schedule fragment which refers a interact	389	* @desc This is a default template for a Schedule fragment which refers a interact
382	* @param p_Id The id of the schedule fragment	390	* @param p_Id The id of the schedule fragment
383	* @param p_version The fragment version	391	* @param p_version The fragment version
384	* @param p_ServiceId The refered service id	392	* @param p_ServiceId The refered service id
385	* @param p_InteractivityDataId the referd interactivity data	393	* @param p_InteractivityDataId the referd interactivity data
386	* @param p_StartTime the start time	394	* @param p_StartTime the start time
387	* @param p_EndTime the end time	395	* @param p_EndTime the end time
388	*/	396	*/
389	template ScheduleType m_def_inter_Schedule(397	template ScheduleType m_def_inter_Schedule(
390	AnyUri p_Id,	398	AnyUri p_Id,
391	UInt p_version,	399	UInt p_version,
392	AnyUri p_ServiceId,	400	AnyUri p_ServiceId,
393	AnyUri p_InteractivityDataId) := {	401	AnyUri p_InteractivityDataId) := {
394	id := p_Id,	402	id := p_Id,
395	version := p_version,	403	version := p_version,
396	defaultSchedule := omit,	404	defaultSchedule := omit,
397	onDemand := omit,	405	onDemand := omit,
398	validFrom := omit,	406	validFrom := omit,
399	validTo := omit,	407	validTo := omit,
400	ServiceReference := {	408	ServiceReference := {
401	idRef := p_ServiceId,	409	idRef := p_ServiceId,
402	audioLanguageIDRef := omit,	410	audioLanguageIDRef := omit,
403	textLanguageIDRef := omit },	411	textLanguageIDRef := omit },
404	InteractivityDataReferences := {	412	InteractivityDataReferences := {
405	{	413	{
406	idRef := p_InteractivityDataId,	414	idRef := p_InteractivityDataId,
407	AutoStarts := omit,	415	AutoStarts := omit,
408	DistributionWindows := omit }	416	DistributionWindows := omit }
409	},	417	},
410	ContentReferences := omit,	418	ContentReferences := omit,
411	PreviewDataReferences := omit,	419	PreviewDataReferences := omit,
412	TermsOfUses := omit,	420	TermsOfUses := omit,
413	PrivateExt := omit	421	PrivateExt := omit
414	}	422	}
415		423	
416	/**	424	/**
417	*	425	*
418	* @desc This template create a AccessType for broadcast (SDP) which is used in the	426	* @desc This template create a AccessType for broadcast (SDP) which is used in the

Datei: LibBCast_ServiceGuide_Templates.ttcn (Fortsetzung)

419	* @param p_bcType the broadcast type	427	* @param p_bcType the broadcast type
420	* @param p_SDP the SDP which describes the session	428	* @param p_SDP the SDP which describes the session
421	*/	429	*/
422	template AccessTypeType m_def_AccessType_bc_sdp(430	template AccessTypeType m_def_AccessType_bc_sdp(
423	UInt8 p_bcType,	431	UInt8 p_bcType,
424	charstring p_sdpId) := {	432	charstring p_sdpId) := {
425	choice := {	433	choice := {
426	BroadcastServiceDelivery := {	434	BroadcastServiceDelivery := {
427	BDSType := {	435	BDSType := {
428	Type := {p_bcType},	436	Type := {p_bcType},
429	Versions := omit	437	Versions := omit
430	},	438	},
431	SessionDescription := {	439	SessionDescription := {
432	// choice := {SDPRef := {p_sdpId,p_sdpId}}, // Change No. 1	440	// choice := {SDPRef := {p_sdpId,p_sdpId}}, // Change No. 1
433	choice := {SDPRef := {uri := omit, idRef := p_sdpId}}, // e	441	choice := {SDPRef := {uri := omit, idRef := p_sdpId}}, // e
434	USBDRef := omit,	442	USBDRef := omit,
435	ADPRef := omit },	443	ADPRef := omit },
436	FileDescription := omit }	444	FileDescription := omit }
437	}	445	}
438	}	446	}
439		447	
440	// change 36 (WK18): according to TC BCAST-1.0-DIST-conf-102 -> access framgment cor	448	// change 36 (WK18): according to TC BCAST-1.0-DIST-conf-102 -> access framgment cor
441	/**	449	/**
442	*	450	*
443	* @desc This template create a AccessType for broadcast which is used in the acces	451	* @desc This template create a AccessType for broadcast which is used in the acces
444	* It contains the SDP within the 'SessionDescription'.	452	* It contains the SDP within the 'SessionDescription'.
445	* @param p_bcType the broadcast type	453	* @param p_bcType the broadcast type
446	* @param p_SDP the SDP which describes the session (Base64 encoded)	454	* @param p_SDP the SDP which describes the session (Base64 encoded)
447	*/	455	*/
448	template AccessTypeType m_def_AccessType(456	template AccessTypeType m_def_AccessType(
449	UInt8 p_bcType,	457	UInt8 p_bcType,
450	charstring p_sdpAsBase64) := {	458	charstring p_sdpAsBase64) := {
451	choice := {	459	choice := {
452	BroadcastServiceDelivery := {	460	BroadcastServiceDelivery := {
453	BDSType := {	461	BDSType := {
454	Type := {p_bcType},	462	Type := {p_bcType},
455	Versions := omit	463	Versions := omit
456	},	464	},
457	SessionDescription := {	465	SessionDescription := {
458	choice := {SDP := {space := "preserve", encoding := "base64	466	choice := {SDP := {space := "preserve", encoding := "base64
459	USBDRef := omit,	467	USBDRef := omit,
460	ADPRef := omit },	468	ADPRef := omit },
461	FileDescription := omit }	469	FileDescription := omit }
462	}	470	}
463	}	471	}
464		472	
465	/**	473	/**
466	*	474	*
467	* @desc This template create a SGDU Access Fragment.	475	* @desc This template create a SGDU Access Fragment.
468	* @param p_Id the access id	476	* @param p_Id the access id
469	* @param p_AccessType the access type	477	* @param p_AccessType the access type
470	* @param p_ServiceId the referenced service id	478	* @param p_ServiceId the referenced service id
471	* @param p_ScheduleId the referenced schedule id	479	* @param p_ScheduleId the referenced schedule id
472	* @param p_ServiceClass the service class	480	* @param p_ServiceClass the service class
473	*/	481	*/
474	template AccessType m_def_Access(482	template AccessType m_def_Access(
475	AnyUri p_Id,	483	AnyUri p_Id,
476	UInt p_version,	484	UInt p_version,
477	AccessTypeType p_AccessType,	485	AccessTypeType p_AccessType,
478	AnyUri p_ServiceId,	486	AnyUri p_ServiceId,
479	template AnyUri p_ScheduleId,	487	template AnyUri p_ScheduleId,
480	charstring p_ServiceClass) := {	488	charstring p_ServiceClass) := {
481	id := p_Id,	489	id := p_Id,
482	version := p_version,	490	version := p_version,
483	validFrom := omit,	491	validFrom := omit,
484	validTo := omit,	492	validTo := omit,
485	AccessType := p_AccessType,	493	AccessType := p_AccessType,
486	KeyManagementSystems := omit,	494	KeyManagementSystems := omit,
487	EncryptionTypes := omit,	495	EncryptionTypes := omit,
488	// choice := {ServiceReferences := { { idRef := p_ServiceId } } },	496	// choice := {ServiceReferences := { { idRef := p_ServiceId } } },
489	choice := {ScheduleReferences := { { idRef := p_ScheduleId, DistributionWir	497	choice := {ScheduleReferences := { { idRef := p_ScheduleId, DistributionWir

Datei: LibBCast_ServiceGuide_Templates.ttcn (Fortsetzung)

490	TerminalCapabilityRequirement := omit,	498	TerminalCapabilityRequirement := omit,
491	BandwidthRequirement := omit,	499	BandwidthRequirement := omit,
492	ServiceClassss := { {p_ServiceClass} },	500	ServiceClassss := { {p_ServiceClass} },
493	PreviewDataReferences := omit,	501	PreviewDataReferences := omit,
494	NotificationReception := omit,	502	NotificationReception := omit,
495	PrivateExt := omit	503	PrivateExt := omit
496	}	504	}
497		505	
498	// change 32 (WK18): (additional to 3) according to ETS; preview data fragment refe	506	// change 32 (WK18): (additional to 3) according to ETS; preview data fragment refe
499	template AccessType m_def_Access_without_Ref(AnyUri p_Id, UInt p_version, AccessTyp	507	template AccessType m_def_Access_without_Ref(AnyUri p_Id, UInt p_version, AccessTyp
500		508	
501	id := p_Id,	509	id := p_Id,
502	version := p_version,	510	version := p_version,
503	validFrom := omit,	511	validFrom := omit,
504	validTo := omit,	512	validTo := omit,
505	AccessType := p_AccessType,	513	AccessType := p_AccessType,
506	KeyManagementSystems := omit,	514	KeyManagementSystems := omit,
507	EncryptionTypes := omit,	515	EncryptionTypes := omit,
508	choice := omit,	516	choice := omit,
509	TerminalCapabilityRequirement := omit,	517	TerminalCapabilityRequirement := omit,
510	BandwidthRequirement := omit,	518	BandwidthRequirement := omit,
511	ServiceClassss := { {p_ServiceClass} },	519	ServiceClassss := { {p_ServiceClass} },
512	PreviewDataReferences := omit,	520	PreviewDataReferences := omit,
513	NotificationReception := omit,	521	NotificationReception := omit,
514	PrivateExt := omit	522	PrivateExt := omit
515	}	523	}
516		524	
517		525	
518	/**	526	/**
519	*	527	*
520	* @desc This template is a default template for PurchaseItem fragment.	528	* @desc This template is a default template for PurchaseItem fragment.
521	* @param p_Id The id of the PurchaseItem	529	* @param p_Id The id of the PurchaseItem
522	* @param p_Version The Version of the PurchaseItem	530	* @param p_Version The Version of the PurchaseItem
523	* @param p_GlobalPurchaseId The global purchase id	531	* @param p_GlobalPurchaseId The global purchase id
524	* @param p_ServiceID The serive id of the referenced service fragment	532	* @param p_ServiceID The serive id of the referenced service fragment
525	* @param p_Name The name of the PurchaseItem	533	* @param p_Name The name of the PurchaseItem
526	*/	534	*/
527	template PurchaseItemType m_def_PurchaseItem(535	template PurchaseItemType m_def_PurchaseItem(
528	AnyUri p_Id,	536	AnyUri p_Id,
529	UInt p_Version,	537	UInt p_Version,
530	AnyUri p_GlobalPurchaseId,	538	AnyUri p_GlobalPurchaseId,
531	AnyUri p_ServiceID,	539	AnyUri p_ServiceID,
532	charstring p_Name) := {	540	charstring p_Name) := {
533	id := p_Id,	541	id := p_Id,
534	version := p_Version,	542	version := p_Version,
535	validFrom := omit,	543	validFrom := omit,
536	validTo := omit,	544	validTo := omit,
537	globalPurchaseItemId := p_GlobalPurchaseId,	545	globalPurchaseItemId := p_GlobalPurchaseId,
538	binaryPurchaseItemId := omit,	546	binaryPurchaseItemId := omit,
539	weight := omit,	547	weight := omit,
540	closed := omit,	548	closed := omit,
541	choice := {ServiceReferences := { { idRef := p_ServiceID } } },	549	choice := {ServiceReferences := { { idRef := p_ServiceID } } },
542	ProtectionKeyIDs := omit,	550	ProtectionKeyIDs := omit,
543	Names := { {lang := omit, text_:=p_Name} },	551	Names := { {lang := omit, text_:=p_Name} },
544	Descriptions := omit,	552	Descriptions := omit,
545	StartTime := omit,	553	StartTime := omit,
546	EndTime := omit,	554	EndTime := omit,
547	ParentalRatings := omit,	555	ParentalRatings := omit,
548	Extensions := omit,	556	Extensions := omit,
549	DependencyReferences := omit,	557	DependencyReferences := omit,
550	ExclusionReferences := omit,	558	ExclusionReferences := omit,
551	PrivateExt := omit	559	PrivateExt := omit
552	}	560	}
553		561	
554	/**	562	/**
555	*	563	*
556	* @desc This template is a default template for PurchaseData.	564	* @desc This template is a default template for PurchaseData.
557	* @param p_Id The id of the PurchaseData	565	* @param p_Id The id of the PurchaseData
558	* @param p_Version The version of the PurchaseData	566	* @param p_Version The version of the PurchaseData
559	* @param p_Description the description for the PurchaseData fragment	567	* @param p_Description the description for the PurchaseData fragment
560	* @param p_PurchaseItemId The id of the referenced of the PurchaseItem	568	* @param p_PurchaseItemId The id of the referenced of the PurchaseItem

Datei: LibBCast_ServiceGuide_Templates.ttcn (Fortsetzung)

561	* @param p_PurchaseChannelId The id of the referenced of the PurchaseChannel	569	* @param p_PurchaseChannelId The id of the referenced of the PurchaseChannel
562	* @param p_PriceInfo The PriceInfo field. It could be omit	570	* @param p_PriceInfo The PriceInfo field. It could be omit
563	*/	571	*/
564	template PurchaseDataType m_def_PurchaseData(572	template PurchaseDataType m_def_PurchaseData(
565	AnyUri p_Id,	573	AnyUri p_Id,
566	UInt p_Version,	574	UInt p_Version,
567	template LanguageStringList p_Descriptions,	575	template LanguageStringList p_Descriptions,
568	AnyUri p_PurchaseItemId,	576	AnyUri p_PurchaseItemId,
569	AnyUri p_PurchaseChannelId,	577	AnyUri p_PurchaseChannelId,
570	template PriceInfoType p_PriceInfo	578	template PriceInfoType p_PriceInfo
571) := {	579) := {
572	id := p_Id,	580	id := p_Id,
573	version := p_Version,	581	version := p_Version,
574	validFrom := omit,	582	validFrom := omit,
575	validTo := omit,	583	validTo := omit,
576	Descriptions := p_Descriptions,	584	Descriptions := p_Descriptions,
577	PriceInfo := p_PriceInfo,	585	PriceInfo := p_PriceInfo,
578	PromotionInfos := omit,	586	PromotionInfos := omit,
579	Extensions := omit,	587	Extensions := omit,
580	PurchaseItemReference := { idRef := p_PurchaseItemId },	588	PurchaseItemReference := { idRef := p_PurchaseItemId },
581	PurchaseChannelReferences := { { idRef := p_PurchaseChannelId } },	589	PurchaseChannelReferences := { { idRef := p_PurchaseChannelId } },
582	PreviewDataReferences := omit,	590	PreviewDataReferences := omit,
583	TermsOfUses := omit,	591	TermsOfUses := omit,
584	PrivateExt := omit	592	PrivateExt := omit
585	}	593	}
586	/**	594	/**
587	*	595	*
588	* @desc Specifies the price information of the purchase item associated with this	596	* @desc Specifies the price information of the purchase item associated with this
589	* @param p_subscriptionType	597	* @param p_subscriptionType
590	* @param p_monetaryPrice	598	* @param p_monetaryPrice
591	* @param p_currency	599	* @param p_currency
592	*/	600	*/
593	template PriceInfoType m_PriceInfo(601	template PriceInfoType m_PriceInfo(
594	UInt p_subscriptionType,	602	UInt p_subscriptionType,
595	float p_monetaryPrice,	603	float p_monetaryPrice,
596	charstring p_currency) := {	604	charstring p_currency) := {
597	subscriptionType := p_subscriptionType,	605	subscriptionType := p_subscriptionType,
598	MonetaryPrices := {	606	MonetaryPrices := {
599	{ currency := p_currency, text_ := p_monetaryPrice }	607	{ currency := p_currency, text_ := p_monetaryPrice }
600	},	608	},
601	TotalNumberToken := omit,	609	TotalNumberToken := omit,
602	SubscriptionPeriod := omit	610	SubscriptionPeriod := omit
603	}	611	}
604		612	
605	/**	613	/**
606	*	614	*
607	* @desc This is a default template for the PurchaseChannel fragment.	615	* @desc This is a default template for the PurchaseChannel fragment.
608	* @param p_Id The id of the PurchaseChannel	616	* @param p_Id The id of the PurchaseChannel
609	* @param p_Version The version of the PurchaseChannel	617	* @param p_Version The version of the PurchaseChannel
610	* @param p_Name The name for PurchaseChannel	618	* @param p_Name The name for PurchaseChannel
611	* @param p_PurchaseUrl and the url of the subscription site	619	* @param p_PurchaseUrl and the url of the subscription site
612	*/	620	*/
613	template PurchaseChannelType m_def_PurchaseChannel(621	template PurchaseChannelType m_def_PurchaseChannel(
614	AnyUri p_Id,	622	AnyUri p_Id,
615	UInt p_Version,	623	UInt p_Version,
616	charstring p_Name,	624	charstring p_Name,
617	AnyUri p_PurchaseUrl) := {	625	AnyUri p_PurchaseUrl) := {
618	id := p_Id,	626	id := p_Id,
619	version := p_Version,	627	version := p_Version,
620	validFrom := omit,	628	validFrom := omit,
621	validTo := omit,	629	validTo := omit,
622	rightsIssuerURI := omit,	630	rightsIssuerURI := omit,
623	Names := { {lang := omit, text_:=p_Name} },	631	Names := { {lang := omit, text_:=p_Name} },
624	Descriptions := omit,	632	Descriptions := omit,
625	ContactInfo := omit,	633	ContactInfo := omit,
626	PortalURL := omit,	634	PortalURL := omit,
627	PurchaseURLs := { {p_PurchaseUrl} },	635	PurchaseURLs := { {p_PurchaseUrl} },
628	Extensions := omit,	636	Extensions := omit,
629	PrivateExt := omit	637	PrivateExt := omit
630	}	638	}
631		639	

Datei: LibBCast_ServiceGuide_Templates.ttcn (Fortsetzung)

632		640	
633	/**	641	/**
634	*	642	*
635	* @desc This this a default InteractivityData template.	643	* @desc This this a default InteractivityData template.
636	* @param p_Id The id of the InteractivityData	644	* @param p_Id The id of the InteractivityData
637	* @param p_Version The version of the InteractivityData	645	* @param p_Version The version of the InteractivityData
638	* @param p_PreListenIndicator a boolean to indecate how the terminal should handle	646	* @param p_PreListenIndicator a boolean to indecate how the terminal should handle
639	* @param p_InteractivityMediaDocumentGroupId The id of a media document group	647	* @param p_InteractivityMediaDocumentGroupId The id of a media document group
640	* @param p_ServiceId the referenced service fragment id	648	* @param p_ServiceId the referenced service fragment id
641	*/	649	*/
642	template InteractivityDataType m_def_InteractivityData(650	template InteractivityDataType m_def_InteractivityData(
643	AnyUri p_Id,	651	AnyUri p_Id,
644	UInt p_Version,	652	UInt p_Version,
645	boolean p_PreListenIndicator,	653	boolean p_PreListenIndicator,
646	AnyUri p_InteractivityMediaDocumentGroupId,	654	AnyUri p_InteractivityMediaDocumentGroupId,
647	AnyUri p_ServiceId,	655	AnyUri p_ServiceId,
648	AnyUri p_ContentId, // change 33 (WK18): (additional to 3) according to ETS	656	AnyUri p_ContentId, // change 33 (WK18): (additional to 3) according to ETS
649	charstring p_InteractivityType) := {	657	charstring p_InteractivityType) := {
650	id := p_Id,	658	id := p_Id,
651	version := p_Version,	659	version := p_Version,
652	validFrom := omit,	660	validFrom := omit,
653	validTo := omit,	661	validTo := omit,
654	preListenIndicator := p_PreListenIndicator,	662	preListenIndicator := p_PreListenIndicator,
655	interactivityMediaDocumentPointer :=	663	interactivityMediaDocumentPointer :=
656	p_InteractivityMediaDocumentGroupId,	664	p_InteractivityMediaDocumentGroupId,
657	// InteractivityTypes := omit,	665	// InteractivityTypes := omit,
658	InteractivityTypes := {{omit, p_InteractivityType}}, // change 24 (WK18):	666	InteractivityTypes := {{omit, p_InteractivityType}}, // change 24 (WK18):
659	ServiceReference := { idRef := p_ServiceId },	667	ServiceReference := { idRef := p_ServiceId },
660	// choice := omit,	668	// choice := omit,
661	choice := {	669	choice := {
662	ContentReferences := {{p_ContentId}} // change 33 (WK18): (addition	670	ContentReferences := {{p_ContentId}} // change 33 (WK18): (addition
663	},	671	},
664	InteractivityDelivery := omit,	672	InteractivityDelivery := omit,
665	Extension := omit,	673	Extension := omit,
666	BackOffTiming := omit,	674	BackOffTiming := omit,
667	TermsofUses := omit,	675	TermsofUses := omit,
668	PrivateExt := omit	676	PrivateExt := omit
669	}	677	}
670	}	678	}
671		679	
672	group ServiceGuideFragments {	680	group ServiceGuideFragments {
673	/**	681	/**
674	*	682	*
675	* @desc This template is a default template for a xml ServiceGuideFragment.	683	* @desc This template is a default template for a xml ServiceGuideFragment.
676	* @param p_fragmentTransportId The Tranport ID of the fragment	684	* @param p_fragmentTransportId The Tranport ID of the fragment
677	* @param p_fragmentVersion The fragment version	685	* @param p_fragmentVersion The fragment version
678	* @param p_xmlFragment the the xml fragment which should be included	686	* @param p_xmlFragment the the xml fragment which should be included
679	*/	687	*/
680	template ServiceGuideFragment m_xml_ServiceGuideFragment(688	template ServiceGuideFragment m_xml_ServiceGuideFragment(
681	UInt32 p_fragmentTransportId,	689	UInt32 p_fragmentTransportId,
682	UInt32 p_fragmentVersion,	690	UInt32 p_fragmentVersion,
683	XmlFragment p_xmlFragment) := {	691	XmlFragment p_xmlFragment) := {
684	fragmentTransportId := p_fragmentTransportId,	692	fragmentTransportId := p_fragmentTransportId,
685	fragmentVersion := p_fragmentVersion,	693	fragmentVersion := p_fragmentVersion,
686	fragmentType := { xmlFragment := p_xmlFragment }	694	fragmentType := { xmlFragment := p_xmlFragment }
687	}	695	}
688		696	
689	/**	697	/**
690	*	698	*
691	* @desc This template is a default template for a sdp ServiceGuideFragment.	699	* @desc This template is a default template for a sdp ServiceGuideFragment.
692	* @param p_fragmentTransportId The Tranport ID of the fragment	700	* @param p_fragmentTransportId The Tranport ID of the fragment
693	* @param p_fragmentVersion The fragment version	701	* @param p_fragmentVersion The fragment version
694	* @param p_sdpFragment the the sdp fragment which should be included	702	* @param p_sdpFragment the the sdp fragment which should be included
695	*/	703	*/
696	template ServiceGuideFragment m_sdp_ServiceGuideFragment(704	template ServiceGuideFragment m_sdp_ServiceGuideFragment(
697	UInt32 p_fragmentTransportId,	705	UInt32 p_fragmentTransportId,
698	UInt32 p_fragmentVersion,	706	UInt32 p_fragmentVersion,
699	SdpFragment p_sdpFragment) := {	707	SdpFragment p_sdpFragment) := {
700	fragmentTransportId := p_fragmentTransportId,	708	fragmentTransportId := p_fragmentTransportId,
701	fragmentVersion := p_fragmentVersion,	709	fragmentVersion := p_fragmentVersion,
702	fragmentType := { sdpFragment := p_sdpFragment }	710	fragmentType := { sdpFragment := p_sdpFragment }

Datei: LibBCast_ServiceGuide_Templates.ttcn (Fortsetzung)

703	}	711	}
704		712	
705	/**	713	/**
706	*	714	*
707	* @desc This template is a default template for a mbms ServiceGuideFragment.	715	* @desc This template is a default template for a mbms ServiceGuideFragment.
708	* @param p_fragmentTransportId The Tranport ID of the fragment	716	* @param p_fragmentTransportId The Tranport ID of the fragment
709	* @param p_fragmentVersion The fragment version	717	* @param p_fragmentVersion The fragment version
710	* @param p_mbmsUsbdFragment the the mbms fragment which should be included	718	* @param p_mbmsUsbdFragment the the mbms fragment which should be included
711	*/	719	*/
712	template ServiceGuideFragment m_mbmsUsbd_ServiceGuideFragment(720	template ServiceGuideFragment m_mbmsUsbd_ServiceGuideFragment(
713	UInt32 p_fragmentTransportId,	721	UInt32 p_fragmentTransportId,
714	UInt32 p_fragmentVersion,	722	UInt32 p_fragmentVersion,
715	MbmsUsbdFragment p_mbmsUsbdFragment) := {	723	MbmsUsbdFragment p_mbmsUsbdFragment) := {
716	fragmentTransportId := p_fragmentTransportId,	724	fragmentTransportId := p_fragmentTransportId,
717	fragmentVersion := p_fragmentVersion,	725	fragmentVersion := p_fragmentVersion,
718	fragmentType := { mbmsUsbdFragment := p_mbmsUsbdFragment }	726	fragmentType := { mbmsUsbdFragment := p_mbmsUsbdFragment }
719	}	727	}
720		728	
721	/**	729	/**
722	*	730	*
723	* @desc This template is a default template for a apd ServiceGuideFragment.	731	* @desc This template is a default template for a apd ServiceGuideFragment.
724	* @param p_fragmentTransportId The Tranport ID of the fragment	732	* @param p_fragmentTransportId The Tranport ID of the fragment
725	* @param p_fragmentVersion The fragment version	733	* @param p_fragmentVersion The fragment version
726	* @param p_adpFragment the the apd fragment which should be included	734	* @param p_adpFragment the the apd fragment which should be included
727	*/	735	*/
728	template ServiceGuideFragment m_adp_ServiceGuideFragment(736	template ServiceGuideFragment m_adp_ServiceGuideFragment(
729	UInt32 p_fragmentTransportId,	737	UInt32 p_fragmentTransportId,
730	UInt32 p_fragmentVersion,	738	UInt32 p_fragmentVersion,
731	FileAdp p_adpFragment) := {	739	FileAdp p_adpFragment) := {
732	fragmentTransportId := p_fragmentTransportId,	740	fragmentTransportId := p_fragmentTransportId,
733	fragmentVersion := p_fragmentVersion,	741	fragmentVersion := p_fragmentVersion,
734	fragmentType := { adpFragment := p_adpFragment }	742	fragmentType := { adpFragment := p_adpFragment }
735	}	743	}
736	}	744	}
737	}	745	}
738		746	
739		747	
740	// change 13 (WK 6): send DVB-H bootstrap files via FLUTE	748	// change 13 (WK 6): send DVB-H bootstrap files via FLUTE
741	group DVBH_Bootstrap {	749	group DVBH_Bootstrap {
742	/**	750	/**
743	*	751	*
744	* @desc	752	* @desc
745	* This is a default template for the ESGProviderDiscoveryType	753	* This is a default template for the ESGProviderDiscoveryType
746	* @param	754	* @param
747	* p_providerURI The provider URI	755	* p_providerURI The provider URI
748	* @param	756	* @param
749	* p_providerName The provider name	757	* p_providerName The provider name
750	* @param	758	* @param
751	* p_providerID The provider ID	759	* p_providerID The provider ID
752	*/	760	*/
753	template ESGProviderDiscoveryType m_def_esg_prov_disc(charstring p_providerURI, charstring	761	template ESGProviderDiscoveryType m_def_esg_prov_disc(charstring p_providerURI, charstring
754	ServiceProviders := {	762	ServiceProviders := {
755	{	763	{
756	format := "urn:oma:xml:bcast:sg:fragments:1.0",	764	format := "urn:oma:xml:bcast:sg:fragments:1.0",
757	ProviderURI := {text_ := p_providerURI},	765	ProviderURI := {text_ := p_providerURI},
758	ProviderName := {text_ := p_providerName},	766	ProviderName := {text_ := p_providerName},
759	ProviderID := {text_ := p_providerID}	767	ProviderID := {text_ := p_providerID}
760	}	768	}
761	}	769	}
762	}	770	}
763		771	
		-+	772
764	/**	=	773
765	*		774
766	* @desc		775
767	* This is a default template for the ESGAccessType		776
768	* @param		777
769	* p_srcIpAddress The source IPv4 address		778
770	* @param		779
771	* p_dstIpAddress The destination IPv4 address		780
772	* @param		781

Datei: LibBCast_ServiceGuide_Templates.ttcn (Fortsetzung)

773	* p_dstIpPort The destination port		782	* p_dstIpPort The destination port
		++	783	* @param
			784	* p_TSI_SGDD The TSI of the SGDD (for reverencing the appropriate SGDD)
774	*/	=	785	*/
775	template ESGAccessType m_def_esg_access_ipv4(charstring p_srcIpAddress, charstring p_dstIp	<>	786	template ESGAccessType m_def_esg_access_ipv4(charstring p_srcIpAddress, charstring p_dstIp
776	ESGEntryVersion := 1,	=	787	ESGEntryVersion := 1,
777	MultipleStreamTransport := false,		788	MultipleStreamTransport := false,
778	IPVersion6 := false,		789	IPVersion6 := false,
779	ProviderID := 1,		790	ProviderID := 1,
780	SrcIpAddress := p_srcIpAddress,		791	SrcIpAddress := p_srcIpAddress,
781	DstIpAddress := p_dstIpAddress,		792	DstIpAddress := p_dstIpAddress,
782	DstIpPort := p_dstIpPort,		793	DstIpPort := p_dstIpPort,
783	TSI := 0	<>	794	TSI := p_TSI_SGDD
784	}	=	795	}
785	}		796	}
786			797	
787	group IMD {		798	group IMD {
788	/**		799	/**
789	*		800	*
790	* @desc		801	* @desc
791	* This is a default template for the InteractivityMediaDocumentType		802	* This is a default template for the InteractivityMediaDocumentType
792	* @param		803	* @param
793	* p_GroupID The id of the group // change 99 (WK 6): Clerical		804	* p_GroupID The id of the group // change 99 (WK 6): Clerical
794	* @param		805	* @param
795	* p_GroupPosition The position of the group		806	* p_GroupPosition The position of the group
796	* @param		807	* @param
797	* p_id The media document id // change 99 (WK 6): Clerical		808	* p_id The media document id // change 99 (WK 6): Clerical
798	* @param		809	* @param
799	* p_MediaObjectGroupList the media object group list		810	* p_MediaObjectGroupList the media object group list
800	*/		811	*/
801	template InteractivityMediaDocumentType m_def_InteractivityMediaDocument (812	template InteractivityMediaDocumentType m_def_InteractivityMediaDocument (
802	AnyUri p_GroupID,		813	AnyUri p_GroupID,
803	UInt16 p_GroupPosition,		814	UInt16 p_GroupPosition,
804	AnyUri p_id,		815	AnyUri p_id,
805	UInt p_version,		816	UInt p_version,
806	MediaObjectGroupList p_MediaObjectGroups) := {		817	MediaObjectGroupList p_MediaObjectGroups) := {
807	groupID := p_GroupID,		818	groupID := p_GroupID,
808	groupPosition := p_GroupPosition,		819	groupPosition := p_GroupPosition,
809	id := p_id,		820	id := p_id,
810	version := p_version,		821	version := p_version,
811	vaildFrom := omit,		822	vaildFrom := omit,
812	valudTo := omit,		823	valudTo := omit,
813	MediaObjectGroups := p_MediaObjectGroups,		824	MediaObjectGroups := p_MediaObjectGroups,
814	PrivateExt := omit		825	PrivateExt := omit
815	}		826	}
816			827	
817	/**		828	/**
818	*		829	*
819	* @desc		830	* @desc
820	* This is a default MediaObjectGroupType template		831	* This is a default MediaObjectGroupType template
821	* @param		832	* @param
822	* p_id The id of the group		833	* p_id The id of the group
823	* @param		834	* @param
824	* p_StartMediaFlag the start media flag		835	* p_StartMediaFlag the start media flag
825	* @param		836	* @param
826	* p_MediaObjectSetList the media object set list		837	* p_MediaObjectSetList the media object set list
827	*/		838	*/
828	template MediaObjectGroupType m_def_MediaObjectGroup (839	template MediaObjectGroupType m_def_MediaObjectGroup (
829	AnyUri p_id,		840	AnyUri p_id,
830	boolean p_startMediaFlag,		841	boolean p_startMediaFlag,
831	template MediaObjectSetList p_MediaObjectSets) := {		842	template MediaObjectSetList p_MediaObjectSets) := {
832	id := p_id,		843	id := p_id,
833	startMediaFlag := p_startMediaFlag,		844	startMediaFlag := p_startMediaFlag,
834	ActionDescriptor := omit,		845	ActionDescriptor := omit,
835	BackOffTiming := omit,		846	BackOffTiming := omit,
836	MediaObjectSets := p_MediaObjectSets,		847	MediaObjectSets := p_MediaObjectSets,
837	SMSTemplate := omit,		848	SMSTemplate := omit,
838	EmailTemplate := omit,		849	EmailTemplate := omit,
839	VoiceCall := omit,		850	VoiceCall := omit,
840	WebLink := omit,		851	WebLink := omit,
841	AlternativeText := omit		852	AlternativeText := omit

Datei: LibBCast_ServiceGuide_Templates.ttcn (Fortsetzung)

842	}	853	}
843		854	
844	/**	855	/**
845	*	856	*
846	* @desc	857	* @desc
847	* This is a default MediaObjectSetType template	858	* This is a default MediaObjectSetType template
848	* @param	859	* @param
849	* p_content_Type The content type of the MediaObjectSetElement	860	* p_content_Type The content type of the MediaObjectSetElement
850	* @param	861	* @param
851	* p_content_Location The location of the content	862	* p_content_Location The location of the content
852	* @param	863	* @param
853	* p_objects The list of media objects or omit	864	* p_objects The list of media objects or omit
854	*/	865	*/
855	template MediaObjectSetType m_def_mediaObjectSet(866	template MediaObjectSetType m_def_mediaObjectSet(
856	charstring p_Content_Type,	867	charstring p_Content_Type,
857	AnyUri p_Content_Location,	868	AnyUri p_Content_Location,
858	template ObjectList p_Objects) := {	869	template ObjectList p_Objects) := {
859	relativePreference := omit,	870	relativePreference := omit,
860	Content_Type := p_Content_Type,	871	Content_Type := p_Content_Type,
861	Content_Location := p_Content_Location,	872	Content_Location := p_Content_Location,
862	Descriptions := omit,	873	Descriptions := omit,
863	Objects := p_Objects,	874	Objects := p_Objects,
864	File := omit	875	File := omit
865	}	876	}
866		877	
867	/**	878	/**
868	*	879	*
869	* @desc	880	* @desc
870	* This is a default ObjectType template	881	* This is a default ObjectType template
871	* @param	882	* @param
872	* p_Content_Location the location of the ObjectType	883	* p_Content_Location the location of the ObjectType
873	* @param	884	* @param
874	* p_Content_Type the type of the ObjectType	885	* p_Content_Type the type of the ObjectType
875	* @param	886	* @param
876	* p_start the start flag of the ObjectType	887	* p_start the start flag of the ObjectType
877	* @param	888	* @param
878	* p_PartTypes the part type list	889	* p_PartTypes the part type list
879	*/	890	*/
880	template ObjectType m_def_Object(891	template ObjectType m_def_Object(
881	template AnyUri p_Content_Location,	892	template AnyUri p_Content_Location,
882	charstring p_Content_Type,	893	charstring p_Content_Type,
883	template boolean p_start_,	894	template boolean p_start_,
884	template PartTypeList p_PartTypes) := {	895	template PartTypeList p_PartTypes) := {
885	Content_Location := p_Content_Location,	896	Content_Location := p_Content_Location,
886	Content_Type := p_Content_Type,	897	Content_Type := p_Content_Type,
887	start_ := p_start_,	898	start_ := p_start_,
888	PartTypes := p_PartTypes	899	PartTypes := p_PartTypes
889	}	900	}
890		901	
891	/**	902	/**
892	*	903	*
893	* @desc	904	* @desc
894	* This is a default template for a SMSTemplateType	905	* This is a default template for a SMSTemplateType
895	* @param	906	* @param
896	* p_SelectChoices the list of SelectChoice	907	* p_SelectChoices the list of SelectChoice
897	*/	908	*/
898	template SMSTemplateType m_def_SmsTemplate(909	template SMSTemplateType m_def_SmsTemplate(
899	SelectChoiceList p_SelectChoices) := {	910	SelectChoiceList p_SelectChoices) := {
900	relativePreference := omit,	911	relativePreference := omit,
901	Descriptions := omit,	912	Descriptions := omit,
902	SelectChoices := p_SelectChoices	913	SelectChoices := p_SelectChoices
903	}	914	}
904	}	915	}
905	}	916	}