



Product Development

# FNV2-SOA Gateway

Version 0.9

Version Date: January 9, 2019

**UNCONTROLLED COPY IF PRINTED**

**FORD CONFIDENTIAL**

The copying, distribution and utilization of this document as well as the communication of its contents to others without expressed authorization is prohibited. Offenders will be held liable for payment of damages. All rights reserved in the event of the grant of a patent, utility model or ornamental design registration.

# Table of Contents

1. Glossary .....	3
1. Summary.....	3
2. Interfaces .....	4
3. High Level Design.....	4
3.1 Gateway .....	5
3.2 Gateway API .....	5
3.3 Gateway client.....	6
3.4 Publish flow .....	6
3.5 Subscribe flow .....	6
3.6 Gateway components .....	8
3.7 Gateway power and network management .....	8
4. Performance (if applicable) .....	9
5. Security features.....	9
6. Dependencies and Risks (if applicable).....	9
7. References .....	9



## 1. Glossary

---

	Description
<b>API</b>	Application Programming Interface
<b>ALM</b>	Application Lifecycle Manager
<b>ECG</b>	Enhanced Central Gateway
<b>ECU</b>	Engine Control Unit
<b>SOA</b>	Service Oriented Architecture
<b>IPC</b>	Inter Process Communication
<b>POSIX</b>	Portable Operating System Interface for uniX
<b>VIM</b>	Vehicle Information Manager

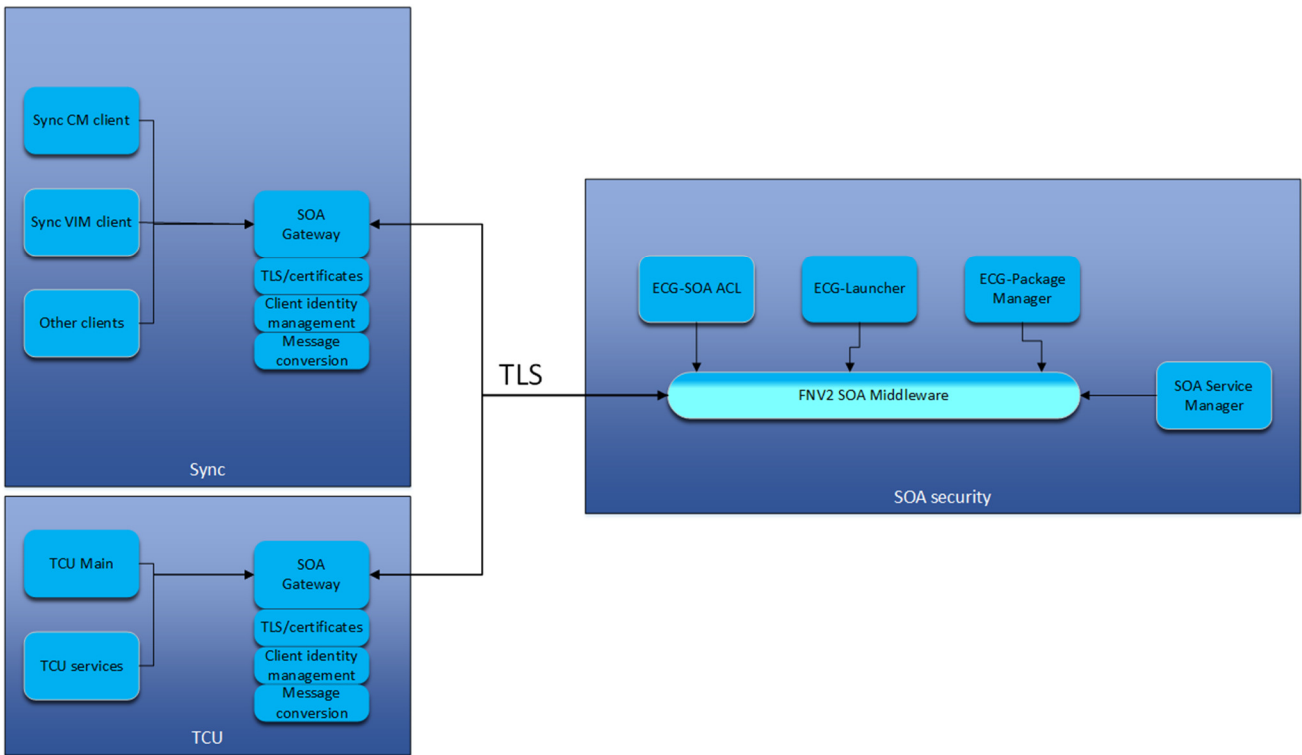
## 1. Summary

---

The SOA framework relies on a set of security requirements [1] which main purpose is to prevent unauthorized access to ECG resources and services. For fast ECUs connected to the ECG, those security requirements are fulfilled by the SOA Gateway. All the clients on an ECU will interface the ECG through the Gateway translating messages over IPC to SOA messages. The clients do not need to know anything about the Gateway as its functionality is abstracted out by the SOA API.

## 2. Interfaces

The SOA Gateway is part of each fast ECUs connected to the ECG. This is the main interface between any SOA clients on those ECUs and the ECG SOA broker.



## 3. High Level Design

On fast ECUs as TCU or Sync, there are mainly three types of SOA components: the Gateway itself, the various SOA Gateway clients connecting to the Gateway to access ECG services and the IPC linking the Gateway to the clients. Those clients can be either service consumers if they request information from services hosted on the ECG for instance or service providers if they offer services to the ECG or to other components.

For the Gateway clients to talk to the Gateway, a communication channel has to be defined. This is the SOA Gateway API over Unix Domain Socket (ALM IPC or IPC Lite now).

Those three pieces, Gateway, Gateway client and Gateway API are describes in the following sections.



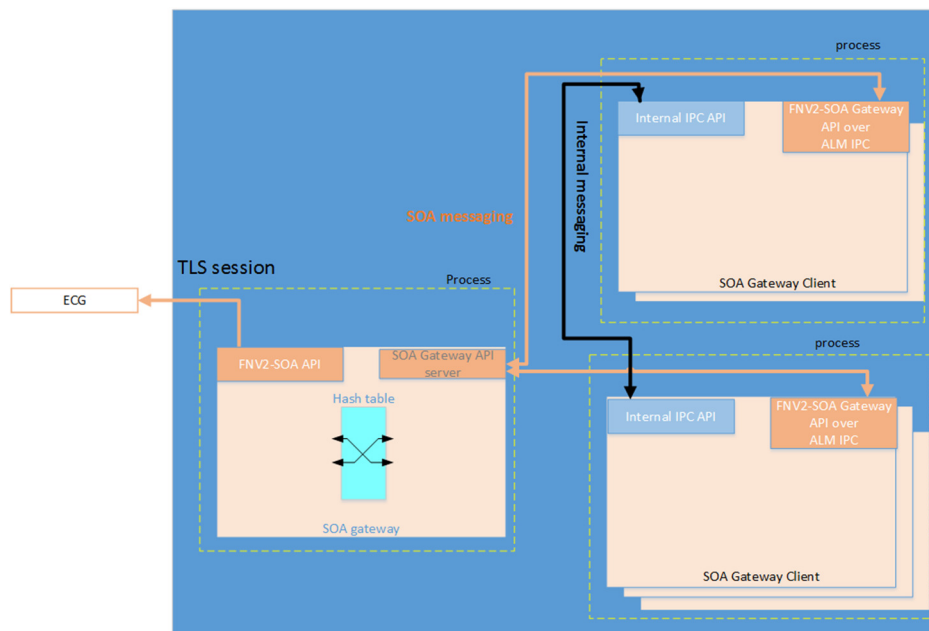
### 3.1 Gateway

The SOA Gateway is a software component residing on all fast ECUs on the network willing to interface with the ECG and to communicate over the SOA framework. The Gateway architecture should be flexible enough to accommodate various OSes running on those ECUs and minimize implementation differences to make the development and support easier.

For this purpose, the following architecture is proposed. The communication between the ECG and Gateway uses a regular SOA messaging not different from what is used on the ECG. This capability is provided by the SOA API.

The Gateway also needs a different interface to enable the Gateway clients to connect to it. This IPC mechanism should be the same on TCU, Sync or any other ECUs. To leverage what is being developed for ECG the Gateway will rely on this IPC as the transport mechanism between the Gateway client and the Gateway. This IPC is generic enough to accommodate various OSes and environments.

One of the role of the Gateway in addition to security is message conversion from the Gateway clients to the ECG. To be able to route properly requests and responses, a hash table is used.



### 3.2 Gateway API

Gateway clients residing on the fast ECUs are connected to the Gateway and will use the Gateway API to publish and subscribe to topics.

The clients do not have to do anything particular to deal with the Gateway as it will be abstracted out by the API.

This API is the same as the SOA API i.e the public methods exposed to the client are the same as the SOA API (see: SOA API description Document/Wiki). What differs is mainly the implementation as the Gateway API will be relying on the ALM IPC rather than on the MQTT protocol.

### 3.3 Gateway client

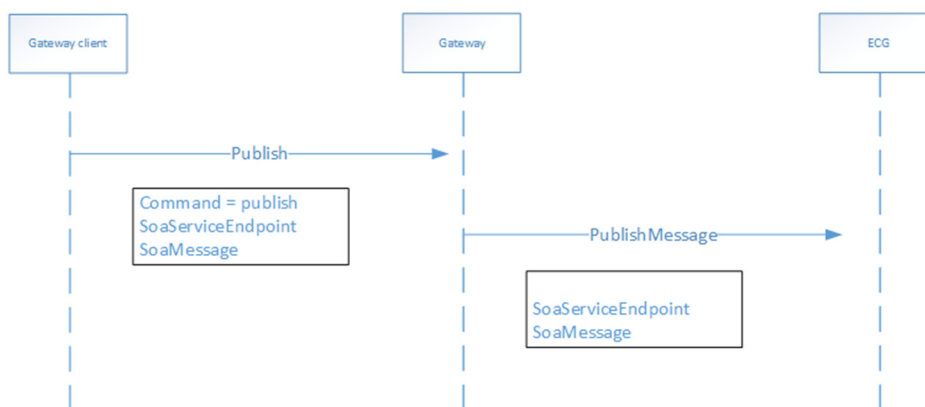
The Gateway client is the component connecting to the SOA gateway. They can communicate between each other using the existing local communication framework but must use the Gateway API to talk to the Gateway.

### 3.4 Publish flow

The purpose of the diagram below is to illustrate how a simple publish from the Gateway client would work. The API only needs to send to the Gateway:

- The type of SOA command: publish, remoteCall, RegisterService, etc.
- The endpoint
- The SOA message

The first publish from the client to the gateway is just a command sent to Gateway. The PublishMessage from the gateway to the ECG is a MQTT publish.



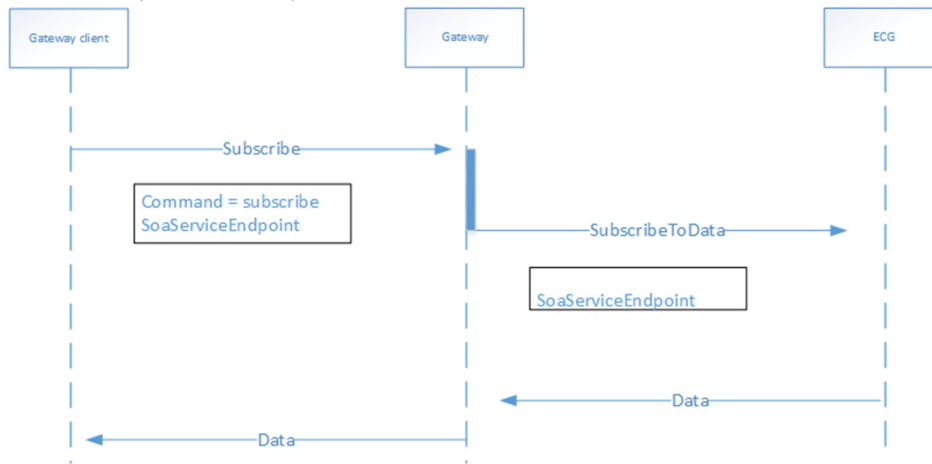
### 3.5 Subscribe flow

For the subscribe,

- The type of SOA command: subscribe
- The endpoint



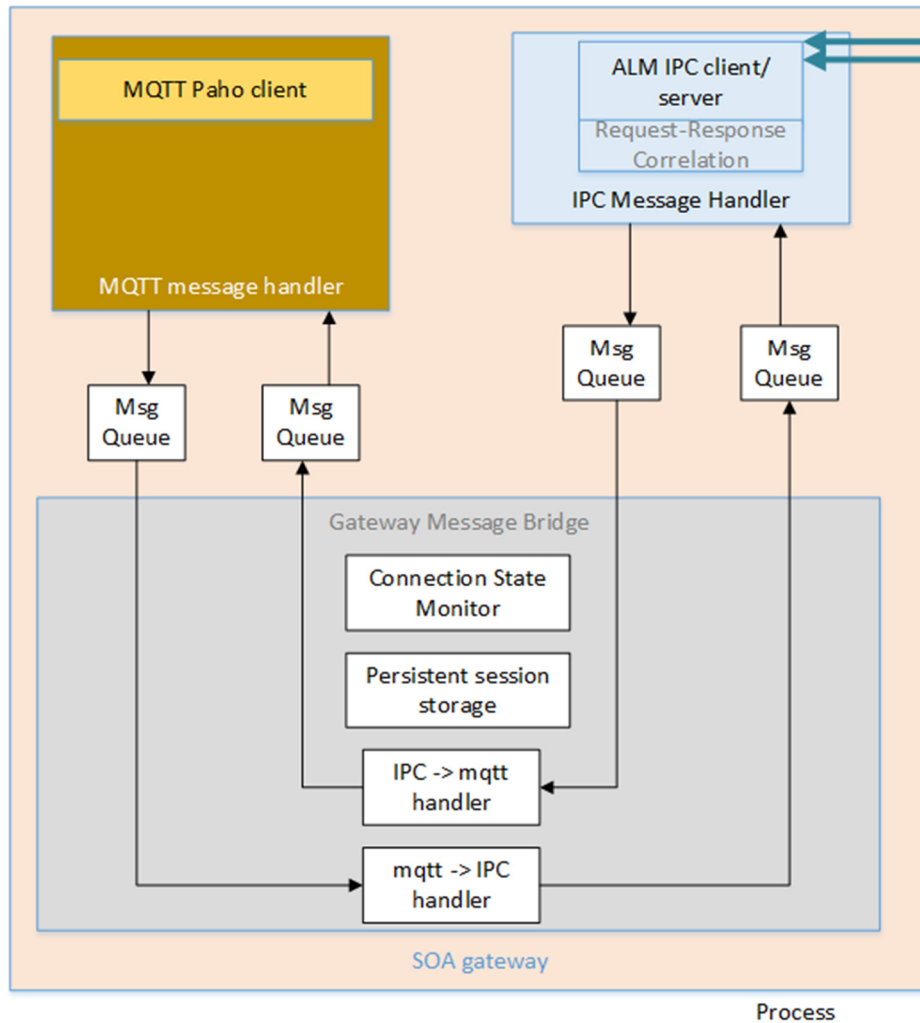
The first subscribe from the client to the Gateway is a simple command targeted to be translated to an actual MQTT subscribe by the Gateway.





### 3.6 Gateway components

The gateway will receive all the pub/sub commands in the IPC message handler.



### 3.7 Gateway power and network management

Gateway API should be notified when:

- Network is not available or connection to the ECG broker failed. In that case, the publish and subscribe calls will return an error.
- ECG/Network are back on line. The Gateway clients should be notified so publishes/subscribes can be resumed.





## 4. Performance (if applicable)

---

The message rate through the Gateway could be high particular if Sync transitions from CAN signals to VIM. So no message translation will happen but each message received by the Gateway need to be identified and forwarded to the appropriate client.

## 5. Security features

---

Security features are described in a separate document, see: SOA Security Architecture Document/Wiki

As the IPC is done over UDS, the client credentials are uid and gid are available from the gateway side to allow/reject connections.

## 6. Dependencies and Risks (if applicable)

---

The Gateway relies on the IPC which is going to be developed first for the ECG. This IPC will be wrapped in the SOA API.

## 7. References

---

SOA security High Level Design Document / Wiki

SOA Gateway Detailed Design Document / Wiki