



## **Database Tool for Master Console Operators**

Sean Ferrell

NE-XC

Major: Mechanical Engineering

Spring Session

Date: 05.04.2018

# Database Tool for Master Console Operators

Sean P Ferrell

University Of Central Florida, Orlando, Florida, 32816

**The Spaceport Command and Control System (SCCS) is the National Aeronautics and Space Administration's (NASA) launch control system for the Orion capsule and Space Launch System, the next generation manned rocket currently in development. This large system requires highly trained and knowledgeable personnel. Master Console Operators (MCO) are currently working on familiarizing themselves with any possible scenario that they may encounter. An intern was recruited to help assist them with creating a tool to use for the process.**

## Nomenclature

<i>CAIDA</i>	=	Customer Avionics Interface Development and Analysis
<i>EMI</i>	=	Exploration Mission 1
<i>FR1/2/3/4</i>	=	Firing Room 1,2,3,4
<i>GSDO</i>	=	Ground Systems Development and Operations
<i>KSC</i>	=	Kennedy Space Center
<i>LCC</i>	=	Launch Control Center
<i>LCS</i>	=	Launch Control System
<i>MCO</i>	=	Master Console Operator
<i>OMU</i>	=	Operational Manager for UNIX
<i>SCCS</i>	=	Spaceport Command and Control System
<i>SLS</i>	=	Space Launch System
<i>VBA</i>	=	Visual Basic for Applications

## I. Introduction

**T**he purpose of this internship was to provide the MCOs with a searchable database for them to use when troubleshooting the OMU messages they receive every day. Before when troubleshooting, the MCOs that don't have the knowledge that the veteran MCOs would have to seek guidance from the veteran MCOs. Since the MCOs receive thousands of OMU messages every day this has the potential to become a great hindrance on the work that they accomplish throughout the day. Now, with the newly created database of messages they no longer have to go to the veteran MCOs. Now they can enter the message into the database and define an action one time to reference in the future.

## II. Objectives

The main objective of this project was to create a database for the MCOs to use when troubleshooting error messages they receive on their consoles. The database has a record of the current user logged onto the computer, as well as relevant information (for example, the date, any attachments, the message text, exc.)

## III. Approach

My approach to the designated assignments was to first focus on learning how to create a database and acquire access to perform my duties in the LCC. To do this my mentor Celia, and I would frequently meet to answer and explain the procedures, tasks, and goals of this project. While doing so she explained the duties of the MCOs and why this database is needed.

**A. Training/Familiarization**

*a. COTS Application*

Understanding the COTS (commercial off-the-shelf) application with all of its functionality would take several months, since I was not using it for my project but still needed to know some information about it, so I met with the MCOs to get an overview of the information I would need. They explained to me why certain messages have a higher priority than others and what they do when they look at them. For instance when they receive a green-highlighted message they do not take action because green is considered a “Normal” message. However if they were to receive a red-highlighted message (Emergency Message) they would take some action to respond to the error.

*b. COTS Database*

A COTS database was picked as the platform for this project. This decision was made because of the availability of the product across NASA. Having never used it or any other database tools before made the task more challenging. I learned and am still learning much about how to design and create databases. Since I came in with no knowledge on creating databases I worked with Bret Sorensen, a Ground Data Systems expert. Bret showed me the utility and power that a database has for almost any application.

After he taught me, Bret gave me some resources for learning the COTS product. I began to make my own database containing information about superheroes. I learned about foreign keys, primary keys, queries, tables, forms, joins, indexes, and how to normalize a database.

*c. Database Development Application*

To ensure some compatibility and implementation to other platforms that could be introduced in the future, A common database development tool was utilized for this project. This enables user-defined functions, automating processes and accessing API’s and other low level functionality. The chosen database inherently has these functionalities built in without the need for any code. The purpose for the code is to implement the same functionality seamlessly in the future without requiring additional effort.

```

Option Compare Database
Option Explicit

.....
'For WIN95/WIN NT / (WFWG?) running Microsoft Networking
.....

Private gIngrReturnStatus As Long
Private Const Success = 1&

Declare Function ADV_GetUserName Lib "advapi32.dll" Alias "GetUserNameA" (ByVal struser As String, lngBuffer As Long) As Long

Public Function Networkuserid() As String
Dim lngBufferSize As Long
Dim struser As String

On Error GoTo NetworkUserID_EH

Networkuserid = "UnknownUser"

lngBufferSize = 255
struser = Space$(lngBufferSize)

gIngrReturnStatus = ADV_GetUserName(struser, lngBufferSize)
If gIngrReturnStatus = Success Then
struser = Left$(struser, lngBufferSize - 1)
Else
Err = gIngrReturnStatus
End If

Networkuserid = If(Len(struser) <> 255, struser, "Unknown")

Exit Function

NetworkUserID_EH:
Networkuserid = "NetworkUserID_Error"
Exit Function
End Function
    
```

**Figure 1:** Database Development user name call code

## **B. Message Analysis**

The objective of the database is to have executable actions for error messages available at the MCO's disposal. With the MCOs receiving thousands of messages a day, this poses an issue when attempting to select the duplicated messages out of the list for implementation to the database.

A script was suggested by Bret to be utilized but ruled out due to the duplicate messages having too much variation between their respective texts. The solution was to manually go through and sort out the duplicates. I was presented with 5 Excel spreadsheets each having over 20,000 OMU messages to sort through. Going through only the caution and warning messages I found there to only be 39 unique messages out of the original 100,000 plus initial data points.

## **C. Documents and Agreements**

### *a. Requirements Document*

To have a clear understanding of what is required of the database, a requirements document was written up by me. To document the functional requirements the database will perform when implemented, the requirements document contains shall statements i.e. "The Date Created fields shall be entered by the application upon initially creating the task by selecting the save button" to convey the requirements.

This document defines the Overview, Scope, Users, Requirements (General System, Roles, System Administration, Task, and Reports), and Constraints.

### *b. Users Guide*

A User Guide document was created to showcase the use of the product. This User Guide includes snapshots of the document with steps and highlights on what to do when utilizing the database. A sniping tool was used to extract and modify the photos onto the Word document.

### *c. Test Document*

To evaluate if the product could be utilized properly a set of steps would need to be tested by the MCO's and by the MCOs and engineers in my branch. These steps are defined in a test document. This document will define what result is expected from the test procedural step and indicate the conditions under which it passes or fails.

## **IV. Conclusion**

At the time of this paper, The MCO's are receiving many new messages with each software delivery. The database is ready to receive thousands of messages input to its message bank. These messages will have to be input and have actions added to them by the MCO's. The NE-XC team has made excellent progress on the framework of the database. For the time being, the database is not being further developed and will only be used as initially intended. If any further changes are required to be made in the future the NE-XC team will be able to support them. It has been a privilege to support the team in preparing for the most historically significant launch since Apollo and Shuttle.

## **Acknowledgments**

There is a plethora of wonderful colleagues that need to be thanked for the progress over my internship. First I would like to thank Charles Jenkins for being a wonderful supervisor and allowing me a flexible schedule. I would like to thank Jamie Szafran, Jill Giles, and Oscar Brooks for supporting the intern tag-ups and the assistance on my resume. Bret Sorensen for being the most knowledgeable software person ever and invaluable advice on automobiles. Allan Villorin and Kelvin Ruiz for all escorts, laughs and friendship. Last, but certainly not least, I would like to thank Celia Brandt for having me as her first intern, it was a wonderful experience and the most enjoyable job I have ever had!

## **References**

N/A

