1.0 Overview

ISO-14229 defines the DTCSnapshotRecordNumber to be a 1-byte value limited to the range 0x01 ... 0xFE for servers supporting multiple DTC snapshot records, depending on the DTC mask record sent (see pp.183/184).

The DTCExtendedDataRecordNumber is even more restricted to a set of values by ISO-14229. For instance, 0x00 and 0xF0 ... 0xFD are reserved, whereas values between 0x90 ... 0xEF are used for OBD. Only values between 0x01 ... 0x8F may be used to request vehicle manufacturer specific extended data records.

Many CANdela templates today implement protocol services 19 04 and 19 06 in a simplified way: both, DTCSnapshotRecordNumber and DTCExtendedDataRecordNumber are either stored in a fixed data component or in a data proxy in the Request message (see "FD" components in Figure 1).

1 0x00 is reserved for OBD compliant servers, 0xFF requests the server to report all snapshot records at once
However, to model the ranges for every DTC mask record and extended data record more exact, the Request message should contain a multiplexer instead (see Figure 2).

In the Fault memory diagnostic instance, the multiplexer component can then be used to store a data object defining the DTC snapshot record numbers / extended data record numbers available for a particular DTC.

Example: let us assume, snapshot record numbers 0x01, 0x03 and 0xF0 are available for DTC “P000001”. To specify this set of values, the multiplexer must contain a data object which defines the values (see column “Data types” on the left side of Figure 2) for DTC “P000001”.

Figure 1: Current implementation of Protocol service 19 06 found in templates
How to create UDS protocol services 19 04 and 19 06

3

Application Note  AN-IDG-1-010

Figure 2: Proposed new implementation of Protocol service 19 06

This re-implementation of Protocol services 19 04 and 19 06 results in a better code generation (e.g. using GENy) because code generators can use the data types to create additional code restricting and validating the data sent.

The following chapters provide a description on how to modify the implementation of both protocol services in Expert view as well as the procedure of entering DTCs and snapshot / extended data in Standard view.

2.0 Editing the Template (Expert view)

2.1 Protocol service 19 04

2.1.1 Request Message

The request message of protocol service 19 04 must contain the following components:

- Constant (Service ID, 1-byte, value = 0x19)
- Constant (Subfunction, 1-byte, value = 0x04)
- Proxy (DTC, 3-byte)
- Multiplexer (selector = DTC) for “AvailableDTCSnapshotRecordNumbers”
The request message should look like in Figure 3.

![Figure 3: Request message of protocol service 19 04](image)

### 2.1.2 Positive Response Message

The positive response message of protocol service 19 04 must contain the following components:

- Constant (Service ID, 1-byte, value = 0x59)
- Constant (Sub function, 1-byte, value = 0x04)
- Proxy (DTC, 3-byte)
- Proxy (Status of DTC, 1-byte)
- Iteration (End) containing components:
  - Proxy (1-byte fixed) for "DTCSnapshotRecordNumber"
  - Proxy (1-byte fixed) for "DTCSnapshotRecordNumberOfIdentifiers"
  - Multiplexer (selector = DTC) for "SnapshotRecords"

The positive response message should look like in Figure 4.

![Figure 4: Positive response message of protocol service 19 04](image)
2.2 Protocol Service 19 06

Protocol service 19 06 is similar to protocol service 19 04. The difference is that instead of the DTC snapshot record number the extended data record number is transmitted in the request message.

2.2.1 Request Message

The request message of protocol service 19 06 must contain the following components:

- Constant (Service ID, 1-byte, value = 0x19)
- Constant (Sub function, 1-byte, value = 0x06)
- Proxy (DTC, 3-byte)
- Multiplexer (selector = DTC) for “AvailableExtendedDataRecordNumbers”

The request message should look like in Figure 5.

![Figure 5: Request message of protocol service 19 06]

2.2.2 Positive Response Message

The positive response message of protocol service 19 06 must contain the following components:

- Constant (Service ID, 1-byte, value = 0x59)
- Constant (Sub function, 1-byte, value = 0x06)
- Proxy (DTC, 3-byte)
- Proxy (Status of DTC, 1-byte)
- Iteration (End) containing components:
  - Fixed data (1-byte) containing the “ExtendedDataRecordNumber”
  - Multiplexer (selector = Fixed data) for “ExtendedDataRecords”

The fixed data component “ExtendedDataRecordNumber” must contain a single data object referencing a text table data type. The text table data type itself must define all available extended data record numbers except 0xFF (which is reserved by ISO).

The positive response message should look like in Figure 6.

**Note:** The multiplexer uses component “ExtendedDataRecordNumber” as selector!
3.0 Editing the Fault memory (Standard view)

In Standard view, the DTCs of the Fault memory using these protocol services must be edited differently to regular DTCs. To specify the combination of available DTC snapshot record numbers and extended data record numbers for a DTC it is necessary to create text table data types containing the available snapshot record numbers / extended data record numbers for this particular DTC. The data types must then be referenced by data objects stored in the tables of the “AvailableDTCSnapshotRecordNumbers” and “AvailableExtendedDataRecordNumbers” component.

3.1 Creating the Data type specifying available snapshot record numbers

The following is a description on how to create a data type specifying available snapshot record numbers. Please note that data types specifying extended data record numbers must be created in the same way.

1. Go to chapter “Data types”. In the entry mask, click on [New…].
2. Select option Text table. The properties dialog of the data type appears.
3. Enter a memorable name (e.g. “Snapshot Record Numbers (01..0A)” if the data type specifies the range 0x01 to 0x0A).
4. Set Bit length to 1Byte.
5. Add available snapshot record numbers to the table.
6. Click on [OK].

Repeat these steps for every DTC that has its own set of available snapshot record numbers. If all DTCs share the same set, create one data type only and re-use it for every DTC.

3.2 Specifying available snapshot record numbers for a particular DTC

1. Go to diagnostic instance “Fault memory”.
2. Add a new DTC to the DTC table.
3. Click on the tab card of the “AvailableDTCSnapshotRecordNumbers” component (see component “qq” in Figure 7).

4. In the combo box (under “DTCMaskRecord” in Figure 7) select the DTC created above.

5. Below the combo box, select the second option.

6. Add a new data object to the table.
   Set the data type to the text table data type created above.

![Figure 7: Tab card “Available DTC snapshot record numbers”](image)

The “AvailableDTCSnapshotRecordNumbers” component now specifies the snapshot record numbers available for this particular DTC.

**Note:** The same procedure applies to extended data records (component “ss” in Figure 7).

### 3.3 Adding snapshot records

1. Click on the tab card of the snapshot data (see Figure 8).

2. In the combo box, select the DTC you like to add snapshot data.

3. Below the combo box, select the second option.

4. Add data objects to the table.

![Figure 8: Tab card “Snapshot data”](image)
4.0 Additional Resources

VECTOR APPLICATION NOTE
AN-IDG-1-001 How to get a CANdela document translated
AN-IDG-1-003 How to use the CANdelaStudio command line
AN-IDG-1-004 ODX support in Vector tools
AN-IDG-1-005 How to edit states and state groups
AN-IDG-1-006 CANdelaStudio product activation FAQ
AN-IDG-1-009 How to associate UDS Record numbers with Snapshot data

5.0 Contacts

<table>
<thead>
<tr>
<th>Vector Informatik GmbH</th>
<th>Vector CANtech, Inc.</th>
<th>VecScan AB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ingersheimer Straße 24</td>
<td>39500 Orchard Hill Pl., Ste 550</td>
<td>Theres Svenssons Gata 9</td>
</tr>
<tr>
<td>70499 Stuttgart</td>
<td>Novi, MI  46375</td>
<td>41755 Göteborg</td>
</tr>
<tr>
<td>Germany</td>
<td>USA</td>
<td>Sweden</td>
</tr>
<tr>
<td>Tel.: +49 711-80670-0</td>
<td>Tel: +1-248-449-9290</td>
<td>Tel: +46 (0)31 764 76 00</td>
</tr>
<tr>
<td>Fax: +49 711-80670-111</td>
<td>Fax: +1-248-449-9704</td>
<td>Fax: +46 (0)31 764 76 19</td>
</tr>
<tr>
<td>Email: <a href="mailto:info@vector-informatik.de">info@vector-informatik.de</a></td>
<td>Email: <a href="mailto:info@vector-cantech.com">info@vector-cantech.com</a></td>
<td>Email: <a href="mailto:info@vecscan.com">info@vecscan.com</a></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Vector France SAS</th>
<th>Vector Japan Co. Ltd.</th>
<th>Vector Korea IT Inc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>168 Boulevard Camellinat</td>
<td>Seafort Square Center Bld. 18F</td>
<td>Daerung Post Tower III, 508</td>
</tr>
<tr>
<td>92240 Malakoff</td>
<td>2-3-12, Higashi-shinagawa,</td>
<td>182-4 Guro-dong, Guro-gu</td>
</tr>
<tr>
<td>France</td>
<td>Shinagawa-ku</td>
<td>Seoul, 152-790</td>
</tr>
<tr>
<td>Tel: +33 (0)1 42 31 40 00</td>
<td>Tokyo 140-0002</td>
<td>Republic of Korea</td>
</tr>
<tr>
<td>Fax: +33 (0)1 42 31 40 09</td>
<td>Japan</td>
<td>Tel.: +82-2-2028-0600</td>
</tr>
<tr>
<td>Email: <a href="mailto:information@vector-france.fr">information@vector-france.fr</a></td>
<td>Email: +81 3 5769 7800</td>
<td>Fax: +82-2-2028-0604</td>
</tr>
<tr>
<td></td>
<td>Fax: +81 3 5769 6975</td>
<td>Email: <a href="mailto:info@vector-korea.com">info@vector-korea.com</a></td>
</tr>
</tbody>
</table>