

NEMSYS 3.5 UUID

Questions and Answers

November 22, 2017

- Introduction 1
- Why is a UUID proposed? 1
- What objects would have a UUID attribute added? 2
- What would a UUID look like? 2
- Would UUIDs be mandatory? 3
- Could UUIDs be generated on the fly when generating a NEMSYS XML export (like CorrelationIDs can be)? 3
- How would UUIDs impact the ability to translate data from pre-3.5 versions to 3.5? 3
- How would UUIDs impact the ability to translate data from 3.5 to pre-3.5?..... 3
- Could a state system accept data in both 3.5 and pre-3.5 versions? 3
- Would end users be affected by UUIDs? 4
- With UUIDs, would a system be able to notify another system about deleted PCRs? 4
- Would the same object (such as a person) have the same UUID across different agencies?..... 4
- If an agency changes software vendors, would UUIDs change? 4
- Could a state assign UUIDs for objects such as agencies and facilities in StateDataSet?..... 4
- Would UUIDs replace CorrelationIDs for certain elements? 5
- Could the UUID be used as the sole identifier for a PCR? 5

Introduction

A proposed change request for NEMSYS 3.5 is the addition of a universally unique identifier (UUID) attribute on certain group elements in the NEMSYS DEMDataSet and on the PatientCareReport group element in the NEMSYS EMSDataSet. When generated following an established algorithm, a UUID is, practically speaking, guaranteed to be unique from all other UUIDs, without the need for a central authority to coordinate the generation of UUIDs.

Why is a UUID proposed?

NEMSYS-compliant Receive and Process systems have had difficulty processing agency demographic data in current versions of NEMSYS. The difficulty is mostly caused by the fact that agency demographic data

contain snapshot information about objects that persist (and slowly change) over time. When a new demographic data submission is received by a Receive and Process system, it is difficult for the system to determine whether an object within the data (for example, personnel) is a new object (for example, a newly hired personnel) or an update to an existing object (for example, an existing personnel whose name has changed).

If objects within DEMDataSet (and PatientCareReport in EMSDataSet) had a UUID, then they could be reliably tracked over time even if all data elements within the objects were modified.

What objects would have a UUID attribute added?

In DEMDataSet:

- DemographicReport
- dAgency.AgencyServiceGroup
- dAgency.AgencyYearGroup
- dContact.ContactInfoGroup
- dConfiguration.ConfigurationGroup
- dConfiguration.ProcedureGroup
- dConfiguration.MedicationGroup
- dLocation.LocationGroup
- dVehicle.VehicleGroup
- dVehicle.VehicleCertificationLevelsGroup
- dVehicle.YearGroup
- dPersonnel.PersonnelGroup
- dPersonnel.ImmunizationsGroup
- dPersonnel.LicensureGroup
- dPersonnel.CertificationLevelGroup
- dDevice.DeviceGroup
- dFacilityGroup
- dFacility.FacilityGroup

In EMSDataSet:

- PatientCareReport

What would a UUID look like?

The NEMESIS XSM Schema (XSD) would require UUIDs to be generated using the [IETF RFC 4122](#) standard. RFC 4122 defines 4 algorithms for UUID generation. Any of the four algorithms would be acceptable. RFC 4122-compliant implementations exist for Java, .Net, JavaScript, PHP, Python, Ruby, etc. The XSD would constrain UUIDs using the following regular expression:

[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[1-5][a-fA-F0-9]{3}-[89abAB][a-fA-F0-9]{3}-[a-fA-F0-9]{12}

A sample valid UUID is:

e48cd734-01cc-4da4-ae6a-915b0b1290f6

A sample data element with a UUID attribute added is:

<dVehicle.VehicleGroup UUID="e48cd734-01cc-4da4-ae6a-915b0b1290f6">...

Would UUIDs be mandatory?

Yes. If the UUIDs are not mandatory, then receiving systems would encounter the same problems that they face today.

Could UUIDs be generated on the fly when generating a NEMESIS XML export (like CorrelationIDs can be)?

No. A Collect Data system would need to generate a UUID for an object at the time it is created. The system would need to store the UUID for as long as the object exists. Every time the system includes the object in a NEMESIS XML export, the system would need to include the UUID, and the UUID should never change.

How would UUIDs impact the ability to translate data from pre-3.5 versions to 3.5?

Pre-3.5 data could not be translated to 3.5 using an XSL transformation, because it is not possible to create and persistently store UUIDs during the translation process. Each time an object is translated, it is possible that a different UUID would be generated, which would not be acceptable.

For data originally created in a pre-3.5 version, it may be possible for the software product to implement a one-time process to add UUIDs to all existing objects. Then the software could translate the data prior to export and ultimately export the data in 3.5 format.

How would UUIDs impact the ability to translate data from 3.5 to pre-3.5?

Version 3.5 data could be translated to earlier versions using an XSL transformation. The translation process would ignore the new UUIDs.

Could a state system accept data in both 3.5 and pre-3.5 versions?

Many states accept data in two versions (such as 3.3.4 and 3.4.0 currently). A state could also accept data in both 3.5 and pre-3.5 versions. Data received in pre-3.5 versions would be processed as it

currently is, and with the current limitations and challenges around identifying updates to objects. Data received in version 3.5 could be processed more efficiently by the state system, utilizing the UUID attributes to identify object updates.

Would end users be affected by UUIDs?

UUIDs would be managed “under the hood” by software products. UUIDs would not be exposed to end users in data entry, and end users would not have the ability to edit UUIDs.

With UUIDs, would a system be able to notify another system about deleted PCRs?

The NEMSIS web services standard does not specify a mechanism for notifying another system that a PCR has been deleted. If the standard were to specify such a mechanism in the future, it could use UUIDs as the way of uniquely identifying a PCR.

Would the same object (such as a person) have the same UUID across different agencies?

The UUID would be assigned by the Collect Data system. If a person works for multiple agencies that all share the same Collect Data system, it is likely that the person would have the same UUID across those agencies. If the person works for multiple agencies that use different Collect Data systems, the person would have a different UUID within each system. The UUID would uniquely identify the person within an agency, not across agencies.

If an agency changes software vendors, would UUIDs change?

If the new software is capable of receiving and processing NEMSIS XML data, then existing data could be exported from the old system and imported into the new system, preserving existing UUIDs.

If the new software is not capable of receiving and processing NEMSIS XML data, the agency would need to set up its data manually in the new system, and the new system would generate new UUIDs for everything. The first time the agency submits demographic data to the state using the new system, it would appear to the state system that the agency had removed all previous objects and created all new objects. It’s possible that some manual work would need to be performed in the state system to re-identify objects after the software change.

Could a state assign UUIDs for objects such as agencies and facilities in StateDataSet?

Possibly, but if so, it would be necessary to require Collect Data systems to be able to import data from a StateDataSet document. This requirement is not currently included in the proposal.

Would UUIDs replace CorrelationIDs for certain elements?

No. For ease of implementation, the proposal is that CorrelationIDs, and the requirements related to them, would be unchanged.

Could the UUID be used as the sole identifier for a PCR?

No two PCRs would ever have the same UUID, so the UUID can be used as the unique identifier for a PCR. However, for security purposes, it would also be necessary for receiving systems to verify that the entity submitting the PCR has permission to do so. Usually, a receiving system needs to check the values in the EMSDataSet header (which identify the agency) in order to determine permissions.