NEMSIS TAC Whitepaper

NEMSIS V3 StateDataSet Guide

Date

January 6, 2016

Authors

Joshua Legler – NEMSIS Consultant
Jorge Rojas – NEMSIS Data Manager / Analyst
N. Clay Mann – NEMSIS P.I.

Overview

The NEMSIS V3 standard includes two "primary" data sets: DEMDataSet (snapshot information about EMS agencies) and EMSDataSet (EMS patient care reports). It also contains "secondary" data sets, including CADDataSet (incident dispatch information), MedDeviceDataSet (medical device event information), and StateDataSet (state reporting requirements and resources). This guide explains StateDataSet, including its purpose and recommended uses.

This guide is written for software developers. It may be of interest to state EMS data managers and EMS agency data managers as well.

Purpose of StateDataSet

EMS agencies and their software vendors must obtain information about their state's EMS data requirements in order to properly configure their patient care reporting software. The NEMSIS Technical Assistance Center (NEMSIS TAC) created the StateDataSet XML Schema (XSD) to provide a consistent format in which to share NEMSIS V3 state-specific information. StateDataSet covers custom data elements; state-recognized certification/licensure levels; state-permitted procedures, medications, and protocols; EMS agencies; and facilities (such as hospitals). StateDataSet was introduced in NEMSIS version 3.4.0.

Vendors of state-level systems are encouraged to build into their systems the ability to generate StateDataSet files. Vendors of local-level systems are encouraged to build into their systems the ability to process StateDataSet files as part of configuration.

Vendors are not required to support StateDataSet in their software. However, StateDataSet has advantages over other methods of conveying state-specific information:

- StateDataSet files can be validated using the NEMSIS StateDataSet XSD.

- StateDataSet files can be easily processed by systems.
- StateDataSet files can be easily viewed online in human-readable form.
- StateDataSet is based on the core NEMSIS V3 XSDs.

StateDataSet Contents and Structure

StateDataSet files contain the following information regarding a state or territory:

- Custom Data Elements
- State Required Elements
- State Certification/Licensure Levels
- Procedures Permitted by the State
- Medications Permitted by the State
- Protocols Permitted by the State
- EMS Agencies
- Facilities (hospitals, etc.)

The StateDataSet data dictionary, API, and XSD files are available on the Version 3 Data Dictionaries page of the NEMSIS Web site.

All sections in StateDataSet are optional, so a state may publish a StateDataSet file containing information for just one section and then fill in additional sections later as information becomes available. If a state does not regulate or provide resources for a particular section, that section may be omitted from the StateDataSet file.

StateDataSet at the State Level

States and territories can publish StateDataSet XML files that are compliant with the NEMSIS V3 StateDataSet XSD. Compliant files published with the NEMSIS TAC are made available to software companies to ensure local software is properly configured to state requirements. Other states may also access these files to, for example, mimic custom elements, etc. State EMS data managers may use the State Data Set Builder provided by the NEMSIS TAC, or the vendor of their state-level NEMSIS-compliant software may develop the ability to generate a StateDataSet file. Vendors of state-level systems are encouraged to build into their systems the ability to generate StateDataSet files.

NEMSIS State Data Set Builder

The NEMSIS V3 State Data Set Builder enables state EMS data managers to create a new StateDataSet file or edit an existing one. The State Data Set Builder is available to all states at www.nemsis.org/media/statedatasetbuilder. For information on using it, see the NEMSIS V3 State Data Set Builder User Guide.

Publishing StateDataSet Files

StateDataSet files should be submitted to the NEMSIS TAC for publishing within NEMSIS V3 state-specific repositories. The NEMSIS TAC validates StateDataSet files and ensures that their content is reasonable before publishing them.

StateDataSet files can be sent to the NEMSIS TAC via e-mail. However, the NEMSIS TAC recommends that developers of state systems automate the submission process using Git. To obtain rights to submit via Git, please contact the NEMSIS TAC.

StateDataSet at the EMS Agency Level

Vendors of local-level systems are encouraged to build into their systems the ability to process StateDataSet files as part of configuration.

NEMSIS StateDataSet Viewer

When a state submits a StateDataSet file to the NEMSIS TAC, it is made available on the state's NEMSIS V3 progress report page, where it can be viewed online. The online view provides a user-friendly way to browse the contents of a StateDataSet file. State progress report pages are available at www.nemsis.org/supportV3/stateProgressReports.

Retrieving StateDataSet Files

StateDataSet files are published by the NEMSIS TAC within each state's repository. For more information about the NEMSIS V3 repositories, see the NEMSIS V3 Resource Repository Guide. If a state has published a StateDataSet file, it is contained in the Resources folder and is named StateDataSet.xml. Software developers are encouraged to clone the state repositories and set up automated systems to monitor and process updates.

StateDataSet Usage Guidelines

This section sets forth guidelines for the each section of StateDataSet. Recommendations are provided for state systems to generate and for local systems to process information in a StateDataSet file.

StateDataSet (the document element of a StateDataSet file)

The `StateDataSet` element contains a mandatory attribute, `@timestamp`.

- **State systems:** Set `@timestamp` to the date/time at which the file was generated.

- **Local systems:** Use `@timestamp` to protect against inadvertently processing an older version of a StateDataSet file than one that has already been processed.

State

`dConfiguration.01` is the only mandatory data element in a StateDataSet file. It identifies the state represented in the file by its two-digit numerical ANSI state code.

- **State systems:** Generate `dConfiguration.01` in the StateDataSet file in the same way as for DEMDataSet files.
- **Local systems:** Use `dConfiguration.01` to protect against processing a file for a state that is not within an EMS agency's service area. Agencies that serve multiple states may need to process a StateDataSet file from each state.

Custom Data Elements

`dCustomConfiguration` and `eCustomConfiguration` contain custom data elements defined by the state.

A NEMSIS-compliant system may be customized with data elements or values that are not defined in the NEMSIS V3 standard. When data are exported from the system, the XML contains a `dCustomConfiguration` or `eCustomConfiguration` section to define the custom elements and a `dCustomResults` or `eCustomResults` section to record the responses to the custom elements within each record. See Approaches to Using NEMSIS V3 Custom Elements for more information about defining and recording custom element information in a NEMSIS XML file.

If a custom element is defined by a state, it should be implemented the same way in all agencies within the state. For each custom element defined by the state, NEMSIS data generated by each local system should contain exactly the same information in `dCustomConfiguration` or `eCustomConfiguration`. The best way to ensure this is for the state system to provide custom element definitions in the NEMSIS `dCustomConfiguration` and `eCustomConfiguration` formats.

- **State systems:** Generate `dCustomConfiguration` and `eCustomConfiguration` sections in the StateDataSet file in the same way as for DEMDataSet and EMSDataSet files.
- **Local systems:** Process the information in `dCustomConfiguration` and `eCustomConfiguration` from StateDataSet to set up custom elements within the system. (After the elements have been created by the system, further work may be necessary for someone to decide where the elements will be placed within the system's data entry form.) Internally, local systems should identify a state-provided custom element definition by not only the `@CustomElementID` attribute but also the state ID, since there is no guarantee that `@CustomElementID` is unique across states. When exporting NEMSIS data, local systems should generate `dCustomConfiguration` or `eCustomConfiguration` exactly as provided in StateDataSet for each custom element defined by the state.

State Required Elements

`dState` and `eState` contain the list of data elements required by the state.

Each state maintains a list of data elements that it requires or recommends agencies to submit to the state EMS data system. (This is usually viewed as a legal requirement rather than a technical requirement, since many "required" elements are not always applicable, depending on the agency or incident. In other words, this list usually represents the elements that should be "turned on" for collection within a patient care reporting system.)

The state required element list may be represented differently in various state systems. A system may contain a configuration area where an administrator can select elements from the complete NEMSIS element list in order to directly populate the `dState` and `eState` sections in NEMSIS DEMDataSet and EMSDataSet file exports. A system may contain a configurable data entry form, where an administrator selects data elements to be shown on the form. Or, a system may support multiple data entry form templates, each with data elements placed within it, with one form set as the system default.

- **State systems:** Generate `dState` and `eState` sections in the StateDataSet file in the same way as for DEMDataSet and EMSDataSet files. This is straightforward for systems that directly support a configuration area for the `dState` and `eState` sections. For systems that indirectly support state required elements through their data entry form configuration, the systems should identify all NEMSIS elements that have been selected for use in the data entry form to generate the `dState` and `eState` sections in the StateDataSet file.

- **Local systems:** State required elements are identified by their XML element names (e.g., "eRecord.01"). Process the list of elements in dState and eState to ensure that all of those elements are turned on for collection in the system's data entry forms. Local systems may go further by adding visual cues in the user interface for elements that are state-required.

State Certification/Licensure Levels

`dConfiguration.02` contains the list of certification/licensure levels recognized by the state.

- **State systems:** Generate `dConfiguration.02` in the StateDataSet file in the same way as for DEMDataSet files. `dConfiguration.02` should not simply contain all values defined in the NEMSIS standard; it should only contain values representing the certification/licensure levels that are recognized by the state.

- **Local systems:** Restrict the list of values within certification/licensure-related data elements to match the values included in `dConfiguration.02`. Most certification/licensure-related data elements do not use the same data values as `dConfiguration.02`; systems should be configured to manage the straightforward translation between the value lists.

Procedures and Medications Permitted by the State

`dConfiguration.ProcedureGroup` contains the list of procedures permitted by the state, by certification/licensure level. `dConfiguration.MedicationGroup` contains the list of medications permitted by the state, by certification/licensure level.

State systems generally have a configuration area where administrators can select the list of procedures and the list of medications that should be available, depending on certification/licensure level. Each agency then selects from the appropriate master list when identifying the medications and procedures that it permits its personnel to use. For example, the state may require agencies at the AEMT level to carry either fentanyl or morphine. From a StateDataSet perspective, the state permits AEMT agencies to

use both fentanyl and morphine. An AEMT agency within the state, however, may choose to carry only fentanyl. When DEMDataSet is generated for that agency, dConfiguration.04 (Medications Permitted by the State) should contain both values. However, dConfiguration.MedicationGroup should include a section applicable to the AEMT level that contains the medication code for fentanyl but not morphine in dConfiguration.07 (EMS Agency Procedures).

- **State systems:** StateDataSet is structured so that the lists of state permitted procedures and medications are grouped by certification/licensure level. Generate a dConfiguration.ProcedureGroup and dConfiguration.MedicationGroup section for each certification/licensure level included in dConfiguration.02 (State Certification/Licensure Levels). dConfiguration.06 (EMS Certification Levels Permitted to Perform Each Procedure) and dConfiguration.08 (EMS Certification Levels Permitted to Administer Each Medication) do not use the same data values as dConfiguration.02 (State Certification/Licensure Levels); state systems should be configured to manage the straightforward translation between the value lists.

- **Local systems:** Restrict the list of allowed procedures and medications, by certification/licensure level, to the lists in StateDataSet. In agency demographic data exports, the values in dConfiguration.07 (EMS Agency Procedures) and dConfiguration.09 (EMS Agency Medications) should be a subset of the values listed in StateDataSet for the certification/licensure level. (There may be exceptions where an EMS agency has been granted permission by the state to have a procedure or medication that is not normally permitted for the certification/licensure level.)

Protocols Permitted by the State

dConfiguration.05 contains the list of protocols permitted by the state.

- **State systems:** Generate dConfiguration.05 in the StateDataSet file in the same way as for DEMDataSet files.

- **Local systems:** Restrict the list of allowed protocols based on the list in StateDataSet. In demographic data exports, the values in dConfiguration.10 (EMS Agency Protocols) should be a subset of the values listed in StateDataSet. (There may be exceptions where an EMS agency has been given permission to implement a protocol that is not normally recognized by the state.)

EMS Agencies

dAgency contains a list of EMS agencies recognized by the state.

Usually, EMS agency data managers already know the ID, number, and name of their own agency. The purpose of dAgency in StateDataSet is to enable agency data managers to properly configure the reporting of agencies other than their own for eScene.02 (Other EMS or Public Safety Agencies at Scene) and eScene.03 (Other EMS or Public Safety Agency ID Number) on patient care reports. Local systems may also use the information in dAgency to configure the reporting for eDisposition.01 (Destination/Transferred To, Name) and eDisposition.02 (Destination/Transferred To, Code) in cases where a patient is transferred to another EMS agency.

- **State systems:** State systems maintain a list of all EMS agencies within a state so that they can process incoming data from each agency. Generate a `dAgency.AgencyGroup` section in the StateDataSet file for each active agency. The values of `dAgency.01` (EMS Agency Unique State ID), `dAgency.02` (EMS Agency Number), and `dAgency.03` (EMS Agency Name) should be the same as what the system generates in the DEMDataSet file for each agency.

- **Local systems:** Populate selection lists for `eScene.ResponderGroup` in patient care report data entry forms based on the values in StateDataSet, with eScene.04 (Type of Other Service at Scene) set to the value representing "Other EMS Agency" (some other values, such as "EMS Mutual Aid," may also be appropriate). Populate selection lists for `eDisposition.DispositionGroup` in patient care report data entry forms based on the values in StateDataSet, for cases where the EMS agency transfers patient care to another EMS agency. Local systems may also provide a configuration area where an administrator can identify from the full list the few EMS agencies with which the EMS agency works.

Facilities

`dFacility` contains lists of facilities, such as hospitals, recognized by the state.

If a facility is recognized by a state, it should be implemented the same way in all agencies within the state. For each facility recognized by the state, NEMSIS data generated by each local system should contain exactly the same information in `dFacility`. The best way to ensure this is for the state system to provide facility information in the NEMSIS `dFacility` format. This information enables local systems to configure the reporting not only for `dFacility` but also for elements related to incident locations and destination locations in EMSDataSet.

- **State systems:** Generate the `dFacililty` section in the StateDataSet file in the same way as for DEMDataSet files, except that it should contain all facilities recognized by the state and not any non-state-recognized facilities that may be set up by local agencies using the state system.

- **Local systems:** Process the information in `dFacility` to set up a master list of facilities. When an administrator selects a facility as being one that the agency uses, the information from the master list should be copied into the agency's demographic data. A facility that has been selected for use by an agency should be made available in both the `eScene` and `eDisposition` sections of the patient care report data entry form (or a local system may allow an administrator to select the sections in which the facility should be available). Local agency demographic data may contain additional facilities beyond those recognized by the state, but facility information for state-recognized facilities should match the information in StateDataSet.

Handling Updates / Change Control

A StateDataSet file represents a point-in-time snapshot. It does not contain historical data. For example, when a facility closes, it should be removed from the StateDataSet file. State and local systems need to maintain historical information about that facility, because it may have been the scene or destination location on previously entered PCRs. However, it should no longer be available to report on future PCRs. Systems usually handle this by flagging the facility record as inactive. Inactive records should not be included in a StateDataSet file.

Local systems should compare a new StateDataSet file to the previous version of the file to identify areas that changed. The NEMSIS standard does not support globally unique identifiers for entities such as custom data elements and facilities, which makes it difficult to recognize whether a change represents a new entity or a modification to an existing entity. The NEMSIS TAC recommends that developers use data warehousing "slowly changing dimension" techniques to handle changes.

States should allow a period of time for the adoption of StateDataSet updates at the local level.

Conclusions

StateDataSet brings a higher level of completeness and consistency to state-provided resources. As a result, state data managers can be more confident that changes in their environment are being implemented within local EMS agencies, and EMS agencies and their vendors can stay up-to-date with the latest state resources in a predictable format.