



Open SDV API

ボデー/キャビン API

バージョン 202503 α

発行 2025年3月31日

変更履歴

バージョン	発行日	備考
202503 α	2025年3月31日	初版

目次

ウインドウ (Window)

ボンネット (Hood)

トランク (Trunk)

クラクション (Horn)

ワイパー (Wiping)

ミラー (Mirror)

室内ミラー (RearviewMirror)

室内ライト (Light)

ドア (Door)

シェイド (Shade)

シート (Seat)

イベント	制御操作 ※スイッチについて、制御操作を定義しない。										
	イベント通知の開始/停止	イベント通知設定 startEventNotification	ウインドウ開閉スイッチID、イベントID、開始/停止	(イベント通知設定の) 成功か失敗							
	優先度1	スイッチの正常判定		ウインドウ開閉スイッチ							
	優先度2	スイッチ入力の変化を検知		ウインドウ開閉スイッチ、スイッチ入力							
	イベントIDの割り振りはTBD										

ボンネット	サービスコール	コンフィグレーション	コンフィグ情報取得 getConfig	なし	成功か失敗 以下の情報から成る構造体を送す。(可変長配列を送す想定) ・識別名 ・IdNumber (API呼び出しに使う) ・位置 ・開閉属性 ・サポートする機能	静的な値を送すことを想定する。				識別名は以下を標準とする。 ・Front 上記以外をオプションで追加することは可能	
		状態参照	デバイス状態参照 getStatus	ボンネットID IdNumber	デバイスの情報 メイン状態とサブ状態を持つ。 mainStatus 正常状態 Normal ・停止 ・閉いている途中 ・閉じている途中 ・作動禁止 ・作動禁止 この状態から容易に復帰できることを想定 この状態から容易に復帰できることを想定 ・換気検知中 ・モータ加熱検知中 故障状態 この状態からは復帰しないことを想定 ・メカ故障 ・電気故障 ・… 故障状態にあるDTCリストを送すことを想定	稼働状態の場合 それぞれのサブ状態において、適切なフェールセーフ処置を行う。 フェールセーフ処置例 ・作動禁止 これは、ボンネットの安全機能と考えることができる。 どの稼働状態を持つかは、コンフィグ情報のサポートする機能で参照できるようにする。 故障が確定していない状態で、フェールセーフ動作している場合は、稼働状態だと考える。 この状態の場合、ボンネットが正常に動くことが期待できないため、無理に制御操作しない。			故障状態の場合 DTC (ダイアグナブルコード) の確定故障 (ConfirmedDTC) のリストを送す。		
			↑	↑	開閉度 position 開閉度割合(パーセント)					0-100 単位は%, 0で全閉、100で全開 不明な場合は、-1を送す。	
			↑	↑	目標開閉度 targetPosition 開閉度割合(パーセント)					0-100 単位は%, 0で全閉、100で全開 不明な場合は、-1を送す。	
			↑	↑	イベント通知の許可/不許可参照 eventEnable						
		制御操作	開閉開始 startMove	ボンネットID、目標開閉度 IdNumber、 TargetPosition	(制御操作の受け入れ) 成功か失敗	・開閉度と目標開閉度が一致していない場合、目標開閉度に向けてボンネットを開閉する。 ・換気検知ができない場合には、ボンネットを開める操作は許可されたアプリケーションのみしか呼び出せない			0-100 単位は%, 0で全閉、100で全開		
			開閉停止 stopMove	ボンネットID IdNumber	(制御操作の受け入れ) 成功か失敗						
		イベント通知の開始/停止	イベント通知設定 startEventNotification	ボンネットID、イベントID、開始/停止	(イベント通知設定の) 成功か失敗						
		優先度1	デバイス状態の変化を検知		ボンネットID、デバイス状態	ボンネットの開閉中に換気込みを検知すると、開閉を停止し(少しだけ戻す?)、目標開閉度を現在の開閉度に設定する					
		優先度2	開閉開始を検知		ボンネットID						
	開閉停止を検知		ボンネットID、停止理由								
	イベントIDの割り振りはTBD										

ボンネット開閉スイッチ	サービスコール	コンフィグレーション	コンフィグ情報取得 getConfig	なし	ボンネット開閉スイッチの個数だけ以下の情報から成る構造体を送す。(可変長配列を送す想定) ・識別名 ・IdNumber (API呼び出しに使う) ・位置 ・サポートする機能 (OPEN, CLOSE)					定義 ボンネットを制御する物理的なスイッチ	このAPIは、ボンネット開閉スイッチとボンネットは直結しておらず、ピークルOSが制御することを仮定する。	
		状態参照	デバイス状態参照 getStatus	ボンネット開閉スイッチID IdNumber	mainStatus 正常状態 稼働状態 ボンネット開閉スイッチでは、この状態は想定していない。 故障状態 この状態からは復帰しないことを想定 ・電気故障 故障状態にあるDTCリストを送すことを想定 スイッチの場合、短絡と断線想定する。							
			↑	↑	スイッチ入力 switchIn 「開ける」「閉じる」							
		制御操作 ※スイッチについて、制御操作を定義しない。										
		イベント通知の開始/停止	イベント通知設定 startEventNotification	ボンネット開閉スイッチID、イベントID、開始/停止	(イベント通知設定の) 成功か失敗							
		優先度1	スイッチの正常判定		ボンネット開閉スイッチ							
		優先度2	スイッチ入力の変化を検知		ボンネット開閉スイッチ、スイッチ入力							
	イベントIDの割り振りはTBD											

トランク	コンフィグレーション	コンフィグ情報取得 getConfig	なし	成功か失敗 ウインドウの個数だけ以下の情報から成る構造体を送す。(可変長配列を送す想定) ・識別名 ・IdNumber (API呼び出しに使う) ・位置 ・ライトの有無	静的な値を送すことを想定する。					識別名は以下を標準とする。 ・Front ・Rear 上記以外をオプションで追加することは可能	定義 開閉できる車室の開閉部の中で、車両の走行中は通常閉じておくもの。または、大きい荷物を出入れすることを想定しているもの。 ・開閉は一次元方向にのみ動作する
------	------------	------------------------	----	---	-----------------	--	--	--	--	--	---

サービスコール	状態参照	デバイス状態参照 getStatus	トランクID IdNumber	<p>デバイスの情報 メイン状態とサブ状態を持つ。 mainStatus 正常状態 Normal ・停止 ・開いている途中 ・閉じている途中 ・オート作動禁止 ・作動禁止 ・作動禁止</p> <p>この状態から容易に復帰できることを想定 ウインドウでは以下を想定 ・ 挟み込み検知中 ・ モータ加熱検知中 ・ 端点未学習</p> <p>故障状態 この状態からは復帰しないことを想定 ・ メカ故障 ・ 電気故障 ・ …… 故障状態にあるDTCリストを返すことを想定</p>	<p>稼働状態の場合 それぞれのサブ状態において、適切なフェールセーフ処置を行う。 フェールセーフ処置例 ・ オート作動禁止 ・ 作動禁止</p> <p>これらは、トランクの安全機能と考えることができる。 どの稼働状態を持つかは、コンフィグ情報のサポートする機能で参照できるようにする。 故障が確定していない状態で、フェールセーフ動作している場合は、稼働状態だと考える。</p> <p>故障状態の場合 この状態の場合、トランクが正常に動くことが期待できないため、無理に制御操作しない。</p>				故障状態の場合 DTC (ダイアグノスブルコード) の 確定故障 (ConfirmedDTC) のリストを返す。	
		↑	↑	<p>ロック状態参照 lockStatus ロック状態 (ロック中、ロック無) ロックされている場合、指定されている操作優先度も返す。</p>						
		↑	↑	<p>開閉度 position 開閉度割合 (パーセント)</p>				0-100 単位は%、0で全開、100で全閉 不明な場合は、-1を返す。		
		↑	↑	<p>目標開閉度 targetPosition 開閉度割合 (パーセント)</p>						
		↑	↑	<p>トランク内ライト点灯状態 lightStatus ライトON/OFF/UNKNOWN</p>	ライト無しの場合は、無効とする					
		↑	↑	<p>イベント通知の許可/不許可参照 eventEnable</p>						
	制御操作	開閉開始 startMove	トランクID、目標開閉度 IdNumber, targetPosition	(制御操作の受け入れ) 成功か失敗	<p>・ 開閉度と目標開閉度が一致していない場合、目標開閉度に向けてトランクを開閉する。 ・ 挟み込み検知ができない場合には、トランクを閉める操作は許可されたアプリケーションのみしか呼び出せない</p>	0-100 単位は%、0で全開、100で全閉				
		開閉停止 stopMove	トランクID IdNumber	(制御操作の受け入れ) 成功か失敗						
		ロック要求 lock	トランクID、操作優先度 IdNumber, priority	(制御操作の受け入れ) 成功か失敗 アクセス制御情報	操作優先度が同じロック要求を受け付けるか、失敗させるか?					
		ロック解除 unlock	トランクID IdNumber	(制御操作の受け入れ) 成功か失敗						
イベント通知の開始/停止	イベント通知設定 startEventNotification	トランクID、イベントID、開始/停止	(イベント通知設定の) 成功か失敗							
イベント	優先度1 イベントIDの割り振り TBD	デバイス状態の変化を検知		トランクID、デバイス状態	トランクの開閉中に挟み込みを検知すると、開閉を停止し (少しだけ戻す?)、目標開閉度を現在の開閉度に設定する					
		他のプログラムによるロック解除		トランクID、イベントID	より操作優先度が高いロック取得が実行された場合にこのイベントが通知される。					
	優先度2 イベントIDの割り振り TBD	開閉開始を検知 開閉停止を検知		トランクID トランクID、停止理由						
トランク開閉スイッチ	サービスコール	コンフィグレーション	コンフィグ情報取得 getConfig	なし	<p>トランク開閉スイッチの個数だけ以下の情報から成る構造体を返す。(可変長配列を返す想定) ・ 識別名 ・ IdNumber (API呼び出しに使う) ・ 位置 ・ スイッチ種別 ※フットセンサによる開閉もある ・ サポートする機能 (OPEN, CLOSE)</p>					
		状態参照	デバイス状態参照 getStatus	トランク開閉スイッチID IdNumber	<p>mainStatus 正常状態 稼働状態 トランク開閉スイッチでは、この状態は想定していない。 故障状態 この状態からは復帰しないことを想定 ・ 電気故障 故障状態にあるDTCリストを返すことを想定 スイッチの場合、短絡と断線を想定する。</p>					
		↑	↑	<p>スイッチ入力 switchIn 「開ける」「閉じる」</p>						
		制御操作 ※スイッチについて、制御操作を定義しない。								
	イベント通知の開始/停止	イベント通知設定 startEventNotification	トランク開閉スイッチID、イベントID、開始/停止	(イベント通知設定の) 成功か失敗						
	イベント	優先度1	スイッチの正常判定		トランク開閉スイッチ					
		優先度2 イベントIDの割り振り TBD	スイッチ入力の変化を検知		トランク開閉スイッチ、スイッチ入力					
クラクション	コンフィグレーション	コンフィグ情報取得 getConfig	なし	<p>成功か失敗 以下の情報から成る構造体を返す。(可変長配列を返す想定) ・ 識別名 ・ IdNumber (API呼び出しに使う) ・ 位置 ・ サポートする機能</p>	静的な値を返すことを想定する。			識別名は以下を標準とする。 ・ Front 上記以外をオプションで追加することは可能		

サービス コール	状態参照	デバイス状態参照 getStatus	クラクションID IdNumber	デバイスの情報 メイン状態とサブ状態を持つ。 mainStatus 正常状態 Normal ・吹鳴停止中 ・吹鳴中 停動状態 Suspend この状態から容易に復帰できることを想定 ・なし 故障状態 この状態からは復帰しないことを想定 ・メカ故障 ・電気故障 ・… 故障状態にあるDTCリストを返すことを想定	故障状態の場合 この状態の場合、クラクションが正常に動くことが期待できないため、無理に制御操作しない。			故障状態の場合 DTC (ダイアグノスチックコード) の 確定故障 (ConfirmedDTC) のリス トを返す。		
		↑	↑	音量 [%] volume 0-100						
		↑	↑	イベント通知の許可/不許可参照 eventEnable						
		制御操作	吹鳴開始 StartHorn	クラクションID、時間[ms] IdNumber, Time	(制御操作の受け入れ) 成功か失敗			時間は[ms]単位 鳴らし続ける場合は型の最大値 を設定する。		
			吹鳴停止 StopHorn	クラクションID IdNumber	(制御操作の受け入れ) 成功か失敗					
音量設定 SetVolume	クラクションID、音量[%] IdNumber, Volume		(制御操作の受け入れ) 成功か失敗			音量[%] 0-100 音量については法規などもあり 要検討箇所なので、一旦の値 を設定しても一定の音量となる ようにする。				
警告吹鳴 AlertHorn	クラクションID、吹鳴パターン IdNumber, Pattern		(制御操作の受け入れ) 成功か失敗	以下から成る構造体を引数で渡して吹鳴パターンで制御 盗難防止アラートなどでの仕様を想定。 ・吹鳴時間[ms] ・吹鳴停止時間[ms] ・回数	以下から成る構造体 ・吹鳴時間[ms] ・吹鳴停止時間[ms] ・回数(65535は回数無制限) 吹鳴時間および吹鳴停止時間の 設定値は上限、下限を決めてお く(具体的な値はT.B.D)。					
イベント通知の開始/停止	イベント通知設定 startEventNotification	クラクションID、イベントID、開始/停止	(イベント通知設定の) 成功か失敗							
イベ ント	優先度1	デバイス状態の変化を検知	クラクションID、デバイス状態							
	イベントIDの割り振り TBD									
	優先度2	クラクションの吹鳴開始を検知	クラクションID							
	イベントIDの割り振り TBD	クラクションの吹鳴停止を検知	クラクションID、停止理由							

サービス コール	ワイパー	コンフィグレーション	コンフィグ情報取得 getConfig	なし	成功か失敗 ワイパーの箇所だけ以下の情報から成る構造体を返す。 (可変長配列を返す想定) ・識別名 ・IdNumber (API呼び出しに使う) ・位置 ・払拭パターン ・ウォッシャー有無	静的な値を返すことを想定する。	getConfig	なし	・識別名は以下を標準とする。 ・Front ・Rear ・Mirror(オプション) ・Light (オプション) ・Camera ・払拭パターンは以下を標準とする ・SingleBlade ・Tandem ・OverWrap 上記以外をオプションで追加すること は可能	定義 自動車に付着した雨水、雪、汚れなどを払拭する装置 払拭ブレード、アーム、それを駆動させるモーター、リンク機構を差す また、洗浄液によるウォッシャーも含む 定義補足 ・COVESA VSS では、WipingはWindshieldの低位オブジェクトとして定義 しているが、OSDVIの定義は、Windshieldと独立なオブジェクトとする。こ れは、WindShieldに限らず、ヘッドライト、サイドミラー、カメラ等に装着さ れている可能性があるため	・車両によってブレードの本数や駆動方式などの払拭パターンが異なる が、払拭面を一次的に払拭するという動作は共通のためこのAPIでは 考慮しない。しかし、将来的な拡張性も踏まえてコンフィグレーション は定義しておく。 ・アプリケーションでの拡張性を担保するため、動作モードに対する ブレードの動作速度、間欠動作時間はAPIで任意の値を指定できる作り とする。 ・ワイパーの初期位置、目標位置となる端点は固定であるため、ワイ パーの目標位置はAPIで指定、参照しない作りとする
		状態参照	デバイス状態参照 getStatus	ワイパーID IdNumber	デバイスの情報 メイン状態とサブ状態を持つ。 mainStatus 正常状態 Normal ・動作停止 ・一時停止中 ・初期位置、又は目標位置へ移動中 停動状態 Suspend この状態から容易に復帰できることを想定 ワイパーでは以下を想定 ・モータ加熱検知中 ・ワイパー摩擦 ・ウォッシャー液不足 故障状態 この状態からは復帰しないことを想定 ・メカ故障 ・電気故障 ・… 故障状態にあるDTCリストを返すことを想定	稼働状態の場合 それぞれのサブ状態において、適切なフェールセーフ処置を行う。 フェールセーフ処置例 ・オート作動禁止 ・作動禁止 どの停動状態を持つかは、コンフィグ情報のサポートする機能で参照できる ようにする。 故障が確定していない状態で、フェールセーフ動作している場合は、停動状 態だと考える。 故障状態の場合 この状態の場合、ワイパーが正常に動くことが期待できないため、無理に制 御操作しない。			DTC (ダイアグノスチックコード) の 確定故障 (ConfirmedDTC) のリス トを返す。		
		↑	↑	動作速度参照 frequency 間欠動作を除く、1分間の往復回数 (例: 80・・・1分間に80回往復)							
		↑	↑	一時停止時間参照 interval 間欠動作による一時停止時間(秒)							
↑	↑	摩耗状態参照 wiperWear ワイパーの摩耗状態 (0%・・・摩耗無し, 100%・・・摩耗, 要交換)									

サービス コール	状態参照	デバイス状態参照 getStatus	ミラー操作スイッチID IdNumber	mainStatus 正常状態 停止状態 ミラー開閉スイッチでは、この状態は想定していない。 故障状態 この状態からは復帰しないことを想定 ・電気故障 故障状態にあるDTCリストを返すことを想定 スイッチの場合、短絡と断線を想定する。							
		↑	↑	スイッチ入力 switchIn ・ミラー調整スイッチ 「UP」「DOWN」「RIGHT」「LEFT」 ・ミラー折り畳みスイッチ 「ON」「OFF」 ・ミラーヒータリングスイッチ 「ON」「OFF」「LEVEL」 ・ミラー防眩スイッチ 「ON」「OFF」「LEVEL」							
		制御操作 ※スイッチについて、制御操作を定義しない。									
	イベント通知の開始/停止	イベント通知設定 startEventNotification	ミラー操作スイッチID、イベントID、開始/停止	(イベント通知設定の) 成功か失敗							
イベント	優先度1	スイッチの正常判定		ミラー操作スイッチ							
	優先度2	スイッチ入力の変化を検知		ミラー操作スイッチ、スイッチ入力							
	イベントIDの割り振りはTBD										

室内ミラー	サービス コール	コンフィグレーション	コンフィグ情報取得 getConfig	なし	成功か失敗 以下の情報から成る構造体を返す。(室内ミラーはひとつしかないことが多いが、他オブジェクトとAPIを合わせるため、可変長配列を返す想定) ・識別名 ・IdNumber (API呼び出しに使う) ・位置 ・サポートする機能 防眩対応できるか/できないか	静的な値を返すことを想定する。			識別名は以下を標準とする。 ・Front 上記以外をオプションで追加することは可能	定義 社室内から後方を確認するミラー 後方を確認する性質上、ヘッドライトに対する防眩機能を持つものが多い。	室内ミラーの防眩度は手動で2段階動かせるタイプをよく見かける。 電動で防眩度を変更できる室内ミラーは、普及していないと思われる。	
		状態参照	デバイス状態参照 getStatus	室内ミラーID IdNumber	デバイスの情報	メイン状態とサブ状態を持つ。 mainStatus 正常状態 Normal ・停止 ・防眩度変更中 停止状態 Suspend 室内ミラーでは停止状態を想定しない。 故障状態 この状態からは復帰しないことを想定 ・メカ故障 ・電気故障 ・・・ 故障状態にあるDTCリストを返すことを想定	故障状態の場合 この状態の場合、ウィンドウが正常に動くことが期待できないため、無理に制御操作しない。					
					↑	↑	防眩度 dimmingLevel 防眩度割合 (0%: OFF~ 100%: 最大防眩)	防眩度(パーセント)				
					↑	↑	目標防眩度 targetDimmingLevel (0%: OFF~ 100%: 最大防眩)	防眩度(パーセント)				
					↑	↑	イベント通知の許可/不許可参照 eventEnable					
		制御操作	防眩度変更開始 startMove	室内ミラーID、目標防眩度 IdNumber、targetDimmingLevel	(制御操作の受け入れ) 成功か失敗	・防眩度と目標防眩度が一致していない場合、目標防眩度に向けて室内ミラーを操作する。						
			防眩度変更停止 stopMove	室内ミラーID IdNumber	(制御操作の受け入れ) 成功か失敗							
		イベント通知の開始/停止	イベント通知設定 startEventNotification	室内ミラーID、イベントID、開始/停止	(イベント通知設定の) 成功か失敗							
		イベント	優先度1	デバイス状態の変化を検知		室内ミラーID、デバイス状態	故障状態を検知した場合、このイベントを発生させる。					
			優先度2	防眩度変更開始を検知		室内ミラーID						
イベントIDの割り振りはTBD	防眩度変更停止を検知			室内ミラーID、停止理由								

室内ミラー防眩度変更スイッチ	サービス コール	コンフィグレーション	コンフィグ情報取得 getConfig	なし	室内ミラー防眩度変更スイッチの個数だけ以下の情報から成る構造体を返す。(可変長配列を返す想定) ・識別名 ・IdNumber (API呼び出しに使う) ・位置 ・サポートする機能 ON/OFFの2Wayなのか3Wayなのか					定義 室内ミラーの防眩度を変更する物理的なスイッチ 補足 COVESA VSS では、Switchの定義はない。	このAPIは、スイッチと室内ミラーは直結しておらず、ビークルOSが制御することを仮定する。
		状態参照	デバイス状態参照 getStatus	室内ミラー防眩度変更スイッチID IdNumber	mainStatus 正常状態 停止状態 ミラー開閉スイッチでは、この状態は想定していない。 故障状態 この状態からは復帰しないことを想定 ・電気故障 故障状態にあるDTCリストを返すことを想定 スイッチの場合、短絡と断線を想定する。						
					↑	↑	スイッチ入力 switchIn 「ON」「OFF」「LEVEL」				
		制御操作 ※スイッチについて、制御操作を定義しない。									
イベント通知の開始/停止	イベント通知設定 startEventNotification	室内ミラー防眩度変更スイッチID、イベントID、開始/停止	(イベント通知設定の) 成功か失敗								

サ ー ビ ス コ ー ル	状態参照	デバイス状態参照 getStatus	ドア開閉スイッチID IdNumber	mainStatus 正常状態 停止状態 チャイルドロックスイッチでは、この状態は想定していない。 故障状態 この状態からは復帰しないことを想定 ・電気故障 故障状態にあるDTCリストを返すことを想定 スイッチの場合、短絡と断線を想定する。							
		1	1	スイッチ入力 switchIn 「チャイルドロックON」「チャイルドロックOFF」							
	制御操作 ※スイッチについて、制御操作を定義しない。										
	イベント通知の開始/停止	イベント通知設定 startEventNotification	チャイルドロックスイッチID、イベントID、開始/停止	(イベント通知設定の) 成功か失敗							
イ ベ ン ト	優先度1	スイッチの正常判定		チャイルドロックスイッチ							
	優先度2	スイッチ入力の変化を検知		チャイルドロックスイッチ、スイッチ入力							
	イベントIDの割り振りはTBD										

シェイド	サ ー ビ ス コ ー ル	コンフィグレーション	コンフィグ情報取得 getConfig	なし	成功か失敗 以下の情報から成る構造体を返す。(可変長配列を返す想定) ・識別名 ・IdNumber (API呼び出しに使う) ・位置 ・閉閉属性 ・サポートする機能 静的な値を返すことを想定する。					標準として用意する識別名は特に定め ない。 ただし、ウインドウやドアの付属の シェイドは、そのウインドウやドア の識別子と同一とする。	
		状態参照	デバイス状態参照 getStatus	シェイドID IdNumber	デバイスの情報	メイン状態とサブ状態を持つ。 mainStatus 正常状態 Normal ・停止 ・開いている途中 ・閉じている途中 停止状態 Suspend この状態から容易に復帰できることを想定 ・挟み込み検知中 ・モータ加熱検知中 ・端点未学習 故障状態 この状態からは復帰しないことを想定 ・メカ故障 ・電気故障 ・・・ 故障状態にあるDTCリストを返すことを想定	稼働状態の場合 それぞれのサブ状態において、適切なフェールセーフ処置を行う。 フェールセーフ処置例 ・オート作動禁止 ・作動禁止 これらは、シェイドの安全機能と考えることができる。 どの稼働状態を持つかは、コンフィグ情報のサポートする機能で参照できる ようにする。 故障が確定していない状態で、フェールセーフ動作している場合は、稼働状 態だと考える。 故障状態の場合、シェイドが正常に動くことが期待できないため、無理に制 御操作しない。				
					1	1	ロック状態参照 lockStatus ロック状態 (ロック中、ロック無) ロックされている場合、指定されている操作優先度 も返す。				
					1	1	開閉度 position 開閉度割合(パーセント)			0-100 単位は%、0で全閉、100で全 開 不明な場合は、-1を返す。	
					1	1	目標開閉度 targetPosition 開閉度割合(パーセント)			0-100 単位は%、0で全閉、100で全 開 不明な場合は、-1を返す。	
					1	1	イベント通知の許可/不許可参照 eventEnable				
		制御操作	開閉開始 startMove	シェイドID、目標開閉度 IdNumber, targetPosition	(制御操作の受け入れ) 成功か失敗	・開閉度と目標開閉度が一致していない場合、目標開閉度に向けてシェイド を開閉する。 ・挟み込み検知ができない場合には、シェイドを閉める操作は許可されたア プリケーションのみしか呼び出せない					
			開閉停止 stopMove	シェイドID IdNumber	(制御操作の受け入れ) 成功か失敗						
		イベント通知の開始/停止	イベント通知設定 startEventNotification	シェイドID、イベントID、開始/停止	(イベント通知設定の) 成功か失敗						
		イ ベ ン ト	優先度1	デバイス状態の変化を検知		シェイドID、デバイス状態	シェイドの開閉中に挟み込みを検知すると、開閉を停止し(少しだけ戻 す?)、目標開閉度を現在の開閉度に設定する				
優先度2	開閉開始を検知			シェイドID							
		イベントIDの割り振りはTBD									
		開閉停止を検知		シェイドID、停止理由							

シェイド開閉スイッチ	サ ー ビ ス コ ー ル	コンフィグレーション	コンフィグ情報取得 getConfig	なし	シェイド開閉スイッチの個数だけ以下の情報から成 る構造体を返す。(可変長配列を返す想定) ・識別名 ・IdNumber (API呼び出しに使う) ・位置 ・サポートする機能 (UP側のみ、DONW側のみ、 2段階スイッチ)					定義 シェイドを制御する物理的なスイッチ
		状態参照	デバイス状態参照 getStatus	シェイド開閉スイッチID IdNumber	mainStatus 正常状態 停止状態 ミラー開閉スイッチでは、この状態は想定してい ない。 故障状態 この状態からは復帰しないことを想定 ・電気故障 故障状態にあるDTCリストを返すことを想定 スイッチの場合、短絡と断線を想定する。					このAPIは、シェイド開閉スイッチとシェイドは連携しておらず、ビー クルOSが制御することを仮定する。
1	1				スイッチ入力 switchIn 「操作なし」「開1段階目」「開2段階目」「閉1段 階目」「閉2段階目」					

ト イ ン ク	制御操作 ※スイッチについて、制御操作を定義しない。										
	イベント通知の開始/停止	イベント通知設定 startEventNotification	シェイド開閉スイッチID、イベントID、開始/停止	(イベント通知設定の) 成功か失敗							
	優先度1	スイッチの正常判定		シェイド開閉スイッチ							
	優先度2	スイッチ入力の変化を検知		シェイド開閉スイッチ、スイッチ入力							
	イベントIDの割り振り TBD										

シ ー ト	サ ー ビ ス コ ー ル	コンフィグレーション	コンフィグ情報取得 getConfig	なし	成功か失敗 シートの数だけ以下の情報から成る構造体返す。 (可変長配列を返す想定) ・識別名 ・IdNumber (API呼び出しに使う) ・位置 ・サポートする機能 どの場所が可変なのかを示す。 マッサージ機能有無 シート温める機能有無 シート冷やす機能有無	静的な値を返すことを想定する。				識別名は以下を標準とする。 ・FrontRight ・FrontMiddle ※一般的ではない ・FrontLeft ・RearRight ・RearMiddle ・RearLeft 上記以外をオプションで追加することは可能	定義 人が座るための場所、複数の部位に分かれている。	・COVESA VSS では、Occupant をメンバーに持つが、VDE + U モデルの Driver の要素とする。 ・COVESA VSS では、マッサージ機能は一次元で強さを指定している。		
		状態参照	デバイス状態参照 getStatus	シートID idNumber	デバイスの情報 メイン状態とサブ状態を持つ。 mainStatus 正常状態 Normal ・停止 ・開いている途中 ・閉じている途中 ・オート作動禁止 ・作動禁止 この状態から容易に復帰できることを想定 ・挟み込み検知中 ・モータ加熱検知中 ・端点未学習 故障状態 この状態からは復帰しないことを想定 ・メカ故障 ・電気故障 ・・・ 故障状態にあるDTCリストを返すことを想定	稼働状態の場合 それぞれのサブ状態において、適切なフェールセーフ始動を行う。 フェールセーフ始動例 ・オート作動禁止 ・作動禁止 これらは、シートの安全機能と考えることができる。 どの稼働状態を持つかは、コンフィグ情報のサポートする機能で参照できるようにする。 故障が確定していない状態で、フェールセーフ動作している場合は、稼働状態だと考える。 故障状態の場合、ドアが正常に動くことが期待できないため、無理に制御操作しない。			故障状態の場合 DTC (ダイアグナブルコード) の確定故障 (ConfirmedDTC) のリストを返す。					
					着座状況 isOccupied 人が座っているかいないかを返す。						boolean 座っている: true 座っていない: false			
					シートベルト状況 isBelted シートベルトをしているかしていないかを返す。						boolean シートベルトをしている: true シートベルトをしていない: false			
					シートベルト高さ seatBeltHeight シートベルトの高さ。						単位mm 0は、車両として最も低い位置			
					マッサージ機能強さ massageLevel マッサージの強さ						単位% 0-100 0でOFF、100で最大レベル			
					シート温度を返す。 heatingCooling						% -100から100 -100は、最大冷却 +100は、最大暖房			
					シート位置 position 以下の要素をメンバーに持つ構造体									
					position 前後位置を返す。 ※position.positionとなる。						前後位置はmm 0で最も前の位置			
					height 高さを返す。						高さはmm 0で最も低い位置			
					tile シート角度を返す。	COVESA VSSでは、Tileを変更しても、座面 (seating) と背もたれ (Backrest) の関係は変わらないとしている。					シート角度は度、0でシートが最も倒れていない方向の位置			
					recline 背もたれ角度を返す。						シート角度は度、0でシートが最も倒れていない方向の位置			
					lumbarSupport ランバーサポートの左右 (外内) を返す。						単位% 0-100 0で最も内側、100で最も外側			
					lumbarHeight ランバーサポートの高さを返す。						単位mm 0は、最も低い位置			
					sideBolesterSupport サイドボルスターのサポート度合いを返す。						単位% 0-100 0で最もサポートが弱い (外側に開く)、100で最もサポートが強い (狭い)。			
					seatingLength 座面の前後位置を返す。						単位mm 0は、最も後ろ位置			
					headrestHeight ヘッドレストの高さを返す。						単位mm 0は、最も低い位置			
					headrestAngle ヘッドレストの角度を返す。						度 0で背もたれと並行、正で前に倒れる角度			
					目標シート位置 targetPosition targetPositionと同じ要素をメンバーに持つ構造体返す。						% -100から100 -100は、最大冷却 +100は、最大暖房			
					イベント通知の許可/不許可参照 eventEnable									
	制御操作	シートベルト高さ変更 changeSeatBeltHeight	シートID、目標シートベルト高さ idNumber、targetSeatBeltHeight	(制御操作の受け入れ) 成功か失敗	・現状の値と指定された目標の値が一致していない場合、目標の値に向けてシートベルト高さを変える。				単位mm 0は、車両として最も低い位置					
		マッサージ機能強さ変更 changeMassageLevel	シートID、目標マッサージ強度 idNumber、targetMassageLevel	(制御操作の受け入れ) 成功か失敗	・現状の値と指定された目標の値が一致していない場合、目標の値に向けてマッサージ機能強さを変える。				単位% 0-100 0でOFF、100で最大レベル					
		シートベルト温度変更 changeHeatingCooling	シートID、目標シートベルト温度 idNumber、targetHeatingCooling	(制御操作の受け入れ) 成功か失敗	・現状の値と指定された目標の値が一致していない場合、目標の値に向けてシート温度を変え。				% -100から100 -100は、最大冷却 +100は、最大暖房					
		シート位置変更 changeTargetPosition	シートID、目標高さ idNumber、targetPosition targetPositionは位置を表す構造体返す。 getStatusを参照	(制御操作の受け入れ) 成功か失敗	・現状の値と指定された目標の値が一致していない場合、目標の値に向けてシートを動作する。 ・挟み込み検知ができない場合には、動作する操作は許可されたアプリケーションのみしか呼び出せない									

