

# USDM(サーバプッシュ仕様 / Server Push Spec) v3.00

30 分前 に Atsushi Yoshida が作成

## 変更履歴 / Change history

日付 Date	Ver	チケットID Ticket ID	変更箇所 Changes	変更内容 Description	修正理由 Reason	変更者 Author
2022/5/31	0.90	－	初版	－	－	Yoshida
2022/6/30	1.00	－	－	21MMの仕様をUSDM形式に変換。 Converted 21MM specifications to USDM format.	要求と仕様の目的(理由)を明確にするため。 To clarify the purpose (reason) of the requirement and specification.	Yoshida
2022/6/30	1.00	－	Push_010_030	protoファイルの変更。 Change proto file.	ACKを返す仕様を入れたため。 Because we put in a specification that returns ACK.	Yoshida
2022/6/30	1.00	－	Push_020_010	サーバプッシュ機能配置を追記。 Add server push function placement.	システム構成が21MMと変わるため。(CECUの新設) Because the system configuration changes from 21MM. (Addition of CECU)	Yoshida
2022/6/30	1.00	－	Push_040_010	HU内部でデータをロストさせない仕様を追加。 Added a specification that does not cause data to be lost inside HU.	21MMではHU内部でデータロストするケースがあったため。 In 21MM, there was a case where data was lost inside HU.	Yoshida
2022/6/30	1.00	－	Push_040_020	ACKを返す仕様を追加。 Added specification to return ACK.	21MMではHU内部でデータロストするケースがあったため。 In 21MM, there was a case where data was lost inside HU.	Yoshida
2022/6/30	1.00	－	Push_050_020 Table 4	リトライ仕様を変更。 Changed retry specifications.	HUCOM仕様に従い500番台相当のレスポンスコードのみリトライするため。 To retry only the response code equivalent to the status code in the 500s according to the HUCOM specifications.	Yoshida
2022/6/30	1.00	－	Push_060_010 Push_060_020	ACC-ON中のみ起動することを明記。 Clarified that the session is connected only during ACC-ON.	ACC-OFF時など、画面が黒画の時プッシュデータを受けてもユーザに通知(画面表示)できないため。 When the screen is black, such as when ACC-OFF, the user cannot be notified (screen display) even if push data is received.	Yoshida
2022/6/30	1.00	－	Push_090_010	機能情報値を追加。 Add function information value.	HUCOMの機能情報アップロード機能にて、センタにアップロードするデータの定義が必要なため。 Because it is necessary to define the data to be uploaded to the center in the uploading function of HUCOM.	Yoshida
2022/6/30	1.00	－	Table 3	サーバプッシュ利用機能の追加/削除。 Addition / deletion of server push usage function.	24MMでサーバプッシュを利用する機能のIDのみに仕様を修正したため。 Because the table was updated to the function to use server push in 24MM.	Yoshida
2022/6/30	1.00	－	Table 5	PING送信間隔の変更。 Change the PING transmission interval.	DCMのNATテーブル有効期限が短くなったため。 The deadline for the DCM's NAT table has been shortened.	Yoshida
2022/8/12	1.10	DC24-3099	Push_010_030 Push_040_020 Push_040_030 sequence03-01 sequence03-02 sequence04-01 sequence04-02 Table 2	ACKに以下を追加。 ・スタック成功 ・プッシュ通知失敗 ・プッシュ対象の機能がない Add the following to ACK. ・Stuck success ・Push notification failed ・There is no push target function	プッシュサーバにあるスタックを削除する/しないの判断に利用するため。 For use in determining whether or not to delete stacks on the push server.	Yoshida
2022/8/12	1.10	DC24-3300	Tale 2	sequence_idをfixed64→stringへ変更 Change sequence_id from fixed64 to string	ユニークなsequence_id(UUID)を利用するため、桁数を増やした。 Increased the number of digits to use unique sequence_id (UUID).	Yoshida
2022/9/30	1.50	DCQA24-561	Table 3	Connected_Audioにsub_idを追加。 Added Sub_id to "Connected_Audio".	MediaPOから追加依頼があったため。 Because there was an additional request from the Media PO.	Yoshida
2022/9/30	1.50	DC24-9352	Table 3	App_Logのapp_idを追加 Added "App_Log".  Camera_Image_Dataにsub_idを追加。 Added Sub_id to "Connected_Audio".  以下IDを削除 ・Remote_Monitoring ・Drive_Recorder ・Camera_Image_Dataのsub_id=3 Delete the following ID ・Remote_monitoring ・Drive_recorder ・sub_id = 03h in Camera_image_data	カメラPO、MediaPOから追加依頼があったため。 Because there was an additional request from the Camera PO and Media PO.	Yoshida
2022/9/30	1.50	DC24-9661	Push_040_030	スタックデータの削除タイミングを明確化。 Clarified when to delete stack data.	スタック期間がACC-ONから一定時間なのか、プッシュデータを受信してから一定時間なのかがあいまいだったため。 Because the description of the start timing of the stack was ambiguous.	Yoshida

日付 Date	Ver	チケットID Ticket ID	変更箇所 Changes	変更内容 Description	修正理由 Reason	変更者 Author
2022/9/30	1.50	－	Push_050_010	備考欄にサービス提供条件を追記。 Add the service provision conditions in the remarks column.	日本のプロビジョニングPOから要望があったため。 We received a request from a provisioning PO in Japan to describe service provision conditions in the specifications.	Yoshida
2022/12/16	1.60	DC24-12735	Table 3	sub_idの追加。 Add sub_id.	21MMと同じ用途でsub_idが必要だったため。 21MMと24MMで同じ用途でもIDが異なると誤解を招く可能性があるため、21MMとIDを揃えた。 Because User profile needed sub_id for the same use as 21MM. If the ID is different for 21MM and 24MM, it may lead to misunderstanding, so the ID is aligned with 21MM.	Yoshida
2022/12/16	1.60	DC24-12735	Table 2	expiresのDescriptionを修正。 Fixed description of expires.	DATEの参照仕様書が間違っていたため。 Because the specification to which it refers is wrong.	Yoshida
2022/12/16	1.60	DC24-12735	Push_010_030	push.protoを修正。	STACK = 0の後に「;」を追加。(記載漏れ) Added ";" after STACK=0. (Forgot to write)	Yoshida
2023/3/17	1.70	DC24-15073	Table 3	DAのsub_id=01hにValueと説明を追記。 Added Value and description to sub_id=01h of DA.	記載が漏れていたため。 Because it was missing a description.	Yoshida
2023/6/2	3.00	WNSDK24-4669	Table 3	ナビのapp_id、sub_idを追加。 Add app_id and sub_id for navi function.	ナビのPO情報を、スマホから操作するユースケースが増えたため。 Because a use case was added to operate the POI information of the navigation from the smartphone.	Yoshida
2023/6/2	3.00	DC24-16511	Table 3	リモートウォーニングのapp_id、sub_idを追加。 Add app_id and sub_id for remote warning function.	19PFv3対応に伴い、リモートウォーニング機能のユースケースが増えたため。 Because the use cases of the remote warning function increased with 19PFv3 support.	Yoshida
2023/6/2	3.00	DC24-18191	Table 3	OTAのsub_idを追加。 Add and sub_id for OTA function.	19PFv3対応に伴い、OTA機能のユースケースが増えたため。 Because the use cases of the OTA function increased with 19PFv3 support.	Yoshida
2023/6/2	3.00	CCB-216	Table 3	VAのsub_idを追加。 Add and sub_id for VA function.	VAのサーバプッシュのユースケースが増えたため。 Due to the increased use case of server push for VA.	Yoshida
2023/6/2	3.00	DC24-18204	Table 3	Camera Image Dataのsub_id=02h, 04hを削除。 Delete sub_id=02h, 04h of Camera Image Data.	Cameraのサーバプッシュのユースケースが減ったため。 Due to reduced server push use cases for camera.	Yoshida

## はじめに / Introduction

サーバプッシュ機能に関するUSDMを記載する。  
本仕様書で定義するサーバプッシュ機能は、以下の特徴を持つ。

- プッシュサーバは、サーバプッシュを利用するアプリケーション／サービスからの依頼を受けて、HUにプッシュ通知を配信する。
- プッシュサーバからHUへの通知は、gRPCを利用して行う。

Specifies USDM for Sserver push.  
Regarding a functionality of this server push, it has features like below.

- Push Server receives REQUESTs from App Servers, then Push Server transmits NOTIFICATIONS to HU.
- Push Server transmits notifications to HU via gRPC.

## USDM

LANG	ID	要求	理由	ID	仕様	理由	備考	Diff from 21MM
jp	Push_010	センタから車載機にイベント通知したい	センタトリガの通信を行う機能があるため	010	プッシュセッションは一つの機能だけが張り、各機能にプッシュデータを配布すること。	プッシュを利用したい機能全てがセッションを維持してしまうと、センタのリソースを膨大に消費してしまうため。		21MM同等
en	Push_010	The server wants to notify the MM of the event from the server.	Because there is a function to communicate with the server trigger.	010	A push session has only one function, and push data is distributed to each function.	If all the functions maintain the session, it consumes a huge amount of server resources.		Carry from 21MM
jp	Push_010			020	プッシュ機能の構成は図1：システム構成図を参照。	21MMを踏襲しているため。 HUCOM_010_010を実現するために、プッシュ配信を1つのHU機能、HUサーバで担うため。		21MM同等
en	Push_010			020	For the configuration of the push function, refer to "Fig. 1: System configuration diagram".	To realize HUCOM_010_010.		Carry from 21MM
jp	Push_010			030	サーバプッシュ機能のプロトコルはgRPCを利用すること。 protoファイルは以下を参照。▲1.60 protoファイルの各パラメータの説明は表2参照。▲1.10	双方向通信が可能であるため。 21MMから利用しているプロトコルのため、開発規模を小さくできるかつ、品質が安定しているため。		21MM改善

LANG	ID	要求	理由	ID	仕様	理由	備考	Diff from 21MM
					<div> push.proto</div>			
en	Push_010			030	Use gRPC as the protocol for the server push function. Refer to above file for the proto file. Refer to Table 2 for a description of each parameter in the proto file. ▲ 1.10	Because two-way communication is possible. Because the protocol has been used since 21MM, the scale of development can be reduced and the quality is stable.		Change
jp	Push_020	H/U終端でサーバプッシュを行いたい	21MMの資産を流用できるため、改修規模も小さく、品質が安定しているため。 DCMのサーバプッシュを利用する場合、CECU経由でデータを受信することになり、レスポンス遅延が発生するため。	010	HU機能のサーバプッシュは、車載機の構成に関わらず、HUでセッションを張ること。 詳細は「表1：サーバプッシュ機能配置」参照。	Push_020と同じ理由		21MM同等
en	Push_020	HU want to have a server push function.	Because the 21MM construction can be diverted, the scale of repair is small and the quality is stable. When using DCM server push, data will be received via CECU, which causes a response delay.	010	The server push of the HU function is to set up a session with HU regardless of the configuration of the in-vehicle device.	Same reason as Push_020.		Carry from 21MM
jp	Push_030	サーバからHU早速ぐにデータを配信してほしい	UX観点から、サーバからのデータをすぐに反映したいため	010	サーバのプッシュ予約から配信完了までの時間を以下とすること <ul style="list-style-type: none"><li>・ Typical：10～19秒</li><li>・ Maximum：1分</li></ul>	サーバプッシュを利用する機能担にヒアリングを行い、1分以内であれば許容できると回答があったため。 21MMの実績から、仕様を踏襲したため。		21MM同等
en	Push_030	TMC wants the data to be delivered immediately from the server to HU	Because TMC want to reflect the data from the server immediately from the UX point of view.	010	A duration time from REQUEST to REACH-COMPLETION is as below： <ul style="list-style-type: none"><li>・ Typical：10 - 19 seconds</li><li>・ Maximum：60 seconds</li></ul>	Because it followed the specifications from the track record of 21MM. We interviewed the person in charge of the function that uses server push, and responded that it was acceptable within 1 minute.		Carry from 21MM
jp	Push_040	HU内部でデータをロストしないようにしてほしい	21MMではHU内部でデータロストするケースがあったため。 データをロストしたことにサーバは気づけないため。	010	サーバプッシュを利用する各機能は、HUCOMからプッシュデータを受け取った際に、データをロストしない作りにすること。 例) プッシュデータを受け取れる状態になってからListener登録をする、受け取れない状態になったらListener登録を解除する。	Push_040と同じ理由。 ロストさせない方法は一つではないため、設計の拡張性を考慮し、具体的な方法は仕様で定義しない。		21MM改善
en	Push_040	Don't lose data in HU	In 21MM, there was a case where data was lost inside HU.	010	Each function that uses server push should be designed so that the data will not be lost when the push data is received from HUCOM. e.g.) Register as a Listener after the function are ready to receive push data. Cancel the Listener registration when it becomes impossible to receive.	Same reason as Push_040. Since there is more than one method to prevent loss, the specific method is not defined in the specifications in consideration of design extensibility.		Change
jp	Push_040			020	プッシュデータを該当機能に配布後、プッシュサーバにgRPCレイヤでACKを返すこと。 以下がわかるACKを返却すること。 <ul style="list-style-type: none"><li>・ スタック成功：Stuck because there was no push destination</li><li>・ プッシュ通知成功：対象機能へプッシュ通知配布完了</li><li>・ プッシュ通知失敗：対象機能へプッシュを通知したが、エラーが返ってきた</li><li>・ プッシュ対象の機能がない：プッシュ通知ができなかった(通知先がなかった)</li></ul>	Push_040と同じ理由。 21MMでは各機能にACK返却を任せていたが、ACK実装されていない機能もあり、問題が発生したため。 プッシュサーバにあるスタックを削除する/しないの判断に利用するため、左記のACKが必要。 <ul style="list-style-type: none"><li>・ スタック成功：機能サーバへプッシュがHUに届いたが、機能にまだ届いていないことを通</li></ul>		21MM改善

LANG	ID	要求	理由	ID	仕様	理由	備考	Diff from 21MM
					詳細はSeq03、Sec04を参照すること。 ▲1.10	知するため ・ブッシュ通知成功：機能サーバへブッシュが対象機能に届いたことを知らせるため ・ブッシュ通知失敗：機能サーバへブッシュ通知が失敗したことを知らせるため ・ブッシュ対象の機能が無い：機能サーバへ対象機能が無いことを知らせるため ▲1.10		
en	Push_040			020	After distributing the push data to the corresponding function, return ACK to the push server at the gRPC layer. Return an ACK that understands the following. ・ Stuck success: HU got stuck because there was no feature to push. ・ Push notification successful: Push notification to target function completed ・ Push notification failed: A push notification was sent to the target function, but an error was returned ・ There is no push target function: push notification was not possible (there was no notification destination) Refer to Seq03 and Seq04 for details : ▲1.10	Same reason as Push_040. In 21MM, ACK was left to each function, but there was a problem because some functions did not implement ACK. The ACK on the left is required because it is used to determine whether or not to delete the stack on the push server. ・ Stuck success: To notify the function server that the push has reached the HU, but has not yet reached the function. ・ Push notification success: To notify the function server that the push has reached the target function ・ Push notification failure: To notify the function server that the push notification has failed ・ No feature to push: To inform the feature server that there is no target feature ▲1.10		Change
jp	Push_040 ▲1.10			030	ブッシュデータを機能へ通知できない場合、ブッシュデータをスタックしておくこと。 同時にスタックするデータの数50個までとすること。 スタックしたデータは以下のタイミングで削除すること。 ・ブッシュデータを受信してから一定時間経過後 ▲1.50 (スタック期限は各機能の起動時間に合わせて調整すること。※) ・ ACC-OFF時 ▲1.10	ACC-OFF→ON時などサーバブッシュ機能よりも起動が遅い機能への通知をロストさせないため。 1分あればブッシュ利用機能は起動完了できると思われるため、スタック期限の目安を1分としている。 サーバにもデータはスタックされているため、メモリ削減の観点からACC-OFFでスタックデータを削除する。 21MMのDTENモデルでスタックは50個で実装しており、問題はなかったため。 ▲1.10	※スタック期限の目安は1分とする。	
en	Push_040 ▲1.10			030	If the push data cannot be notified to the function, stack the push data. Up to 50 pieces of data can be stacked at the same time. Stacked data should be deleted at the following timing. ・ After a certain period of time has passed since the push data was received ▲1.50 (Adjust the stack time limit according to the activation time of each function. *) ・ ACC-OFF. ▲1.10	In order not to lose notifications to functions that start later than the server push function, such as when ACC-OFF→ON. Since it seems that the push usage function can be completed in 1 minute, the stack deadline is set to 1 minute. Since data is also stacked on the server, the stack data is deleted with ACC-OFF from the viewpoint of memory reduction. Since data is also stacked on the server, the stack data is deleted with ACC-OFF from the viewpoint of memory reduction. This is because the 21MM DTEN model is implemented with 50 stacks and no problems have occurred. ▲1.10		

LANG	ID	要求	理由	ID	仕様	理由	備考	Diff from 21MM
jp	Push_050	サーバに負荷をかけたくない	過去負荷によってサーバが落ちたことがあるため	010	サービスフラグによるON/OFF制御ができること。 サービスフラグは"serverpush"を利用すること。 サービスフラグの初期値はOFFとすること。 サーバプッシュセッション開設中にON状態以外に変化した場合はすみやかにサーバプッシュセッションを破棄すること。 サービスフラグの値ごとのHU動作は以下とすること。 ON：サーバプッシュのセッションを張る OFF/NONE：サーバプッシュのセッションを切る	サーバの負荷が高い場合などはプッシュセッションを張ってほしくないため。	日本のプロビジョニングPOより、サービス提供条件を仕様書に記載する要望を受けた。 そのため、以下にサービス提供条件を記す。 ・ユーザ契約(テレマフラグON)をトリガにONとする。 ・基本的にはユーザ契約中(テレマフラグON時)は常にONだが、サーバ負荷が高い場合などに、サーバ負荷を下げるためOFFとする。▲1.50	21MM同等
en	Push_050	TMC don't want to overload the server	Because the server has been down due to the load.	010	This server push feature is on a service flag, named as "serverpush". The initial value of the service flag should be OFF. When this flag leaves "ON" after session established, the session will be closed promptly. The HU operation for each service flag value shall be as below. ON: Connect the server push session OFF/NONE: Disconnect the server push session	Server don't want HU to have a push session when the server load is high.	We received a request from a provisioning PO in Japan to describe service provision conditions in the specifications. Therefore, the service provision conditions are described below. ・ Turn on the user contract (Telematics flag = ON) as a trigger. ・ Basically, it is always ON during a user contract (when the Telema flag=ON), but when the server load is high, etc., it is turned OFF to reduce the server load. ▲1.50	Carry from 21MM
jp	Push_050			020	レスポンスコード0以外を受けた場合、「表4：レスポンスコードとリトライ要否」に従い、リトライを実施すること。 リトライ間隔とリトライ頻度は1分×3回とすること。 gRPC 上のアプリケーションは"Status.statuscodes" を利用すること。 gRPCでは、サーバ、クライアント双方の利用を想定しているが、本項ではサーバが用いた場合のクライアント（HU）の動作のみ定義する。 リトライの例は図3参照。	gRPC は特有のステータスコードを持っており、通常は200 OKのみをレスポンスコードとして用いているため。 復帰の期待できない無意味なリトライを避けるため、HUCOM仕様書に従い500番台のステータスコード相当のレスポンスコードのみリトライを実施する。		21MM改善
en	Push_050			020	If HU receive a response code other than 0, perform a retry according to "Table 4: Response code and retry". The retry interval and retry frequency should be 1 minute × 3 times. Push Service on HU treats status codes, those are "Status.statuscodes", apart from HTTP response. This section defines how to transmit codes from Push Server to HU, and how HU reacts it. Refer to Figure 3 for an example of a retry.	Because gRPC has a unique status code, usually only 200 OK is used as the response code. In order to avoid meaningless retries that cannot be expected to return, only the response code equivalent to the status code in the 500s is retried according to the HUCOM specifications.		Change
jp	Push_050			030	プッシュセッションが切れた場合、「図2：接続状態遷移図」に従い再接続を行うこと。  接続状態遷移図の注釈を以下に記す。 *1：NG応答は、grpcのStatus codeが0以外の場合(詳細は「4.Status codes」)と、サーバからPushServerStatus.statuscodes=BUSY が返答された場合を指す。 Wi-Fi途絶→復帰、DCM電波強度圏内→圏外が起こっても即座に動作せず、grpcが異常を検知してStatus codeの0以外を返してから動作すること。 *2：「ベアラの接続状態不良→良好」は以下のようなイベントを指す。 リトライ満了後は、変化イベント発生時にのみ状態遷移する。 ・ベアラ未接続→DCM接続 ・ベアラ未接続→Wi-Fi接続 ・DCM電波強度圏外→圏内	プッシュ通知を受信できる確立を増やすため、プッシュセッションは張り続けなければならない。 PING応答待ちの場合、最大3分ほど再接続に時間がかかってしまう。 そのため、ベアラ切り替えを契機に再接続を実施する。  再接続の契機が多いため、リトライ回数は抑えてサーバ負荷を減らしている。  *1の仕様：Wi-Fi途絶→復帰、DCM電波強度圏内→圏外は頻繁に切り替わる可能性があるため。 *2の仕様：無限リトライを防ぐため。		21MM同等



LANG	ID	要求	理由	ID	仕様	理由	備考	Diff from 21MM
					<p>*3：「ベアラ切り替え」が起こった場合は、即時にリトライ回数をクリアして、grpc切断と接続を行う。</p> <p>ベアラの切り替えはDCM⇔Wi-Fiの切り替えのようにベアラの種類が変わるものだけでなく、Wi-Fiルータの切り替えも含む。</p> <p>*4：以下の条件を「ベアラ接続状態良好」と判断する。</p> <ul style="list-style-type: none"><li>・ベアラがDCM &amp;&amp; DCM接続 &amp;&amp; DCM電波強度圏内</li><li>・ベアラがWi-Fi &amp;&amp; Wi-Fi接続</li></ul> <p>再接続の例は図4、5を参照すること。</p>	<p>*3の仕様：車載機とサーバ間のソケット再生成が必要なため。</p>		
en	Push_050			030	<p>If the push session is disconnected, reconnect according to "Fig. 2: Connection state transition diagram".</p> <p>The notes in Figure 2 are given below.</p> <p>*1：The NG response indicates a case where the Status code of the grpc is other than 0 and a case where a PushServerStatus.statuscodes=Busy is returned from the server.</p> <p>Wi-Fi disruption → return, DCM area → outside the area may switch frequently, so if they occur, it will not operate immediately, grpc will detect an abnormality and operate after returning Status code other than 0 .</p> <p>*2：Bad bearer connection status → good refers to the following events. In order to prevent infinite retry, after the retry expires, state transition is made only when a change event occurs.</p> <ul style="list-style-type: none"><li>・Bearer not connected → DCM connect</li><li>・Bearer not connected → Wi-Fi connect</li><li>・Outside the DCM signal strength range → DCM singal strength range</li></ul> <p>*3：When the bearer switching occurs, it is necessary to regenerate the socket between the HU and the server. Therefore, the number of retries is immediately cleared, and the grpc disconnection and connection are performed. Switching of bearers, includes not only switching of bearer types like DCM → Wi-Fi switching, but also switching of Wi-Fi routers.</p> <p>*4：The below conditions are judged to be "Good bearer connection".</p> <ul style="list-style-type: none"><li>・Bearer is DCM &amp;&amp; DCM connect &amp;&amp; DCM can communication area</li><li>・Bearer is Wi-Fi &amp;&amp; Wi-Fi connect</li></ul>	<p>Push sessions must continue to increase the chances of receiving push notifications. When waiting for a PING response, it takes up to 3 minutes to reconnect. Therefore, reconnection will be carried out when the bearer is switched.</p> <p>Since there are many opportunities for reconnection, the number of retries is suppressed to reduce the server load.</p> <p>* 1 Specifications: Wi-Fi interruption → return, DCM signal strength range → out of service area may switch frequently.</p> <p>* 2: Specifications: To prevent infinite retries.</p> <p>* 3: Specifications: Because it is necessary to regenerate the socket between the in-vehicle device and the server.</p>		Carry from 21MM
jp	Push_060	画面が表示されない(ディスプレイが消灯している)状態でPushを受けてほしくない	ACC-OFF時など、画面が黒画の時プッシュデータを受けてもユーザに通知(画面表示)できないため。 21MMで上記問題が発生したため。	010	<p>プッシュセッションの確立は以下条件の時にを行うこと</p> <p>①サービスフラグON かつ ACC-ON時 または ②ACC-ON中 かつ サービスフラグOFF→ON時</p>			21MM改善
en	Push_060	TMC don't want HU to receive a push when the screen is not displayed (the display is off)	When the screen is black, such as when ACC-OFF, the user cannot be notified (screen display) even if push data is received. Because the above problem occurred at 21MM.	010	<p>Establish a push session under the following conditions：</p> <p>① When the service flag is ON and at ACC-ON or ② During ACC-ON and when the service flag is OFF → ON</p>			Change
jp	Push_060			020	<p>プッシュセッションの終了は以下条件の時にを行うこと</p> <p>①ACC-OFF時 または ②ACC-ON中 かつ サービスフラグがON→OFF</p>			21MM改善
en	Push_060			020	<p>End the push session under the following conditions</p> <p>① At ACC-OFF or</p>			Change

LANG	ID	要求	理由	ID	仕様	理由	備考	Diff from 21MM
					② ACC-ON is in progress and the service flag is ON → OFF			
jp	Push_070	プッシュセッションが切れたことをサーバで検知したい	セッション切れたことをサーバで検知できないと、プッシュセッションを張り続けてしまうため。 無駄な通信を行ってしまうため。	010	通信を終了する場合には、grpcインスタンスを解放することにより、TCPコネクションをクローズすること。	きれいにセッションを切らないとセンタでセッション切れたことを検知できないため		21MM同等
en	Push_070	Server want to detect that the push session has deisconnected	If the server cannot detect that the session has expired, it will continue to hold a push session. Because it makes useless communication.	010	By releasing grpc instance, TCP Connection is closed.	Because it is not possible to detect that the session has disconnected at the server unless the session is officially terminated.		Carry from 21MM
jp	Push_070			020	一定間隔でHTTPレイヤのPINGを送信すること。 詳細は表5参照。	DCM内のNATテーブル期限切れを防ぐ、通信途絶検知のため。 期限切れになるとプッシュを受信できなくなるが、HUやセンタからは通信できないことを検知できないため。	送信間隔はDCMのNATテーブル期限による	21MM同等
en	Push_070			020	Send HTTP layer pings at regular intervals. Refer to Table 5 for details.	To detect communication blackouts to prevent NAT table expiration in DCM. When it expires, HU will not be able to receive pushes, but HU and server cannot detect that communication is not possible from the HU or center.		Carry from 21MM
jp	Push_080	なるべくプッシュセッションを張り続けたい	サーバから重要な情報をプッシュしたいため、プッシュセッションが切れている時間を極力減らしたいため	010	サービスフラグOFF、ACC-OFF以外(圏外や通信機器の切り替え等)でプッシュセッションが切れた場合、再接続を行うこと。 プッシュの再接続は「図2：接続状態遷移図」に従うこと。 再接続の例は図4を参照すること。	プッシュ通知を受信できる確率を増やすため、プッシュセッションは張り続けなければならないため。 ただし、サーバ負荷に負荷をかけないように、無限リトライは避ける必要があるため。 無駄なリトライは行わず、ベアラ切り替え時に再接続を試みる。	DCMから圏外かどうかの情報があるため、それを利用する？	21MM同等
en	Push_080	HU wants to keep pushing sessions as much as possible	Because server want to push important information from the server, HU wants to reduce the time when the push session is cut off as much as possible.	010	If the push session expires except for the service flag OFF and ACC-OFF (outside the service area, switching of communication devices, etc.), reconnect. To reconnect the push, follow "Fig. 2: Connection state transition diagram". See Figure 4 for an example of reconnection.	Push sessions must be kept up to increase the chances of receiving push notifications. However, it is necessary to avoid infinite retries so as not to put a load on the server load. Do not uselessly retry, and try to reconnect when switching bearers.		Carry from 21MM
jp	Push_090	機能情報値の定義をしたい	HUCOMの機能情報アップロード機能にて、センタにアップロードするデータの定義が必要なため。	010	機能情報値は以下とする。 ・キー値：serverpush ・値：1.0	Push_090と同じ理由。 値についてはプロビジョニング仕様書に従い、verの定義とした。		21MM新規
en	Push_090	TMC wants to define a function information value	Because it is necessary to define the data to be uploaded to the center in the uploading function of HUCOM.	010	Function information value is below. ・ Key name : serverpush ・ Value : 1.0	Same reason as Push_090. The value is defined as ver according to the provisioning specifications.		New

補足

- Push\_010\_020  
HUとサーバの構成図は以下。  
The configuration diagram of HU and server is as follows.

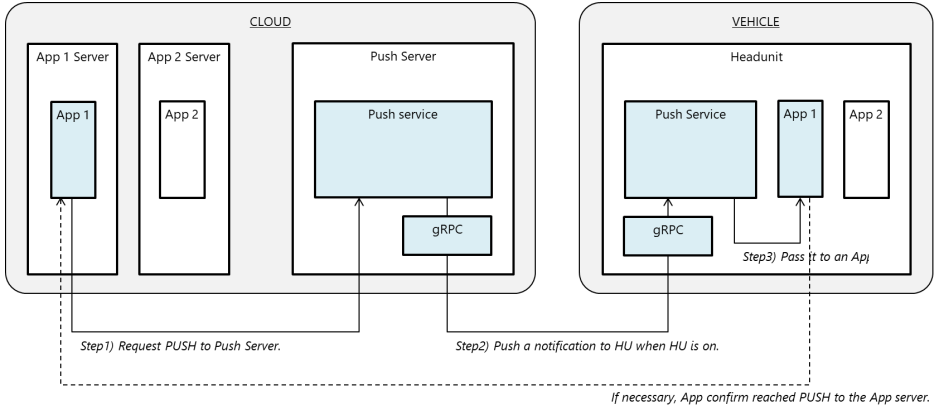


図1：システム構成図 / Fig. 1: System configuration diagram

- Push\_020\_010  
車載機の構成ごとのサーバプッシュ機能配置は以下  
The server push function placement for each in-vehicle device configuration is as follows.

表1：サーバプッシュ機能配置 / Table 1：Server push function placement

#	構成 Constitution	PF	機能配置 Function placement	備考 Note
1	HU+CECU+DCM+ADAS	ePF3.0	HU	
2	HU+CGW+DCM+ADAS	19PFv2	HU	
3	CECU+DCM+ADAS	ePF3.0	－	本仕様書の範囲外 Out of scope of this specification

- Push\_020\_020  
プロトファイルの各パラメータの説明を以下に記す。  
A description of each parameter in the protofile is given below.

表2：プロトファイルの各パラメータ / Table 2：Proto file parameters

Type	Parameters			Value	Example	Description
message	Result			-	-	プッシュサーバへHUの受信状態を示すメッセージ。 A message indicating the reception status of HU to the push server.
string			sequence_id	app_id、sub_idに紐づくシーケンスID	b84d92b9-ce75-48db-9339-f395d774098d	プッシュデータが2個連続で送信されてしまったものか、2個に分割して送信したものかを判別するID。 ACKを返却する際にどのデータに対するACKか判断するために利用する。 gRPCセッション確立時はオール0を設定すること。 UUIDを付与できるように、データ長は最大40byteとする。(「ハイフン」も文字列として扱い、削除しないこと) 最大データ長を超えたsequence_idを受信した場合は、上位40byteをsequence_idとすること。 An ID that determines whether the push data has been sent in two consecutive steps or divided into two pieces. It is used to determine which data the ACK is for when returning the ACK. Set to all 0 when connecting a gRPC session. The maximum data length is 40 bytes so that UUID can be assigned. ("Hyphen" should be treated as a string and should not be removed. ) If HU receive a sequence_id that exceeds the maximum data length, use the upper 40 bytes as the sequence_id.
enum			hu_ack_code	{STACK=0, SUCCESS=1, FAILURE=2, NOFUNCTION=3}	1	STACK(0)：スタック成功：プッシュ通知先がなく、スタックをした SUCCESS(1)：対象機能へプッシュ通知配布完了 FAILURE(2)：対象機能へプッシュを通知したが、エラーが返ってきた NOFUNCTION(3)：プッシュ通知ができなかった(通知先がなかった)
message	Response			-	-	プッシュサーバからの応答メッセージ。 Response from Push Server.
message		PushServerStatus		-	-	プッシュサーバの状態を示すメッセージ。 Status of Push Server.
enum			srv_response_code	{OK=0,RETRYAFTER,BUSY}	0	OK(0)：正常稼働時に応答。 RETRYAFTER(1)：サーバが、ヘッドユニットに一定時間待機させたい場合に、応答。詳細はretry_after_secの説明に従う。 BUSY(2)：RETRYAFTER以外で、サーバが忙しいときに応答。詳細は本書"2-5. Retry"に従う。  OK(0)の場合のみ、Notificationを有効とすること。  OK (0): Respond when operating normally. RETRYAFTER (1): Response when the server wants the head unit to wait for a certain time. For details, follow the description of retry_after_sec. BUSY (2): responds when the server is busy other than RETRYAFTER. For details, refer to "2-5. Retry" in this document.



						Only when OK (0), Notification should be valid.
fixed32			retry_after_sec	ヘッドユニットが待機すべき秒数	3600	"srv_response_code"が"RETRYAFTER"の場合のみ、有効となる値。HUはただちにコネクションを切断し、本通知を受けてから本値が示す秒数だけ待機した後、再びプッシュサーバにアクセスを行う。サーバが何らかの理由で一時的にサービス提供をやめたいとき、利用するパラメータ。 This will be available only when "srv_response_code" is set as "RETRYAFTER". HU will wait for a certain seconds designated by this value, and then, HU connects to Push Server again. This feature is used for a case Push Server wants to halt providing push service temporarily.
message		Notification		-	-	プッシュ通知の内容を示すメッセージ。 A message describes contents of a notification
fixed32			app_id	プッシュ先アプリのID	0x0013	プッシュ先アプリを示すID。あらかじめ本書で定義される。 An ID indicates an app should be notified. ID are allocated in this spec.
fixed32			sub_id	プッシュ先アプリのサブID	0x0001	プッシュ先アプリで利用するサブID。あらかじめ本書で定義される。 An ID indicates a type of notification treated on an App. ID are allocated in this spec.
bytes			app_data	アプリ毎の任意のデータ	(アプリ定義による)	アプリ毎の任意のデータ。 2048byte未満とすること。 A data that is set based on App's discretion. Must be less than 2048 bytes.
string			expires	有効期限	"Wed, 06 May 2020 00:52:21 GMT"	通知されたプッシュの有効期限。有効期限をどう扱うかは、アプリ側の判断による。 記述形式は 703仕様書「HUCOM Communication Spec」HUCOM_050_030のDATEタグについての記載を参照のこと。 An expiration date of a notification. How this date is treated is on App's discretion. A format follows "HUCOM_050_030" of "HUCOM Communication Spec" in 703_spec. ▲1.60
string			sequence_id	app_id、sub_idに紐づくシーケンスID	b84d92b9-ce75-48db-9339-f395d774098d	プッシュデータが2個連続で送信されてしまったものか、2個に分割して送信したものかを判別するID。 ACKを返却する際にどのデータに対するACKか判断するために利用する。 gRPCセッション確立時はオール0を設定すること。 UUIDを付与できるように、データ長は最大40byteとする。 最大データ長を超えたsequence_idを受信した場合は、上位40byteをsequence_idとすること。 An ID that determines whether the push data has been sent in two consecutive steps or divided into two pieces. It is used to determine which data the ACK is for when returning the ACK. Set to all 0 when connecting a gRPC session. The maximum data length is 40 bytes so that UUID can be assigned. If HU receive a sequence_id that exceeds the maximum data length, use the upper 40 bytes as the sequence_id.

表3：サーバプッシュのID / Table 3：Server push ID

app_id	app_id value	app_id description	sub_id	sub_id value	sub_id Description
00h	(Used by Push Server)		00h		Push ServerがHUの初回接続時の応答に利用する アプリケーションは利用しない  詳細は以下を参照のこと seq01_mainstream [004] ▲1.10  Push Server uses it for the response when the HU is connected for the first time. Do not use the application  Refer to below for details seq01_mainstream [004] ▲1.10
01h	HUCOM	通信ライブラリ Communication library	00h	(Reserve)	
			01h	Get_Service_flag	サービスフラグの取得 Get service flags
05h	Audio	オーディオ(ラジオ、TV) Audio (Radio, TV)	00h	(Reserve)	
			01h	SXM360L_Status	SXMのステータスアップデート SXM Status update
0Ch ▲3.00	VirtualAgent ▲3.00	エージェント Agent ▲3.00	00h	(Reserve)	
			01h	Home_Push	Home画面に対して必要な情報をPushする Push the required information to the Home screen
			02h	VR_Navi_Ons	Navi画面上で、エージェントとしてONSで情報を提供する(エージェント+後継サービス) On the Navi screen, provide information on ONS as an agent (agent + successor service)

app_id	app_id value	app_id description	sub_id	sub_id value	sub_id Description
			03h	VR_Notice	ユーザに対するPushやお知らせメールを格納するサービス(お知らせメール後継サービス) A service that stores pushes and notification emails for users (successor service to notification emails)
			04h	VR_Reco_Spot	ナビ地図上にお勧めスポットを表示する Display recommended spots on the navigation map
			05h ▲3.00	VR_PAI ▲3.00	PAI情報をホーム画面にプッシュする。 Push the PAI information to the Home Screen. ▲3.00
0Dh	PersonalIdentify	個人特定システム Personally Identifiable System	00h	(Reserve)	
			01h	Sync_Profile	Sync Profile ▲1.60
			02h	GetRegistrationStatus	'GetRegistrationStatus'のトリガ Shoulder tap for 'GetRegistrationStatus' ▲1.60
			03h ▲1.60	Sign_out	スマホアプリなどからサーバにサインアウトを要求する。 Request sign-out from the server from a smartphone app or the like.
			04h ▲1.60	Reset	Fleetユーザは自社システムからトヨタサーバにリセットを要求する。 Fleet users request a reset from their system to the TOYOTA server.
0Fh	Notification	お知らせ Notification	00h	(Reserve)	
			01h	Notice	お知らせ通知 Notification
10h ▲3.00	OTA ▲3.00	リプロ Reprogramming ▲3.00	00h	(Reserve)	
			01h ▲3.00	All_Usecase ▲3.00	以下のユースケースで利用する。 ①OTAシーケンス開始を通知する。 ②外部HMI(スマホアプリ, ブラウザ等)でのユーザ操作(承諾・中止)を通知する。 ③割り込み指示(ログ取得・キャンペーン中止)を通知する どのユースケースかをプッシュデータの中に含める。 詳細はOTA仕様書参照。  It is used in the following use cases. Include which use case in the push data. ①Notifies the start of the OTA sequence. ②Notify HU of user operations (accept/cancel) on external HMI (smartphone app, browser, etc.). ③Notify interrupt instructions (log acquisition/campaign stop). In push data, include data representing which of the above use cases. Please refer to OTA specification.
13h	Connected_Audio	Connected Audio	00h	(Reserve)	
			01h ▲1.50	Token	OneAppで各CP(Apple/Amazon/Spotify)アカウントと連携すると、AccessTokenをCPから入手し、Toyotaサーバへ渡す。入手したAccessTokenはHUからのStreaminAudio再生などに使用される。 AccessTokenがサーバへ渡されるタイミングをHUが把握するため、Push通知を行う。 When linking with each CP (Apple/Amazon/Spotify) account with OneApp, AccessToken is obtained from CP and passed to Toyota server. Obtained AccessToken is used for StreaminAudio playback from HU. In order for HU to grasp the timing when AccessToken is passed to the server, push notification is performed.
			02h ▲1.50	Preference	OneAppでMusicPreference(Explicit設定、DefaultSource、AudioQuality)の変更を受け、HUの設定を変更する必要がある。 各設定値をOneAppで変更したタイミングを把握するためPush通知を行う。 HU need to change the Hu settings in response to the change of MusicPreFrence (Explicit settings, DefaultSource, AudioQuality) in ONEAPP. Push notification is performed to understand the timing of each set value changed in OneApp.
14h	Destination_Assist	NA's Operator Service	00h	(Reserve)	
			01h	Get_req	サーバ側から位置情報の再送をしてほしいとき、目的地を取得しに来てほしいときにPush通知を行う。 Send push notifications when HH wants the server to resend location information and when HH wants the server to come to get the destination. ▲1.70
45h ▲1.50	Remote_Monitoring	Remote Monitoring	00h	(Reserve)	
			04h	Remo_Start_up	スマホによる遠隔マイカー見守りサービス起動要求 Request to start remote private car watching service by smartphone

app_id	app_id value	app_id description	sub_id	sub_id value	sub_id Description
			02h	End_Cause	<del>スマホによる遠隔マイカー見守りサービス終了を通知</del> <del>Notify the end of the remote car watching service by smartphone</del>
16h ▲1.50	Drive_Recorder	Drive-Recorder	00h	(Reserve)	
			01h	Dreco_Start_up	<del>スマホによる設定変更のための車載機起動要求</del> <del>On-board unit start request for setting change by smartphone</del>
			02h	Set_Info	<del>スマホによる設定変更のための設定情報通知</del> <del>Notification of setting information for changing settings using a smartphone</del>
17h ▲3.00	Camera_Image_Data ▲3.00	Camera Image Data ▲3.00	00h	(Reserve)	
			01h ▲3.00	Pro_Reco_up ▲3.00	<del>センターでアップロードされたデータにもとづき画像アップロード可否判断をした後、アップロードを開始するケース</del>  <del>センタの通知を受けて画像データをアップロードする。</del>  <del>A case where uploading is started after judging whether or not to upload an image based on the data uploaded at the center.</del>  Upload the image data upon receiving notification from the center. ▲3.00
			02h ▲3.00	<del>Gen_Reco_up</del> ▲3.00	<del>車載器に蓄積しておいた過去データを事後的にセンター要求で取得するケース</del> <del>A case where the past data accumulated in the in-vehicle device is acquired after the fact at the center request.</del> ▲3.00
			03h ▲1.50	Streaming_up	<del>センター側の要求でストリーミング映像データ取得を開始するケース</del> <del>A case where streaming video data acquisition is started at the request of the center</del>
			04h <del>▲4.50</del> ▲3.00	<del>Coll_Cond_Update</del> ▲3.00	<del>収集条件を更新する。</del> <del>Update the collection conditions:</del> ▲3.00
18h ▲1.50	Applog	アプリケーションログ Application log	00h	(Reserve)	
			01h	Log_up	サーバの要求で、HUのアプリケーションログをアップロードする。 Upload HU application logs at the server request.
19h ▲3.00	Navigation ▲3.00	ナビゲーション Navigation ▲3.00	00h ▲3.00	(Reserve) ▲3.00	
			01h ▲3.00	SendPOI ▲3.00	POI情報をHUに送る。 Send POI to HU. ▲3.00
1Ah ▲3.00	Remote warning ▲3.00	e-Care ▲3.00	00h ▲3.00	(Reserve) ▲3.00	
			01h ▲3.00	eCare_Ons ▲3.00	HUにWNGのONSを表示する。 Display a warning ONS in HU. ▲3.00

- Push\_050\_020  
gRPCのレスポンスコードに対するリトライ要否は以下。  
The retry for the gRPC response code is as follows.

表4：レスポンスコードとリトライ要否 / Talbe4：Response code and retry

gRPCステータスコード gRPC status code	HTTPステータスコード HTTP status code	リトライ要否 Retry
OK = 0	200 OK	×
CANCELLED = 1	499 Client Closed Request	×
UNKNOWN = 2	500 Internal Server Error	○
INVALID_ARGUMENT = 3	400 Bad Request	×
DEADLINE_EXCEEDED = 4	504 Gateway Timeout	○
NOT_FOUND = 5	404 Not Found	×

gRPCステータスコード gRPC status code	HTTPステータスコード HTTP status code	リトライ可否 Retry
ALREADY_EXISTS = 6	409 Conflict	×
PERMISSION_DENIED = 7	403 Forbidden	×
RESOURCE_EXHAUSTED = 8	429 Too Many Requests	×
FAILED_PRECONDITION = 9	400 Bad Request	×
ABORTED = 10	409 Conflict	×
OUT_OF_RANGE = 11	400 Bad Request	×
UNIMPLEMENTED = 12	501 Not Implemented	○
INTERNAL = 13	500 Internal Server Error	○
UNAVAILABLE = 14	503 Service Unavailable	○
DATA_LOSS = 15	500 Internal Server Error	○
UNAUTHENTICATED = 16	401 Unauthorized	×

参考資料 / Reference : (1)<https://github.com/googleapis/googleapis/blob/master/google/rpc/code.proto>  
(2)<https://github.com/grpc/grpc/blob/master/doc/statuscodes.md>

- Push\_050\_020

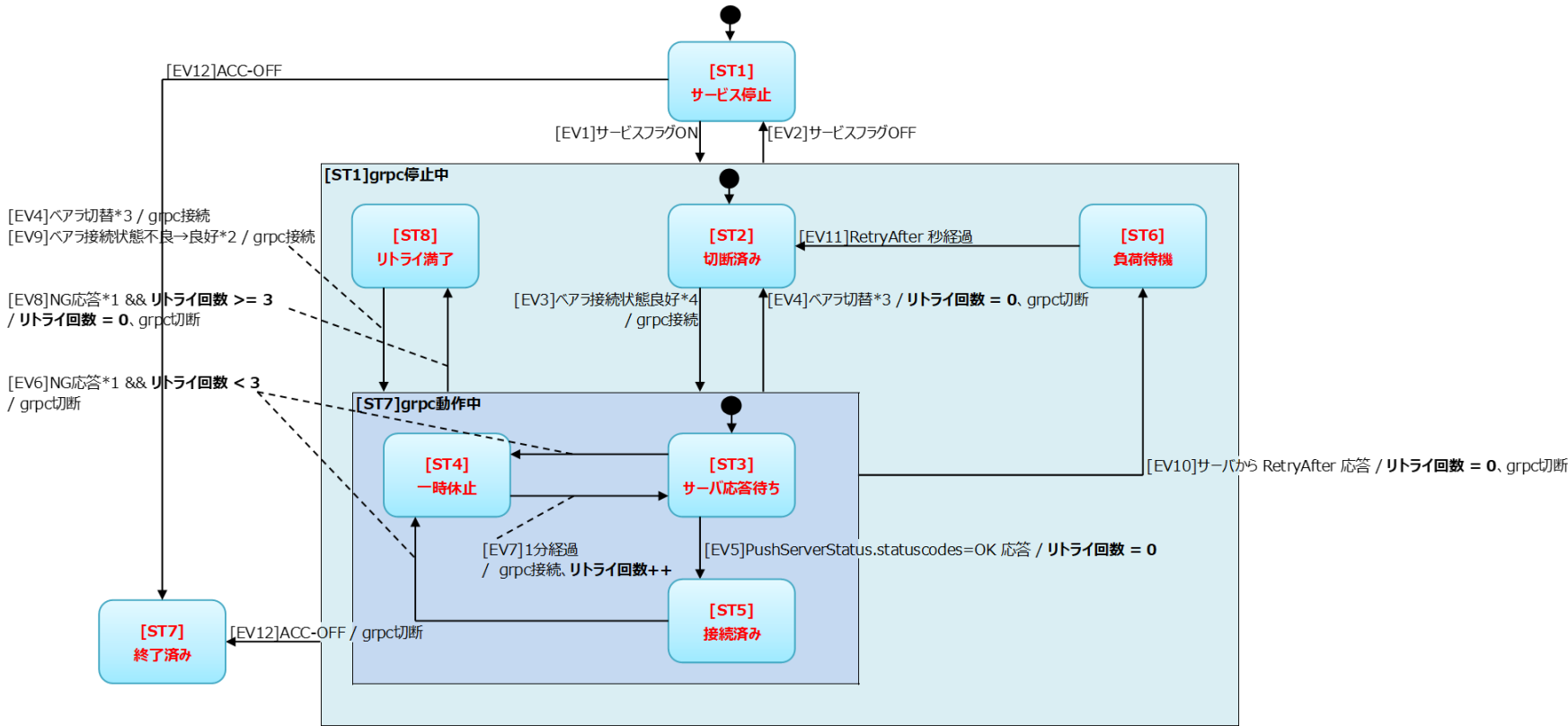


図2：接続状態遷移図

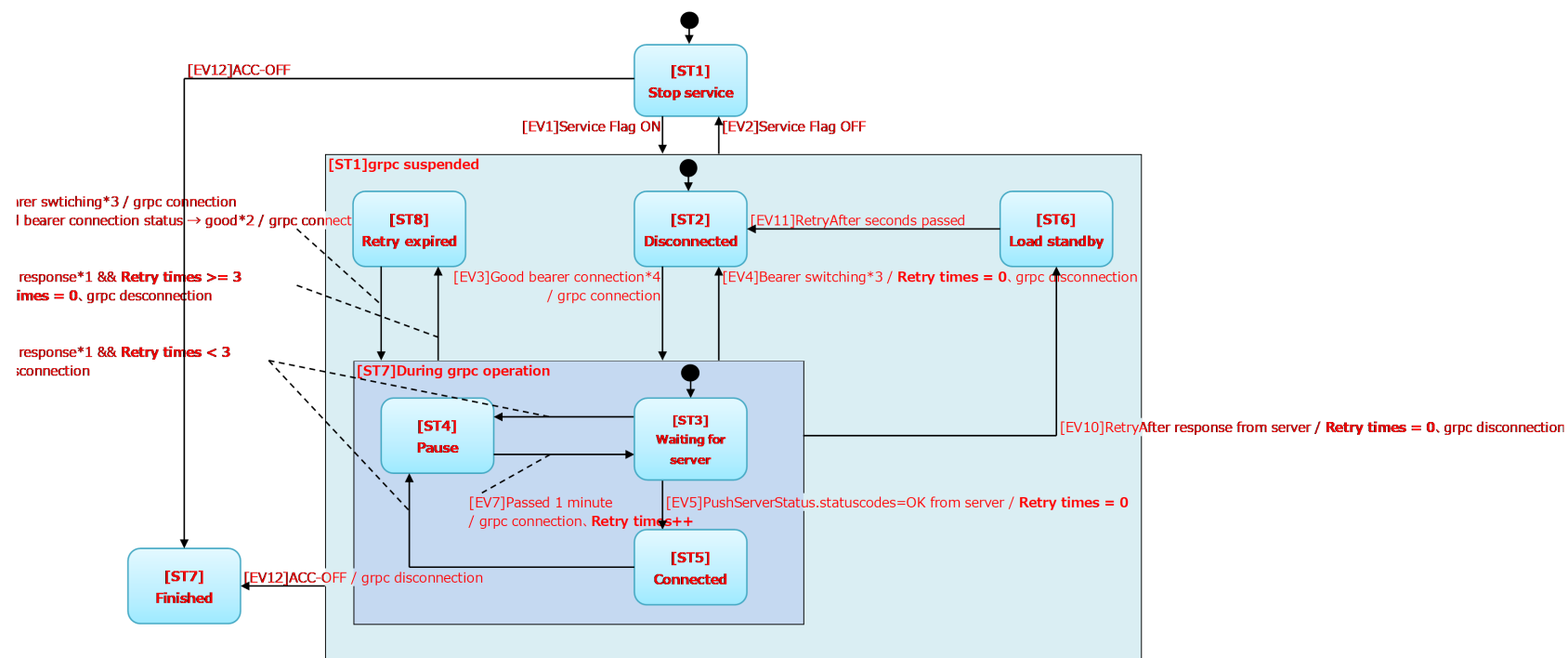


Fig 2 : Connection state transition diagram

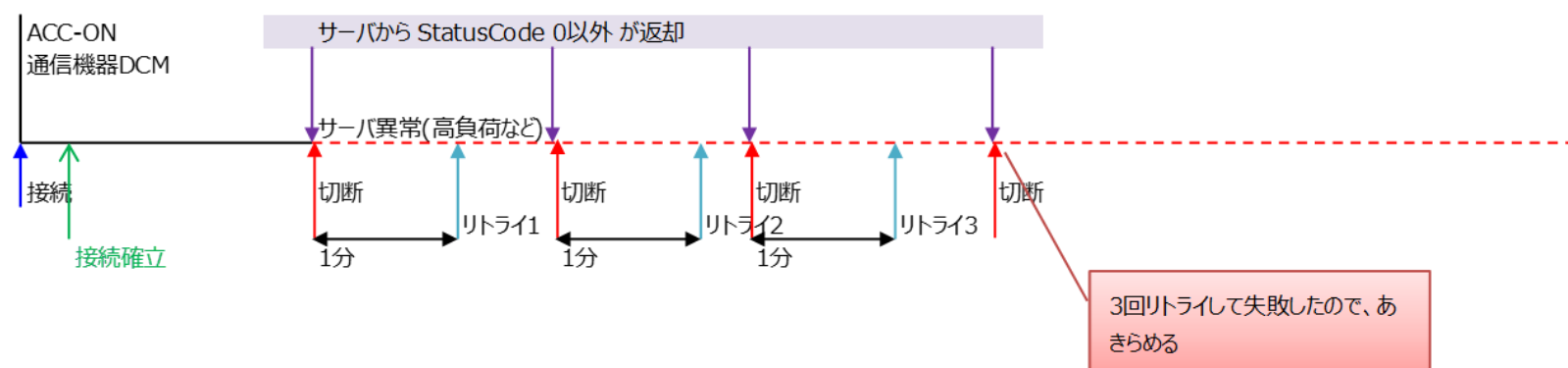


図3 : サーバ異常

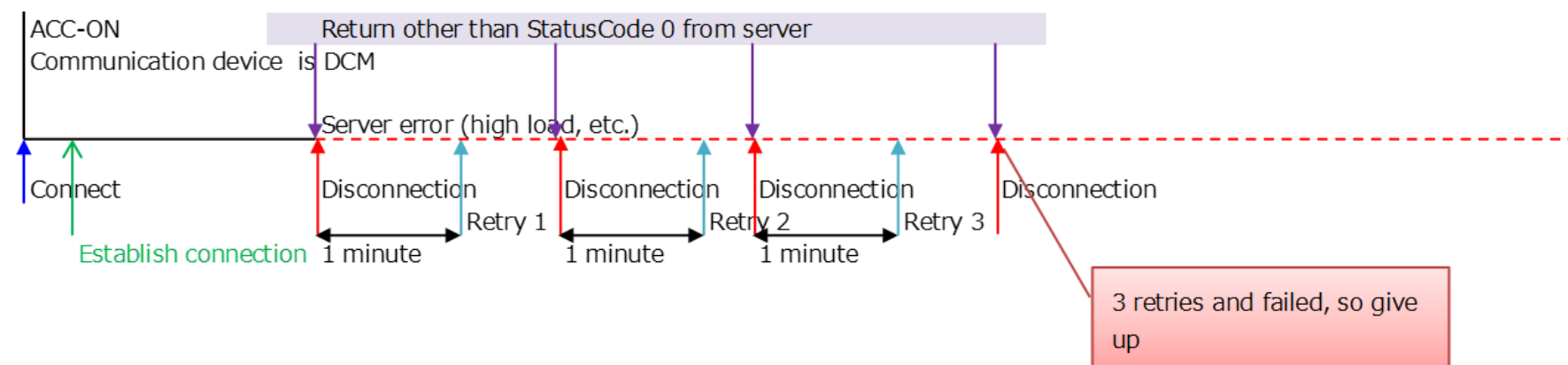


Fig 3 : Server error



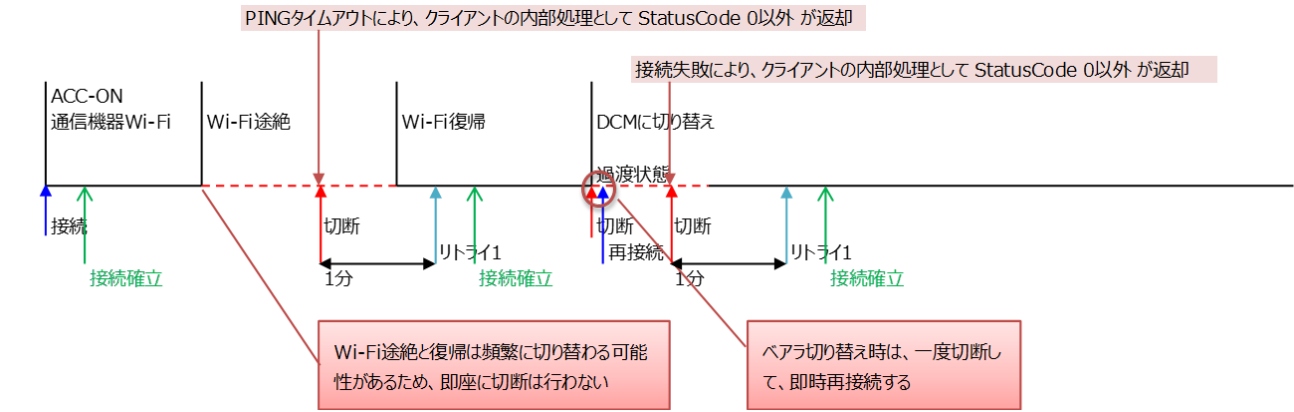


図4：Wi-Fi途絶→復帰後の通信ベアラ切り替え

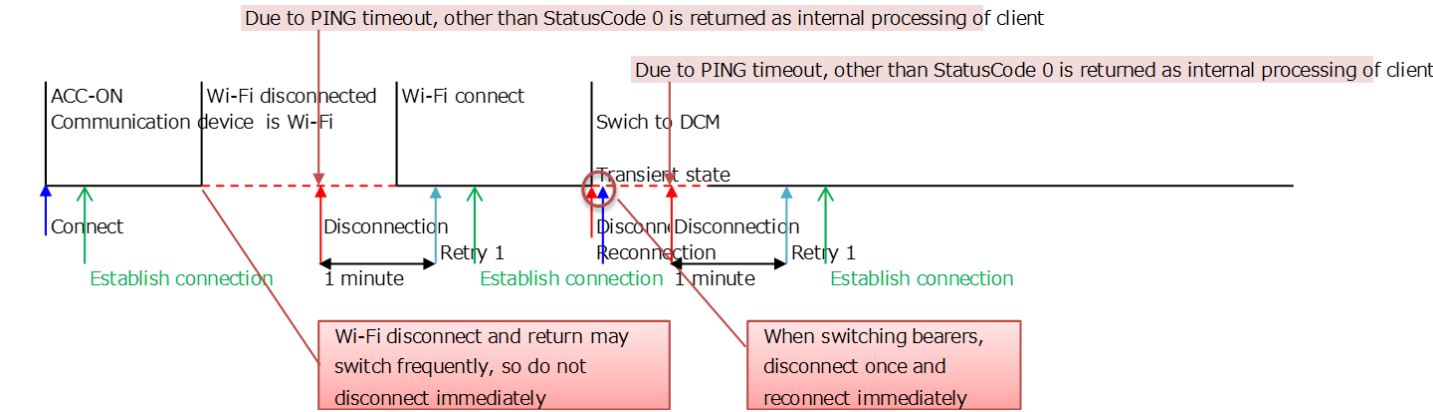


Fig 4：Wi-Fi interruption → communication bearer switching after recovery

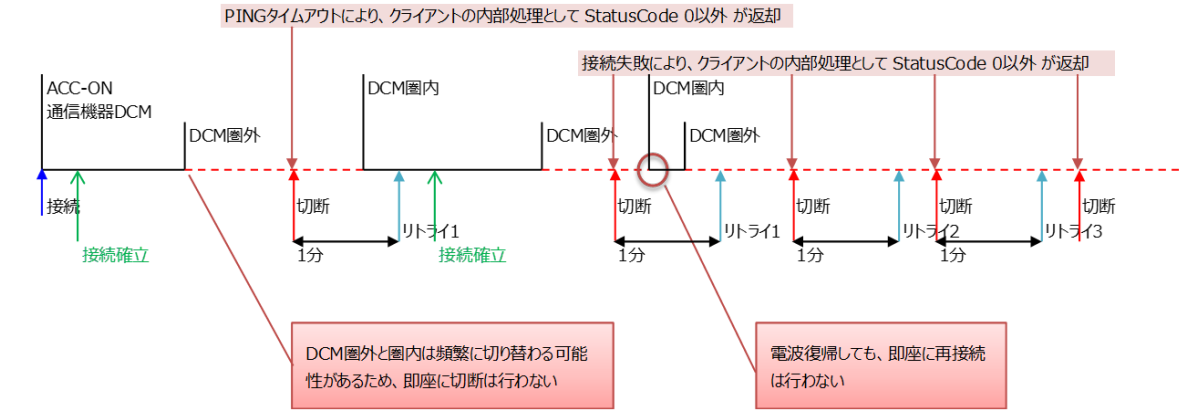


図5：DCM電波強度不安定（圏内→圏外を繰り返す）

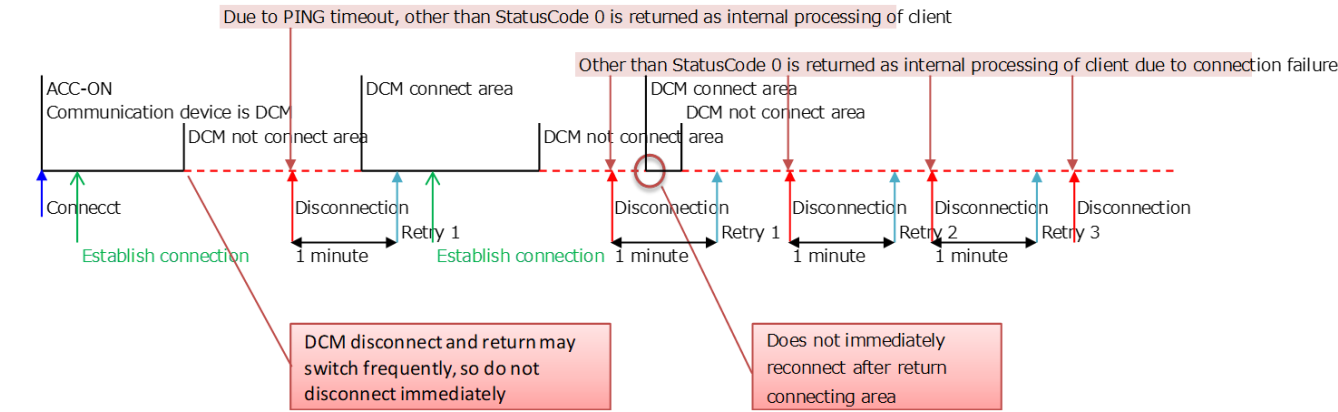


Fig 5：DCM signal strength unstable（Repeat from inside to outside the connecting area）

表5：PING定義 / Table 5：PING

N O.	Symbol	Description	Default Value		Pattern(A) Direct		Pattern(B) via Proxy			Reason of HU's Setting Value	Note
		*Reprint of [6. Reference # 2]	Client	Server	Client	Server	Client	Proxy	Server		
⑤	GRPC_ARG_KEEPALIVE_TIME_MS	This channel argument controls the period (in milliseconds) after which a keepalive ping is sent on the transport.	INT_MAX (disabled)	7200000 (2 hours)	120000 (2 minutes)	7200000 (2 hours)	120000 (2 minutes)	規定しない。gRPCと同様のパラメータが存在するかどうかは、採用するProxy次第となるため。	規定しない。ProxyとgRPC-server間のやりとり	・ pingを送信する頻度（ミリ秒）。 ・ DCM網の制約より短い時間とすること。 ・ ping送信時、下記のようなgRPCログが出	

								ただし、Keep-Alive監視時間 (GRPC_ARG_HTTP2_MIN_RECV_PING_INTERVAL_WITHOUT_DATA_MS に相当)はクライアントが GRPC_ARG_KEEPALIVE_TIME_MS で指定する時間よりも長くすること。  Not specified. Because the existence of parameters similar to gRPC depends on the parameters used. However, the Keep-Alive monitoring time (GRPC_ARG_HTTP2_MIN_RECV_PING_INTERVAL_WITHOUT_DATA_MS) must be longer than the time specified by the client in GRPC_ARG_KEEPALIVE_TIME_MS.	になるため自由に決めてよい。  Not specified. Server can decide freely because it is an exchange between proxy and gRPC-server.	力される。  CLIENT: Ping sent XXXXX  ・ Frequency of ping (milli second) ・ The time must be shorter than the restrictions of the DCM network. ・ The following gRPC log is output when ping is sent.  CLIENT: Ping sent XXXXX	
⑥	GRPC_ARG_KEEPALIVE_TIMEOUT_MS	This channel argument controls the amount of time (in milliseconds) the sender of the keepalive ping waits for an acknowledgement. If it does not receive an acknowledgment within this time, it will close the connection.	20000 (20 seconds)	20000 (20 seconds)	120000 (2 minutes)	20000 (20 seconds)	120000 (2 minutes)			PINGを送ってから、PINGのACKが返ってくるまでどれだけ待つか。この時間が経過すると、セッションをクローズしていく。 ・ タイムアウト発生時、gRPCログ上に下記が出力される。  keepalive watchdog timeout  After sending a ping, how long to wait until the ping acknowledgment is returned. After this time, the session is closed. ・ The following gRPC log is output when a timeout occurs.  keepalive watchdog timeout	・ PoC時には10秒で試験していたとのこと。送った後のPING 応答なので、もう少し短くてもよい。
⑦	GRPC_ARG_KEEPALIVE_PERMIT_WITHOUT_CALLS	This channel argument if set to 1 (0 : false; 1 : true), allows keepalive pings to be sent even if there are no calls in flight.	0 (false)	0 (false)	1 (true)	0 (false)	1 (true)			・ streamが存在せずとも、PINGを送ってよい か？ ・ trueの場合、streamを終了 (ClientReader→Finish()) した後でもpingが送信される。falseの場合、stream終了後は pingが送信されない。gRPCログにも「Ping sent」が出ない。  CLIENT: Ping sent XXXXX  ・ Can HU sent PING even if stream does not exist? ・ If true, ping is sent even after the stream ends (ClientReader → Finish ()). If false, no ping is sent after the stream ends. "Ping sent" does not appear in the gRPC log.  CLIENT: Ping sent XXXXX	・ streamを終了した後にTCP/IPを維持するユースケースはないため、FALSE(デフォルト)でもよい。
③	GRPC_ARG_HTTP2_MAX_PINGS_WITHOUT_DATA	This channel argument controls the maximum number of pings that can be sent when there is no other data (data frame or header frame) to be sent. gRPC Core will not continue sending pings if we run over the limit. Setting it to 0 allows sending pings without sending data.	2	2	0	2	0			・ Pushデータを受信していない間、ping送信可能な最大回数。 ・ Pushデータがなく、データを受信しないケースは異常ではないため、最大回数を規定してはならない。 ・ 本設定値を超える回数pingを送信しようとすると、下記のようなgRPCログが出力される。  CLIENT: Ping delayed XXXXX too many recent pings XXXXX  ・ The maximum number of times ping can be sent while no push data is received. ・ The case where there is no push data and no data is received is not abnormal, so the maximum number of times must not be specified. ・ If you try to send pings more than this setting value, the following gRPC log will be output.  CLIENT: Ping delayed XXXXX too many recent pings XXXXX	
①	GRPC_ARG_HTTP2_MIN_SENT_PING_INTERVAL_WITHOUT_DATA_MS	If there are no data frames being received on the transport, this channel argument controls the minimum time (in milliseconds) gRPC Core will wait between successive pings.	300000 (5 minutes)	300000 (5 minutes)	120000 (2 minutes)	300000 (5 minutes)	120000 (2 minutes)			・ 本設定値未満の間隔でpingを送ろうとするとブロックされる。 ・ ブロックされた場合、下記のようなgRPC ログが出力される。  CLIENT: Ping delayed [XXXXX]: not enough time elapsed since last ping. Last ping XXXXX  ・ If you try to send a ping at an interval less than this setting, it will be blocked. ・ If blocked, the following gRPC log will be output.	・ PING送信頻度が短い場合、本パラメータで不正PING(PING 攻撃)にならないよう調整するとよい。

②	GRPC_ARG_HTTP2_MIN_RECV_PING_INTERVAL_WITHOUT_DATA_MS	If there are no data frames being sent on the transport, this channel argument on the server side controls the minimum time (in milliseconds) that gRPC Core would expect between receiving successive pings. If the time between successive pings is less than this time, then the ping will be considered a bad ping from the peer. Such a ping counts as a 'ping strike'. On the client side, this does not have any effect.	N/A	300000 (5 minutes)	None (Server Paramater)	120000 (2 minutes)	None (Server Paramater)
④	GRPC_ARG_HTTP2_MAX_PENDING_STRIKES	This arg controls the maximum number of bad pings that the server will tolerate before sending an HTTP2 GOAWAY frame and closing the transport. Setting it to 0 allows the server to accept any number of bad pings.	N/A	2	None (Server Paramater)	2	None (Server Paramater)

CLIENT: Ping delayed [XXXXX]: not enough time elapsed since last ping. Last ping XXXXX	
<div><div><div>・ PING受信から次のPINGを受けるまでの時間の、サーバが許容できる最小値。</div><div>・ クライアントの「GRPC_ARG_KEEPALIVE_TIME_MS」より短い時間とすること。</div><div>・ The minimum value the server can tolerate between receiving a ping and receiving the next ping.</div><div>・ The time must be shorter than "GRPC_ARG_KEEPALIVE_TIME_MS" of the client.</div></div></div>	
<div><div><div>・ サーバが不正なpingを受け入れる最大数。設定値を超える不正pingを送った場合、サーバからGO_AWAYを受信し、gPRCが切断される。</div><div>・ 例えば、サーバへpingを大量に連射すると、不正なpingとしてカウントされる。カウントが最大数に達すると、下記gRPCログのようにGO_AWAYが返却され、切断される。0を設定すると、不正なpingを許容する回数の上限がなくなり切断されなくなる。</div><div>Received a GOAWAY with error code ENHANCE_YOUR_CALM and debug data equal to "too_many_pings"</div><div>If an ping exceeding the set value is sent, a GO_AWAY is received from the server and gPRC is disconnected.</div><div>・ For example, if a large number of pings are fired at a server, it is counted as an ping.When the count reaches the maximum number, GO_AWAY is returned and disconnected as shown in the following gRPC log.If 0 is set, there is no upper limit on the number of times an ping is allowed, and the ping will not be disconnected.</div><div>Received a GOAWAY with error code ENHANCE_YOUR_CALM and debug data equal to "too_many_pings"</div></div></div>	<div><div>・ PING送信頻度が短い場合、本パラメータを大きめにしておくといいが、今回は3分と長めにとっているため、デフォルトのままでよい。</div></div>