

Data Acquisition Standard Operating Procedures

EPA STOrage and RETrieval Data Warehouse (STORET) (ID# 103)

Last Updated: 5/6/2023

Program Summary

Water quality data repository This is not a monitoring program but a database of data from monitoring programs. EPA's repository of the water quality monitoring data collected by water resource management groups across the country. Includes water quality, chemical parameter measurements made in the field or lab, physical parameter measurements such as water temperature or pH, biological sampling results, and habitat assessments or observations.

URLs

- Program - <http://www.epa.gov/STORET/dbtop.html>
- DDI - <https://data.florida-seacar.org/programs/details/103>

Contacts

Contact Name	Organization	Email	Phone
--------------	--------------	-------	-------

Data Tables

- Data_103A_Final
- Data_103A_Load

Data Stored Procedures

- usp_Data_103A_Load_insert
- usp_combined_wq_wc_nut_insert_103A

Data Acquisition Standard Operating Procedures: ProgramID 103

Date Created: 02/04/2019

Created By: *Claude Kershaw*

Date Modified: 04/26/2022

Modified By: *Jennifer Baker*

Data File Path:

1. "\\forest.usf.edu\data\PDrive\CAS-WI\Misc Projects\SEACAR_FDEP\Data\ID_0103_EPA_STORET\DataToLoad\result_OrgsNotInFLSTORET.txt"
2. Spatial Data: [\\forest.usf.edu\data\PDrive\CAS-WI\Misc Projects\SEACAR_FDEP\Data\ID_0103_EPA_STORET\DataToLoad\EPA_Stations_1feb19.txt](#)
3. U:\Misc_Projects\SEACAR_FDEP\Data\ID_0103_EPA_STORET\DataToLoad\ result.txt
4. Locations Data: U:\Misc_Projects\SEACAR_FDEP\Data\ID_0103_EPA_STORET\DataToLoad\ station.txt
5. Sample Data: U:\Misc_Projects\SEACAR_FDEP\Data\ID_0103_EPA_STORET\DataToLoad\result (1)\ result.tx
6. Locations Data:
U:\Misc_Projects\SEACAR_FDEP\Data\ID_0103_EPA_STORET\DataToLoad\station (1)\station.txt

DDI URL: <http://dev.seacar.waterinstitute.usf.edu/datadiscovery/programs/details/103>

Procedure Overview:

1. Use SQL Server Import Export Wizard to load the file "EPA_Stations_1feb19.txt" into table **Locations_103A**.
2. Use SQL Server Import Export Wizard to load the file "station.txt" into table **Locations_103A**.
3. Use SQL Server Import Export Wizard to load the file "result_OrgsNotInFLSTORET.txt" into table **Data_103A_Load**.
4. Use SQL Server Import Export Wizard to load the file "result.txt" into table **Data_103A_Load**.
5. Execute procedure usp_Data_103A_Load_insert to load the data into table **Data_103A_Final**.
6. Add the Monitoring Locations from tables **Locations_103A** to the **SampleLocation_Point** table if they do not exist there already.
7. Add new Monitoring Locations into the **SampleLocation** table. This will generate a LocationID for each Monitoring Location.
8. Update the **SampleLocation_Point** table with the LocationID generated in the **SampleLocation** table. Run procedure usp_SampleLocation_Point_update to do this.
9. Update the LocationID column in table **Data_103A_Final** with the LocationID in the **SampleLocation** table. Join on the [MonitoringLocationIdentifier] column in **Data_103A_Final** and the ProgramLocationID column in **SampleLocation**.
10. Update the **Data_103A_Final** table with the locationIDs from the **samplelocation** table

11. Combine the data into the **combined_wq_wc_nut** table using **usp_combined_wq_wc_nut_insert_103A**

Data Tables

1. Data_103A_Load
2. Data_103A_Final

Data Stored Procedures

1. usp_Data_103A_Load_insert
2. exec usp_SampleLocation_update @ProgramID = 103
3. usp_combined_wq_wc_nut_insert_103A

GIS Procedures

1. The Monitoring Location information is found in the table **Locations_103A**.
2. Complete steps 6 through 9 in the "Procedure Overview" section of this document.

```
SET ANSI_NULLS ON
SET QUOTED_IDENTIFIER ON
```

```
CREATE PROC [dbo].[usp_combined_wq_wc_nut_insert_103A]
```

```
AS
BEGIN
SET NOCOUNT ON;
SET XACT_ABORT ON;
```

```
-- Constants - PLEASE SET NOW!!
DECLARE @combinedTable varchar(50) = 'Combined_WQ_WC_NUT'
DECLARE @dataLoadCode varchar(10)
```

```
-- Temporary
SET @dataLoadCode = '103A'
```

```
-- Setup data load
DECLARE @runBy varchar(50) = SYSTEM_USER;
DECLARE @programID varchar(10);
DECLARE @dataStreamID varchar(10);
```

```
SELECT @dataStreamID = DataStreamID,
@programID = ProgramID
FROM DataStreamProcedure
WHERE DataLoadCode = @dataLoadCode;
```

```
--delete existing data
exec usp_delete_combined @datastreamID, 'Combined_WQ_WC_NUT'
```

```

INSERT INTO
Combined_WQ_WC_NUT(ProgramID, DataStreamID, ParameterID, [LocationID], [SampleDate], [ActivityDepth_m],
epth_m, [RelativeDepth], ResultValue, DateAdded)
SELECT DISTINCT @programID, @dataStreamID, c.ParameterID, LocationID, a.ActivityStartDate, null if (case
when a.ActivityDepthHeightUnitCode is not null and a.ActivityDepthHeightUnitCode <> 'm' then
dbo.udf_convert_units(a.ActivityDepthHeightUnitCode, 'm', a.ActivityDepthHeightValue) ELSE
a.ActivityDepthHeightValue end, ''), null if (case when d.ResultMeasureUnitCode is not null and
d.ResultMeasureUnitCode <> 'm' then dbo.udf_convert_units(d.ResultMeasureUnitCode, 'm',
d.ResultMeasureValue) else d.ResultMeasureValue end, ''), NULL, dbo.udf_convert_units(b.OriginalUnits,
c.Units, CAST(a.ResultMeasureValue as numeric(25, 8))), getdate()
FROM Data_103A_Final
INNER JOIN Combined_Conversion_Parameters b ON LTRIM(RTRIM(a.CharacteristicName)) =
b.OriginalParameter and LTRIM(RTRIM(a.ResultMeasureUnitCode)) = b.OriginalUnits AND b.datastreamID =
141
INNER JOIN Combined_Parameters c ON b.TargetParameterID = c.ParameterID
LEFT JOIN (SELECT ActivityIdentifier, ResultMeasureValue, ResultMeasureUnitCode
FROM Data_103A_Final
WHERE CharacteristicName = 'Depth, bottom') d ON a.ActivityIdentifier = d.ActivityIdentifier
WHERE ISNUMERIC(a.ResultMeasureValue + 'e0') = 1
AND a.ActivityStartDate is not null
AND a.LocationID is not null
```

```
INSERT INTO Combined_Data_Tracking (ProgramID, IndicatorID, DataStreamID, ParameterID,
CombinedTableName, NumRowsFinal, NumRowsCombined, LastUpdateDate, LastUpdateBy)
SELECT @ProgramID, b.IndicatorID, a.DataStreamID, a.ParameterID, @CombinedTable,
COUNT(a.ResultValue), COUNT(a.ResultValue), GETDATE(), @runBy
FROM Combined_WQ_WC_NUTa
INNER JOIN Combined_Parameters b on a.ParameterID = b.ParameterID
WHERE a.ProgramID = @ProgramID
AND a.DataStreamID = @dataStreamID
GROUP BY a.ProgramID, b.IndicatorID, a.DataStreamID, a.ParameterID
```

```
/*
SELECT *
FROM Combined_WQ_WC_NUT
```

```
SELECT *
FROM Data_103A_Final
```

```
SELECT *
FROM Combined_Parameters a
join Indicator b on a.IndicatorID = b.IndicatorID
where b.Habitat = 'Water Column'
and b.IndicatorName <> 'Nekton'
```

```
SELECT *
FROM DataStreamProcedure
WHERE ProgramID = 103
```

```
SELECT a.*
FROM Combined_Parameters a
join Indicator b on a.IndicatorID = b.IndicatorID
where b.IndicatorName IN ('Water Quality', 'Water Clarity', 'Nutrients')
```

```
SELECT *
FROM Combined_Data_tracking
where programid = 103
exec usp_delete_combined 141, 'Combined_WQ_WC_NUT'
*/
```

```
END
```

```
GO
```

```
SET ANSI_NULLS ON
SET QUOTED_IDENTIFIER ON
CREATE PROC usp_Data_103A_Load_insert
AS
BEGIN
SET NOCOUNT ON
SET XACT_ABORT ON

INSERT INTO Data_103A_Final (OrganizationalIdentifier, OrganizationalFormalName, ActivityIdentifier,
ActivityTypeCode, ActivityMediaName, ActivityMediaSubdivisionName, ActivityStartDate,
ActivityStartTime, ActivityStartTimeZoneCode, ActivityEndDate, ActivityEndTime,
ActivityEndTimeZoneCode, ActivityDepthHeightValue, ActivityDepthHeightUnitCode,
ActivityDepthAltitudeRefPoint, ActivityTopDepthHeightValue, ActivityTopDepthHeightUnitCode,
ActivityBottomDepthHeightValue, ActivityBottomDepthHeightUnitCode, ProjectIdentifier,
ActivityConductingOrgText, MonitoringLocationIdentifier, ActivityCommentText, SampleAquifer,
HydrologicCondition, HydrologicEvent, SampleCollectionMethodId, SampleCollectionMethodIdContext,
SampleCollectionMethodName, SampleCollectionEquipmentName, ResultDetectionConditionText,
CharacteristicName, ResultSampleFractionText, ResultMeasureValue, ResultMeasureUnitCode,
MeasureQualifierCode, ResultStatusIdentifier, StatisticalBaseCode, ResultValueTypeName,
ResultWeightBasisText, ResultTimeBasisText, ResultTemperatureBasisText,
ResultParticleSizeBasisText, PrecisionValue, ResultCommentText, USGSPCode,
ResultDepthHeightMeasureValue, ResultDepthHeightMeasureUnitCode, ResultDepthAltitudeRefPointText,
SubjectTaxonomicName, SampleIssueAnatomyName, ResultAnalyticalMethodId,
ResultAnalyticalMethodIdContext, ResultAnalyticalMethodName, MethodDescriptionText,
LaboratoryName, AnalysisStartDate, ResultLaboratoryCommentText, DetectionQuantLimitTypeName,
DetectionQuantLimitValue, DetectionQuantLimitUnitCode, PreparationStartDate, ProviderName)
SELECTOrganizationalIdentifier, OrganizationalFormalName, ActivityIdentifier,
ActivityTypeCode, ActivityMediaName, ActivityMediaSubdivisionName, ActivityStartDate,
ActivityStartTime, ActivityStartTimeZoneCode, ActivityEndDate, ActivityEndTime,
ActivityEndTimeZoneCode, ActivityDepthHeightValue, ActivityDepthHeightUnitCode,
ActivityDepthAltitudeRefPoint, ActivityTopDepthHeightValue, ActivityTopDepthHeightUnitCode,
ActivityBottomDepthHeightValue, ActivityBottomDepthHeightUnitCode, ProjectIdentifier,
ActivityConductingOrgText, MonitoringLocationIdentifier, ActivityCommentText, SampleAquifer,
HydrologicCondition, HydrologicEvent, SampleCollectionMethodId, SampleCollectionMethodIdContext,
SampleCollectionMethodName, SampleCollectionEquipmentName, ResultDetectionConditionText,
CharacteristicName, ResultSampleFractionText, ResultMeasureValue, ResultMeasureUnitCode,
MeasureQualifierCode, ResultStatusIdentifier, StatisticalBaseCode, ResultValueTypeName,
ResultWeightBasisText, ResultTimeBasisText, ResultTemperatureBasisText,
ResultParticleSizeBasisText, PrecisionValue, ResultCommentText, USGSPCode,
ResultDepthHeightMeasureValue, ResultDepthHeightMeasureUnitCode, ResultDepthAltitudeRefPointText,
SubjectTaxonomicName, SampleIssueAnatomyName, ResultAnalyticalMethodId,
ResultAnalyticalMethodIdContext, ResultAnalyticalMethodName, MethodDescriptionText,
LaboratoryName, AnalysisStartDate, ResultLaboratoryCommentText, DetectionQuantLimitTypeName,
DetectionQuantLimitValue, DetectionQuantLimitUnitCode, PreparationStartDate, ProviderName
FROMData_103A_Loada
END
GO
```