

LiveCad Technical Document

The purpose of this document is to provide some technical details on how LiveCad works and the process of extracting data from the cad database.

LiveCad operates by taking basic incident information (address, city, nature, etc.) from the cad database and sending it through a webservice to our LiveCad server (hosted on the Azure platform). Our server will then parse the data and update the appropriate configuration views in real-time. No data is sent back to the cad database and nothing is written to the cad database (it's read-only access). It's important to note that LiveCad only displays Fire & EMS runs, not Law Enforcement runs.

The common way of extracting data from the cad database is that 2 stored procedures are set up, either in the cad database itself, or in a separate database on the same database server. The first stored procedure extracts Incident information (address, city, nature, etc.), and the second stored procedure extracts Truck Status information (truck name, current status, etc.). These 2 stored procedures can either be set up by the cad vendor (if cooperative), or NIPST can attempt to extract the information needed.

Once the 2 stored procedures are set up (which are read-only), then NIPST would install (or work with your IT department to install) a small Windows service ("LiveCad Service") somewhere on the network that has access to the cad database and access to the internet. The LiveCad Service runs the 2 stored procedures on a timed interval (usually every second), then determines if there is any new data to be sent. If so, the LiveCad Service sends the incident and truck status data to the LiveCad server via a webservice over HTTPS (secure). The data is then processed by our server and updates the appropriate views (web page) in real-time. The LiveCad Service is pretty small (about 1 MB) and data is sent in JSON format, so the service has very little impact on the computer/server performance and on internet traffic.

The process of getting the 2 stored procedures created is usually the biggest "hurdle" to get completed. Some cad vendors see LiveCad as a product that is similar or would compete with something they want to sell, so they are sometimes reluctant to assist, or will charge a hefty fee to create what's needed. Other cad vendors (or the database administrators for the cad system) have been cooperative in creating the needed items. A third option is for NIPST to create a separate database on the same database server that would house the needed stored procedures, and to set up desired filters for law enforcement and other types of "sensitive runs". For scenarios where it's desired to have NIPST set up the stored procedures, the database would need to be hosted on site, not "hosted by the cad vendor on their cloud".

For agencies who have their cad database hosted by the vendor in the "vendor's cloud", LiveCad may not be a viable option, unless the cad vendor is cooperative in providing what is needed. You'll need to contact your cad vendor if yours is a hosted environment and see if LiveCad can be configured to work on their database end.

The following 3 pages are technical details on how the stored procedures should look and what data they should extract.

For more information regarding LiveCad, send an email to info@livecad.us

Create a stored procedure named **sp_LiveCadIncidents** This stored procedure should export the following fields in order with these field names:

1. **IncidentId**
 - a. Unique identifier for the incident. Use varchar() as type, even if stored in database as int()
2. **IncidentDate**
 - a. DateTime format for call received time in **UTC Time**
3. **Address**
 - a. Full street address of the incident (do **not** include City, State, or ZipCode)
4. **City**
 - a. If not known, just export an empty string
5. **State**
 - a. If not known, just export an empty string
6. **ZipCode**
 - a. If not known, just export an empty string
7. **County**
 - a. If not known, just export an empty string
8. **Nature**
 - a. Type of Call (Chest Pain, Traffic Accident, etc)
9. **Zone**
 - a. Specific area/zone for the incident (usually names of fire box zones, etc)
 - b. If not known, just export an empty string
 - c. Can also be combined with another field such as Radio Channel... to separate the 2 fields use a pipe symbol between them (eg. dbo.Incidents.Zone + ' | ' + dbo.Incidents.RadioChannel)
10. **Latitude**
 - a. Latitude in decimal degrees converted to string (eg. "40.030245")
 - b. If not known, just export '0'
11. **Longitude**
 - a. Longitude in decimal degrees converted to string (eg. "-82.394442")
 - b. If not known, just export '0'
12. **AgencyTag**
 - a. Usually an abbreviation of the dispatch center's name (eg. MECC, KCSO, etc) Limit to 4-5 characters

NOTES:

All fields (including Latitude and Longitude) should be exported as UPPER CASE strings except for the **IncidentDate** field.

To get the UTC time for the **IncidentDate** field, you can use (in Microsoft SQL Server):

DATEADD(hh, DATEDIFF(hh, GETDATE(), GETUTCDATE()), dbo.Incidents.CreationTime)

May need to convert **Latitude** and **Longitude** values from decimal to strings. To do that, one option is to create a function named **fn_ConvertFloatToString** (see end of document for code)

Create a stored procedure named **sp_LiveCadResources** with an optional parameter named **@IncidentId** that will export all Resource (truck) status information if no parameter is provided, or export all Resource information for a particular **IncidentId** if this parameter is provided. This stored procedure should export the following fields in order with these field names:

1. **DepartmentId**
 - a. Some kind of department identifier to identify which department this resource (truck) belongs to.
2. **ResourceId**
 - a. Name of the Resource (truck) E.g. M121, E123, L4
3. **ResourceType**
 - a. Type of Resource (“Medic”, “Engine”, etc)
 - b. If not known, just export an empty string
4. **ResourceStatus**
 - a. Current status of the resource (Eg. “Enroute”, “On Scene”, etc.) Preferred status names are:
 - DISPATCHED
 - ENROUTE
 - ON SCENE
 - TO HOSPITAL (or TO <HospitalAbbrev>, such as TO KCH, TO GRANT, TO WCH)
 - AT HOSPITAL (or AT <HospitalAbbrev>, such as AT KCH, AT GRANT, AT WCH)
 - IN SERVICE
 - IN QUARTERS
 - OUT OF SERVICE
 - ON AIR
 - b. If there are other statuses used, please submit a list so they can be color coded correctly
5. **IncidentId**
 - a. If the Resource is on a run, provide the **IncidentId** for that run
 - b. If the Resource is not on a run, just export an empty string

NOTES:

All fields should be exported UPPER CASE strings.

Here is an example of what logic the Resources stored procedure should have...

```
1  -- Stored Procedure for LiveCad - Resources
2  CREATE PROCEDURE [dbo].[sp_LiveCadResources]
3      @IncidentId varchar(30) = ''
4  AS
5  BEGIN
6      IF @IncidentId <> ''
7      BEGIN
8          SELECT *
9          FROM vwDN_LiveCadResources
10         WHERE (IncidentId = @IncidentId)
11     END
12 ELSE
13     BEGIN
14         SELECT *
15         FROM vwDN_LiveCadResources
16     END
17 END
```

Create a user name of **NIPST_LIVECAD_USERRO** with a password of **ReadOnly#1234** Give this username **EXEC** rights for the stored procedures above, and **SELECT** (read-only) access to the necessary fields that are contained within the stored procedures.

Example of **sp_LiveCadIncidents** stored procedure

IncidentId	IncidentDate	Address	City	State	ZipCode	County	Nature	Zone	Latitude	Longitude	AgencyTag
25FD120.2016-000794	2016-08-03 14:08:32.370	7525 WEST CAMPUS RD	NEW ALBANY	OH	43081	MECC	BREATHING PROBLEMS-EMS	121 E	0	0	MECC
25FD160.2016-002733	2016-08-03 14:18:38.927	1996 OPAL ST	REYNOLDSBURG	OH	43068	MECC	ILLNESS-EMS	162 E	0	0	MECC
25MECC.2016-006323	2016-08-03 14:39:30.657	3193 PENDLETON CT	COLUMBUS	OH	43219	MECC	ILLNESS-EMS	CFD 28	0	0	MECC
25FD150.2016-003602	2016-08-03 14:50:36.603	4590 E BROAD ST	WHITEHALL	OH	43213	MECC	UNCONSCIOUS PERSON-EMS	151 C	0	0	MECC

Example of **sp_LiveCadResources** stored procedure (no **IncidentId** parameter provided)

DepartmentId	ResourceId	ResourceType	ResourceStatus	IncidentId
25FD160	C160	C	OFF DUTY	
25FD160	SC166	SC	OFF DUTY	
25FD160	E162	E	IN QUARTERS	
25FD160	M162	M	ER MCE	25FD160.2016-002733
25FD160	E164	E	OFF DUTY	
25FD160	M164	M	OFF DUTY	
25FD160	B161	B	ON AIR	
25FD160	PREV162	PREV	OFF DUTY	
25FD160	SC162	SC	OFF DUTY	
25FD160	PREV161	PREV	OFF DUTY	
45FD460	C461	C	IN QUARTERS	
45FD460	C462	C	IN QUARTERS	
45FD460	CMND461	CMND	IN QUARTERS	
45FD460	E462	E	IN QUARTERS	
45FD460	E463	E	IN QUARTERS	
45FD460	GF467	GF	IN QUARTERS	

Example of **sp_LiveCadResources** stored procedure (with **IncidentId** parameter provided)

DepartmentId	ResourceId	ResourceType	ResourceStatus	IncidentId
25FD150	E151	E	DISPATCHED	25FD150.2016-003602
25FD150	B151	B	ENROUTE	25FD150.2016-003602
25FD150	M151	M	ENROUTE	25FD150.2016-003602

Function for converting floating point decimal numbers into strings:

```
-- =====
-- Description: Convert a float to string
--           Made specifically for converting Latitude/Longitude values to strings
-- =====
CREATE FUNCTION [dbo].[fn_ConvertFloatToString]
(
    @p1 float
)
RETURNS varchar(40)
AS
BEGIN
    DECLARE @Result varchar(30) = '0'

    If @p1 = 0 OR @p1 > 180.0 OR @p1 < -180.0
        Set @Result = '0'
    Else
        Set @Result = Cast(Cast(@p1 as decimal(20,6)) as varchar(30))

    RETURN @Result
END
```