***Assessing the Effectiveness of Information Operations in the European Theater***

Ian Morris

IST 659

29 Mar 2020

**Summary**

The United States European Command (USEUCOM) is a major United States military command responsible for deterring conflict, countering transnational threats and advancing the United States Government’s (USG) interests throughout Europe and its periphery. Beginning in 2019, the commander of USEUCOM has noticed a marked increase in adversarial disinformation activities, also called Information Operations (IO). Adversary IOs are principally targeted toward the weakening of the North Atlantic Treaty Organization (NATO) Alliance, erosion of USG influence and promotion of a counter-democratic world-view.

The USEUCOM Commander is struggling to implement a strategy to counter the adversaries’ IO due to constraints placed on their currently allocated forces. The Commander has requested the assignment of an Information Operations Command (IOC) to develop courses of action, conduct operations and assess the effectiveness of those operations. In order to do this, the IOC must first develop a database that encompasses baseline data about the European Theater, target audiences and stakeholders; organize and assign IO objectives and tasks to an implementing unit; and measure the accomplishment of tasks and assess their effectiveness. Through this method, and along with ongoing military USG efforts, the commander of USEUCOM believes they will be able to reassure USG allies and partners of further US commitment to European Security and thwart adversaries from promoting European instability and division.

**Stakeholders**

The Stakeholders for this project are: Commander of USEURCOM, Commander of Information Operations Command and the implementing subordinate units.

**Glossary of Terms**

**United States European Command** – Major US military organization with several 10s of thousands of assigned personnel, 5 subordinate commands and responsibility for 51 European countries.

**Unit** – organization of varying personnel sizes

**Commander** – senior leader of an organization responsible for everything that organization does or fails to do.

**USG Objectives** – priority outcomes for the United States Government.

**Commander Objectives** – Priority outcomes for Commanders

**Information Operations Objectives** – Priority outcomes for IO commanders in support of the Commander’s objectives

**Disinformation Activities** – activities taken by an entity to knowingly propagate false or misleading information with the malicious intent to alter decision making or perception.

**Information Operations** – Activities conducted in and through the information environment to influence an entity and alter its decision-making process

**Adversary** – an entity with objectives that conflict with those of the United States Government.

**Forces** – permanently (regionally) assigned military units.

**Information Operations Command (IOC)** – an (fictional) organization with multiple sub-units that possess specialties necessary to conduct information operations in varying forms.

**Target Audience** – entity (friendly, neutral or adversarial) at which information operations are directed.

**Operations / Task** – activities or actions executed to achieve a specific purpose or objective.

**Effectiveness** – a criterion to assess changes in system behavior, capability or operational environment.

**Implementing Unit** – a sub-unit with the capacity to conduct a specific information operation or task

**Supports** means that an objective or task is inherently achieving the objective or task assigned by the next higher echelon. A IO task must always support an IO objective and transitively support a Commander’s objective.

**Develops** means that an entity is creating and working towards the achievement of the linked task or objective.

**Importance** is a subjective term denoting to which varying magnitude an activity directly supports a higher-level objective.

A **Category** is a multivariable attribute that assigns a task or objective to a predetermined classification.

**Business Rules**

A Commander objective must support USG Objectives

IO Objectives must support CDR Objectives

IO Tasks must support IO Objectives

An Implementing Unit can have many skills

An Operation must have an implementing unit, target audience and IO task.

A Target Audience may have one location

A Target Audience must have at least 1 belief, motivation and grievances

An Implementing Unit Assesses efficacy of each task

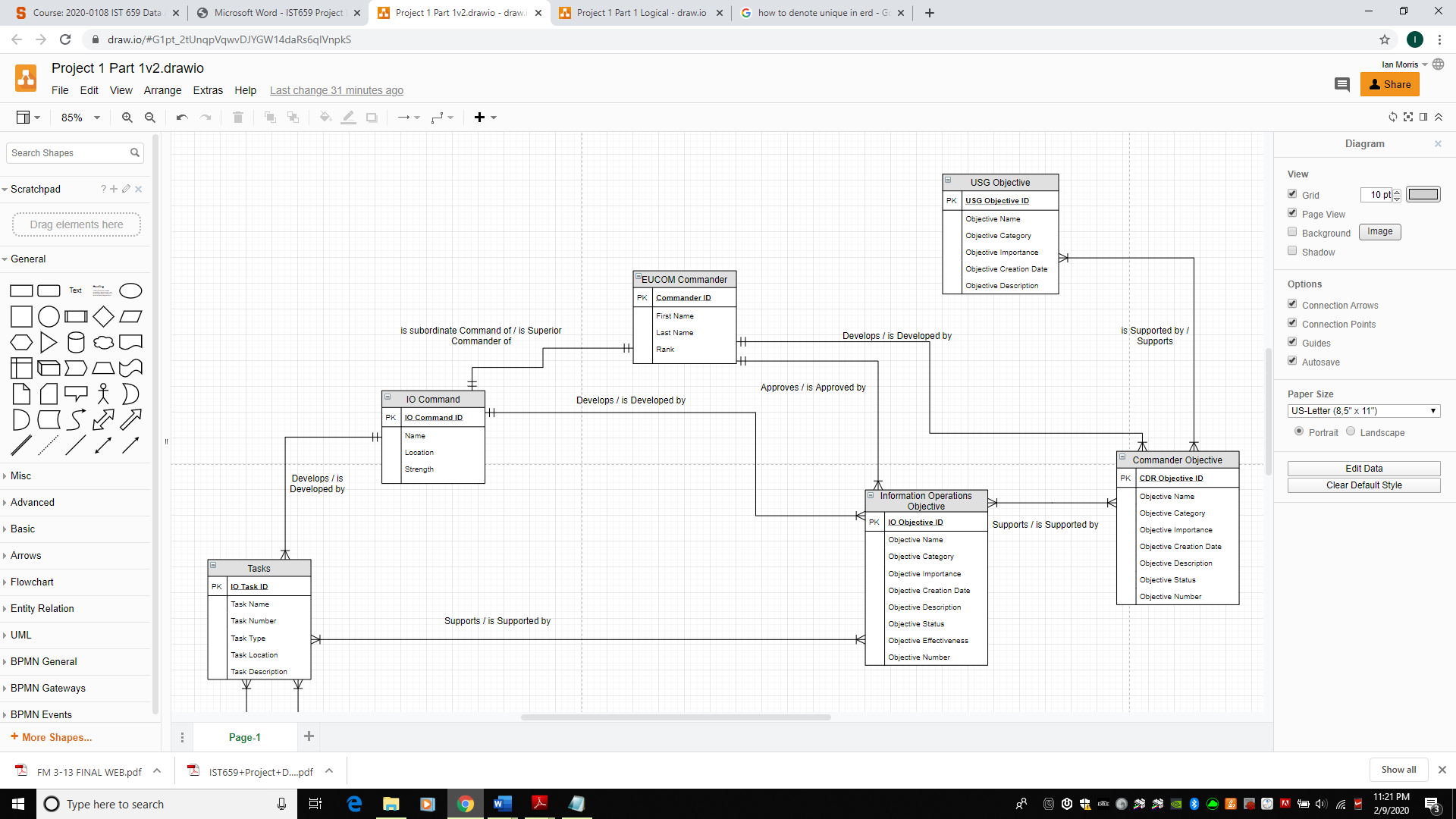
**Business Questions (answered at end)**

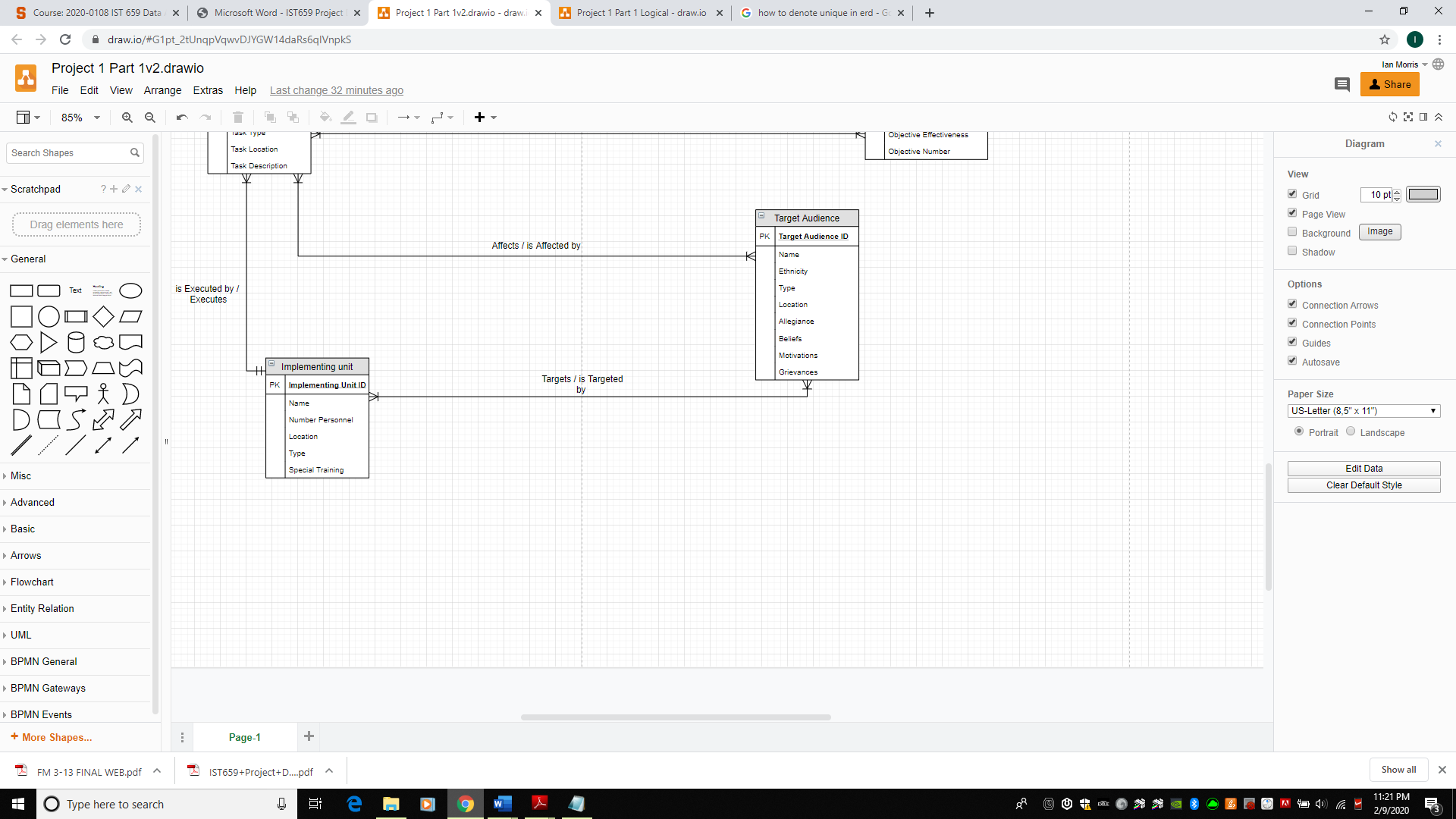
1. Which Countries are most receptive to USG Information Operations?
2. What socio-economic, diplomatic and developmental conditions are necessary for USG IO to thrive in a particular country?
3. Which type of information related capability is having the most general success and to what can we attribute this success?
4. Are our priority information operations objectives sufficiently supporting the commander’s priority objectives?

**Expected Outcome**

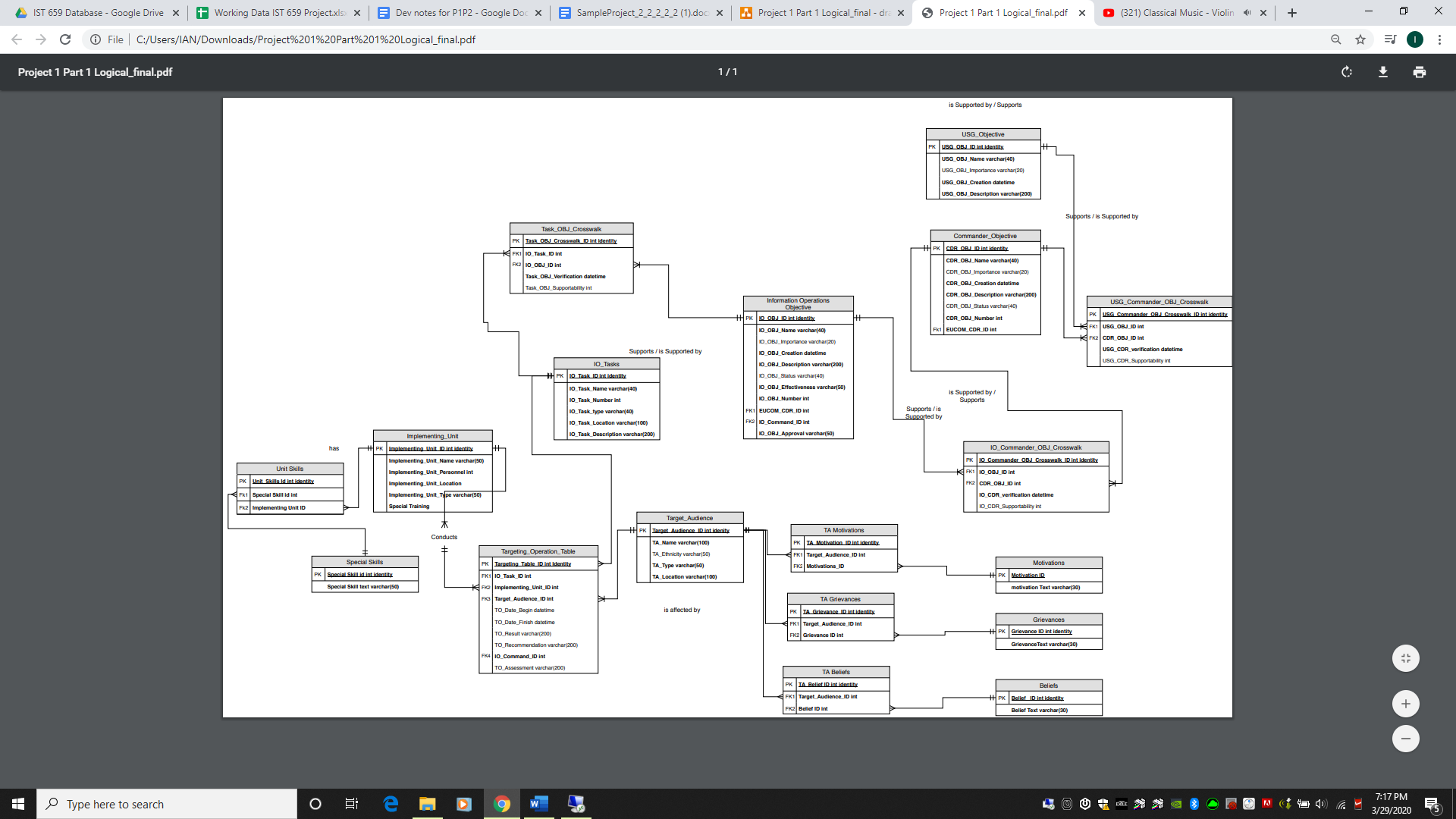
With this project we intend to inform senior leadership of precisely how their objectives are being accomplished by the activities of subordinates on the ground. Due to the political sensitivities of this work, it is essential that leaders at all levels ensure that junior leaders are executing tasks that fall within an acceptable set of constraints indicated by senior military and civilian officials. This database will enable leaders to assess iterative activities over a given period of time to determine true changes of behavior in target audiences and adversaries.

**Conceptual ERD**





**Logical ERD**



**Raw Data Sample**

This information was documented by one particular implementing team and submitted via-email to their immediate supervisor, located at their continental United States duty station. From there, the information was submitted through the United States Army Forces Command hierarchy before it was transferred to the United States European Command Staff for processing. This is a lengthy and inefficient process that we want to eliminate.

Unit: Civil Affairs Team 495

Home Station: Fort Bragg, North Carolina

Team Type: Civil Affairs

Location of Activity: Tallinn, Estonia

Date Began: 2 FEB 2020

Date Concluded: 10 FEB 2020

Type of Activity: Deliberate Assessment

Partnered Group: Estonia Group 1

Summary of Activity: CAT 495 was contacted by Estonia Group 1 through the US-Embassy Tallinn on January 27th, 2019 to conduct an assessment of a religious primary school inside the city. EST Group 1 is a **conservative**, **religious group that** has been in contact with Embassy Tallinn regarding persecution and **violence** conducted against them by rival religious groups. They are concerned about **self-preservation** and believe that the Estonian government is unable or unwilling to support their needs and provide adequate protection due to external influence, lobbying and blackmail against Estonian officials.

CAT 495 conducted a deliberate structural assessment of the religious primary school and had lengthy discussions with various stakeholders within the EST Group 1 community. Due to the extensive presence of police and other government officials, CAT 495 believes that EST Group 1 was unable to speak at length about their grievances and support requirements. Due to this fact, CAT 495 assesses that the operation was moderately effective and recommends that future teams continue to engage with the community.

**Issues with Raw Data Sample**

This report is characteristic of what a typical information related capability or “implementing unit” might submit when trying to summarize an individual or series of operations. Though this particular submission is a rather scarce report, it is still full of information of value to planners and decision makers at a higher echelon. The “free flow” writing style allows the submitting entity to include information that they believe is relevant to the mission at hand.

The primary issue with a report of this kind and the submission process is that the free-flow summary style requires that an analyst or another staff member must sift through the report and pull the relevant information out of it for further review. We have highlighted some of the key words in this report that an information manager would need to associate with the entities of Target Audience, such as grievances, location, motivations and type of activity or operation. The intent of this database is that these keywords will be submission fields on the GUI, which will allow teams to make keyword submissions from a drop-down menu from a phone or other device while they are on the move. Instead of the convoluted hierarchy that exists now, once submitted, information will be updated in the database to allow analysts and other planners immediate access.

**Physical Database Design**

--Drop Tables

IF OBJECT\_ID('dbo.ApproveCDRStatus') IS NOT NULL

DROP PROCEDURE dbo.ApproveCDRStatus

IF OBJECT\_ID('dbo.Mil\_Operations') IS NOT NULL

DROP TABLE dbo.Mil\_Operations

IF OBJECT\_ID('dbo.Mil\_IOOBJ\_Task\_XWalk') IS NOT NULL

DROP TABLE dbo.Mil\_IOOBJ\_Task\_XWalk

IF OBJECT\_ID('dbo.Mil\_CDR\_IO\_OBJ\_Xwalk') IS NOT NULL

DROP TABLE dbo.Mil\_CDR\_IO\_OBJ\_XWalk

IF OBJECT\_ID('dbo.Mil\_CDR\_USG\_XWalk') IS NOT NULL

DROP TABLE dbo.Mil\_CDR\_USG\_XWalk

IF OBJECT\_ID('dbo.Mil\_TA\_Grievances') IS NOT NULL

DROP TABLE dbo.Mil\_TA\_Grievances

IF OBJECT\_ID('dbo.Mil\_TA\_Motivations') IS NOT NULL

DROP TABLE dbo.Mil\_TA\_Motivations

IF OBJECT\_ID('dbo.Mil\_TA\_Beliefs') IS NOT NULL

DROP TABLE dbo.Mil\_TA\_Beliefs

IF OBJECT\_ID('dbo.Mil\_Unit\_Skill') IS NOT NULL

DROP TABLE dbo.Mil\_Unit\_Skill

IF OBJECT\_ID('dbo.Mil\_IO\_Task') IS NOT NULL

DROP TABLE dbo.MIL\_IO\_Task

IF OBJECT\_ID('dbo.Mil\_IO\_OBJ') IS NOT NULL

DROP TABLE dbo.MIL\_IO\_OBJ

IF OBJECT\_ID('dbo.Mil\_CDR\_OBJ') IS NOT NULL

DROP TABLE dbo.MIL\_CDR\_OBJ

IF OBJECT\_ID('dbo.Mil\_USG\_OBJ') IS NOT NULL

DROP TABLE dbo.MIL\_USG\_OBJ

IF OBJECT\_ID('dbo.Mil\_Grievances') IS NOT NULL

DROP TABLE dbo.Mil\_Grievances

IF OBJECT\_ID('dbo.Mil\_Motivations') IS NOT NULL

DROP TABLE dbo.Mil\_Motivations

IF OBJECT\_ID('dbo.Mil\_Beliefs') IS NOT NULL

DROP TABLE dbo.Mil\_Beliefs

IF OBJECT\_ID('dbo.Mil\_Target\_Audience') IS NOT NULL

DROP TABLE dbo.Mil\_Target\_Audience

IF OBJECT\_ID('dbo.Mil\_Special\_Skill') IS NOT NULL

DROP TABLE dbo.Mil\_Special\_Skill

IF OBJECT\_ID('dbo.Mil\_Implementing\_Unit') IS NOT NULL

DROP TABLE dbo.Mil\_Implementing\_Unit

--Creating the Database Tables, respective columns and constraints

CREATE TABLE Mil\_Implementing\_Unit (

Implementing\_Unit\_ID int identity

, Implementing\_Unit\_Name varchar(20) NOT NULL

, Implementing\_Unit\_Personnel int

, Implementing\_Unit\_Location varchar(30)

, Implementing\_Unit\_Type varchar(50)

, CONSTRAINT PK\_Mil\_Implementing\_Unit PRIMARY KEY (Implementing\_Unit\_ID)

)

CREATE TABLE Mil\_Special\_Skill (

Special\_Skill\_ID int identity

, Special\_Skill\_Text varchar(50) NOT NULL

, CONSTRAINT PK\_Mil\_Special\_Skill PRIMARY KEY (Special\_Skill\_ID)

, CONSTRAINT U1\_Mil\_Special\_Skill UNIQUE (Special\_Skill\_Text)

)

--Test Newly Created Tables

SELECT \* FROM dbo.Mil\_Implementing\_Unit

SELECT \* FROM dbo.Mil\_Special\_Skill

--Continue to Create Tables

CREATE TABLE Mil\_Target\_Audience (

Target\_Audience\_ID int identity

, Target\_Audience\_Name varchar(50) NOT NULL

, Target\_Audience\_Ethnicity varchar(50) NOT NULL

, Target\_Audience\_Type varchar(50)

, Target\_Audience\_Location varchar(50)

, CONSTRAINT PK\_Mil\_Target\_Audience PRIMARY KEY (Target\_Audience\_ID)

, CONSTRAINT U1\_Mil\_Target\_Audience UNIQUE (Target\_Audience\_Name)

)

CREATE TABLE Mil\_Beliefs (

Belief\_ID int identity

, Belief\_Text varchar(50) NOT NULL

, CONSTRAINT PK\_Mil\_Beliefs PRIMARY KEY (Belief\_Id)

, CONSTRAINT U1\_Mil\_Beliefs UNIQUE (Belief\_Text)

)

CREATE TABLE Mil\_Motivations (

Motivation\_ID int identity

, Motivation\_Text varchar(30) NOT NULL

, CONSTRAINT PK\_Mil\_Motivations PRIMARY KEY (Motivation\_Id)

, CONSTRAINT U1\_Mil\_Motivations UNIQUE (Motivation\_Text)

)

CREATE TABLE Mil\_Grievances (

Grievance\_Id int identity

, Grievance\_Text varchar(30) NOT NULL

, CONSTRAINT PK\_Mil\_Grievances PRIMARY KEY (Grievance\_ID)

, CONSTRAINT U1\_Mil\_Grievances UNIQUE (Grievance\_Text)

)

CREATE TABLE Mil\_USG\_OBJ (

USG\_OBJ\_ID int identity

, USG\_OBJ\_Name varchar (200) NOT NULL

, USG\_OBJ\_Est datetime

, USG\_OBJ\_Priority int

, CONSTRAINT PK\_Mil\_USG\_OBJ PRIMARY KEY (USG\_OBJ\_ID)

, CONSTRAINT U1\_Mil\_USG\_OBJ UNIQUE (USG\_OBJ\_Name)

)

CREATE TABLE Mil\_CDR\_OBJ (

CDR\_OBJ\_ID int identity

, CDR\_OBJ\_Number int NOT NULL

, CDR\_OBJ\_Name varchar (200) NOT NULL

, CDR\_OBJ\_Est datetime

, CDR\_OBJ\_Priority int

, CDR\_OBJ\_Status varchar(30)

, CONSTRAINT PK\_Mil\_CDR\_OBJ PRIMARY KEY (CDR\_OBJ\_ID)

, CONSTRAINT U1\_Mil\_CDR\_OBJ UNIQUE (CDR\_OBJ\_Name)

, CONSTRAINT U2\_Mil\_CDR\_OBJ UNIQUE (CDR\_OBJ\_Number)

)

CREATE TABLE Mil\_IO\_OBJ (

IO\_OBJ\_ID int identity

, IO\_OBJ\_Number int NOT NULL

, IO\_OBJ\_Name varchar (200) NOT NULL

, IO\_OBJ\_Est datetime

, IO\_OBJ\_Priority int

, IO\_OBJ\_Status varchar(30)

, IO\_OBJ\_Approval varchar(30)

, CONSTRAINT PK\_Mil\_IO\_OBJ PRIMARY KEY (IO\_OBJ\_ID)

, CONSTRAINT U1\_Mil\_IO\_OBJ UNIQUE (IO\_OBJ\_Name)

, CONSTRAINT U2\_Mil\_IO\_OBJ UNIQUE (IO\_OBJ\_Number)

)

CREATE TABLE Mil\_IO\_Task (

IO\_Task\_ID int identity

, IO\_Task\_Name varchar(200) NOT NULL

, IO\_Task\_Type varchar(30)

, IO\_Task\_Description varchar(200)

, CONSTRAINT PK\_Mil\_IO\_Task PRIMARY KEY (IO\_Task\_Id)

, CONSTRAINT U1\_Mil\_IO\_Task UNIQUE (IO\_Task\_Name)

)

--Begin to import data into created tables, starting with Implementing Unit

INSERT INTO Mil\_Implementing\_Unit(Implementing\_Unit\_Name, Implementing\_Unit\_Personnel,

Implementing\_Unit\_Location, Implementing\_Unit\_Type)

VALUES ('CAT495', '4', 'Fort Bragg', 'Civil Affairs')

, ('CAT577', '4', 'Fort Bragg', 'Civil Affairs')

, ('CAT599', '4', 'Fort Bragg', 'Civil Affairs')

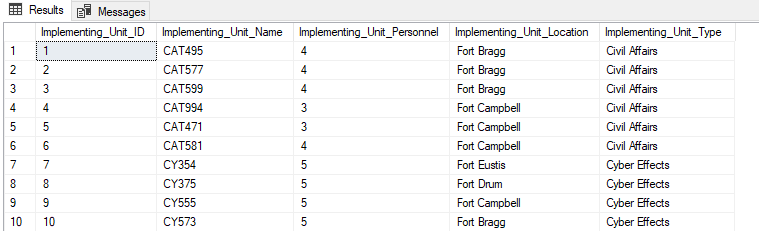
, ('CAT994', '3', 'Fort Campbell', 'Civil Affairs')

, ('CAT471', '3', 'Fort Campbell', 'Civil Affairs')

--At this point, and further along in the code, we have completed insert via excel for the remainder of the implementing units. Displayed below.

--Ensure our INSERT STATEMENT was executed Correctly

SELECT \* FROM Mil\_Implementing\_Unit --Works



--Execute conditional Select on same table

SELECT \* FROM Mil\_Implementing\_Unit

WHERE Implementing\_Unit\_Personnel >3 -- Works. all values are > 3 personnel (not shown)

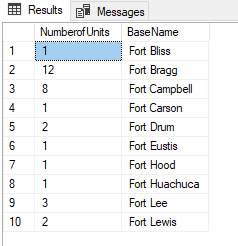
--Select Statement to determine how many units are located at each base.

SELECT Count(Implementing\_Unit\_Name) AS NumberofUnits, Implementing\_Unit\_Location AS BaseName

FROM Mil\_Implementing\_Unit

GROUP BY Implementing\_Unit\_Location

--Returns each Base with Number of Units at that Base



--Select Statement same as above with

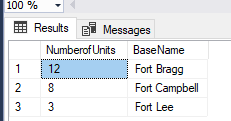
SELECT Count(Implementing\_Unit\_Name) AS NumberofUnits, Implementing\_Unit\_Location AS BaseName

FROM Mil\_Implementing\_Unit

GROUP BY Implementing\_Unit\_Location

HAVING COUNT(Implementing\_Unit\_Name) >2

--Returns each base with number of units > 3 currently stationed there.



--Continuing one with inserting 5 rows for each of the already created tables. The rest will be imported from excel

--Altering the Mil\_Special\_Skill table to include another column for "Special Skill Description"

ALTER TABLE Mil\_Special\_Skill

ADD Special\_Skill\_Description varchar(200)

--Checking table

SELECT \* FROM Mil\_Special\_Skill

INSERT INTO Mil\_Special\_Skill(Special\_Skill\_Text, Special\_Skill\_Description)

VALUES ('Advanced Negotiation', 'Dealmaking and leverage')

, ('Counter-Surveillance', 'Tracking Avoidance')

, ('Advanced Language', 'In-depth Target Language Training')

, ('Joint Operational Planning', 'Strategic Planning')

, ('Network Development', 'Utilization of Organic Human Networks')

--Continuing on with this table because there are only 12 rows

INSERT INTO Mil\_Special\_Skill(Special\_Skill\_Text, Special\_Skill\_Description)

VALUES ('Humanitarian Operations', 'Interagency Disaster Response')

, ('Strategic Communications', 'Strategic Messaging')

, ('Conflict Mediation', 'Ensuring Compromise')

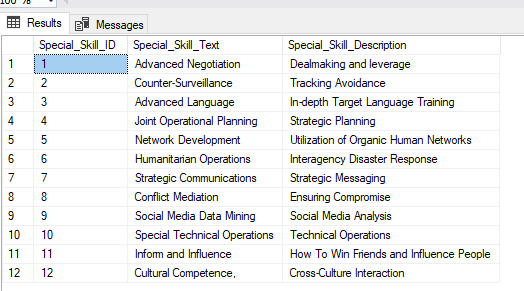
, ('Social Media Data Mining', 'Social Media Analysis')

, ('Special Technical Operations', 'Technical Operations')

, ('Inform and Influence', 'How To Win Friends and Influence People')

, ('Cultural Competence,', 'Cross-Culture Interaction')

SELECT \* FROM Mil\_Special\_Skill



--Next Table, Target Audience (5 Rows Only)

SELECT \* FROM Mil\_Target\_Audience --Ensure we have the correct columns for the insert, Rest will be inserted from Excel (not shown, select statement not shown at this point)

INSERT INTO Mil\_Target\_Audience

VALUES ('AZE Group 1', 'Azerbaijani', 'Religious', 'Baku')

, ('AZE Group 2', 'Azerbaijani', 'Service', 'Baku')

, ('AZE Group 3', 'Azerbaijani', 'Cultural', 'Sheki')

, ('AZE Group 4', 'Azerbaijani', 'Patrioritc', 'Sheki')

, ('EST Group 1', 'Estonian', 'Religious', 'Tallinn')

--Next Table, Beliefs all rows, including altering table to include Belief Description

ALTER TABLE Mil\_Beliefs

ADD Belief\_Description varchar (200)

SELECT \* FROM Mil\_Beliefs

INSERT INTO Mil\_Beliefs

VALUES ('Anarchism', 'Abolition of all government')

, ('Fascism', 'Authoritarian Ultra-Nationalism')

, ('Environmentalism', 'Protection of the Environment')

, ('Liberalism', 'Freedom of the Individual')

, ('NeoLiberalism', 'Economic Liberalism')

, ('Conservatism', 'Traditional Social Institutions')

, ('NeoConservatism', 'Interventionism in International Affairs')

, ('Socialism', 'Public Ownership')

, ('Nationalism', 'National Sovereignty')



--Next Table, motivations, all entries

SELECT \* FROM Mil\_Motivations

INSERT INTO Mil\_Motivations

VALUES ('Financial')

, ('Political')

, ('Ideological')

, ('Religious')

, ('Preservational')

, ('Reputational')

, ('Vengeful')

--not showing Mil\_Motivations or Mil\_Grievances currently because redundant

--Next Table, Mil\_Grievances all values

SELECT \* From Mil\_Grievances

INSERT INTO Mil\_Grievances

VALUES ('Persecution')

, ('Violence')

, ('Oppression')

, ('Exclusion')

, ('Racism')

, ('Economic')

, ('Corruption')

, ('Elitism')

, ('Unemployment')

, ('Capacity Deficit')

--Insert into USG OBJs, all values

SELECT \* FROM Mil\_USG\_OBJ

INSERT INTO Mil\_USG\_OBJ

VALUES ('Strengthen Alliances and Partnerships', '1/25/2019', '3') --checking

SELECT\* FROM Mil\_USG\_OBJ

--Continuing with insert

INSERT INTO Mil\_USG\_OBJ

VALUES ('Deter Adversaries', '1/25/2019', '2')

, ('Set Conditions for Future Operations', '2/24/2019', '5')

, ('Stabilization and Development', '2/24/2019', '6')

, ('Bolster Partners Against Coercion', '1/25/2019', '4')

, ('Prevent Attacks Against the US Homeland', '2/24/2019', '1')

SELECT \* FROM Mil\_USG\_OBJ

ORDER BY USG\_OBJ\_Priority ASC



--the above illustrates our inserts in the Mil\_USG\_OBJ table and orders them in ascending --pattern with emphasis on priority

--Continuing with Mil\_CDR\_OBJ, all values

SELECT \* FROM Mil\_CDR\_OBJ

INSERT INTO Mil\_CDR\_OBJ

VALUES ('1', 'Training and Preparing Full Spectrum Capable Forces', '2/15/2019', '4', 'Ongoing')

, ('2', 'Strengthening Alliances and Building Partner Military Capacity', '4/14/2019', '5', 'Ongoing')

, ('3', 'Providing Support to Other Government Services', '5/30/2019', '8', 'Ongoing')

, ('4', 'Detering Adversary Aggression', '2/15/2019', '2', 'Ongoing')

, ('5', 'Informing and Influencing External Audiences', '4/14/2019', '7', 'Ongoing')

, ('6', 'Supporting Partner Nation Governance Capabilities', '4/14/2019', '6', 'Ongoing')

, ('7', 'Minimizing Negative Effects of Non-State Actors', '7/5/2019', '12', 'Suspended')

, ('8', 'Increasing Regional Capacity to Respond to Natural Disasters', '6/15/2019', '9', 'Ongoing')

, ('9', 'Developing Resilient Logistical and Human Networks', '6/15/2019', '10', 'Ongoing')

, ('10', 'Infrastructure Development', '6/15/2019', '11', 'Suspended')

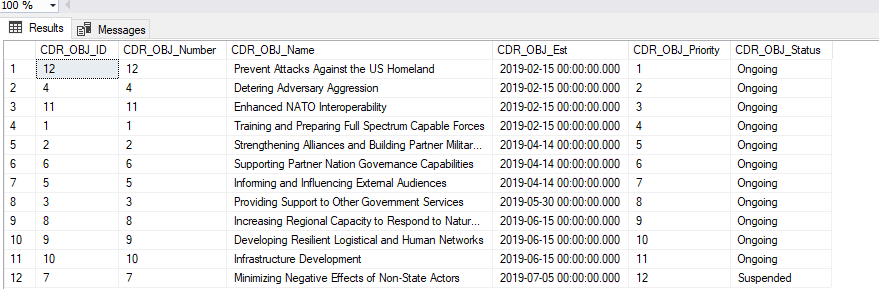
, ('11', 'Enhanced NATO Interoperability', '2/15/2019', '3', 'Ongoing')

, ('12', 'Prevent Attacks Against the US Homeland', '2/15/2019', '1', 'Ongoing')

--Checking, ordering by priority importance.

Select \* FROM Mil\_CDR\_OBJ

ORDER BY CDR\_OBJ\_Priority ASC



--The Commander has decided to approve Infrastructure Development, since this has already been completed in our database, it is reflected in the above screen shot already.

UPDATE Mil\_CDR\_OBJ

SET CDR\_OBJ\_Status = 'Ongoing'

WHERE CDR\_OBJ\_Name = 'Infrastructure Development'

SELECT \* FROM Mil\_CDR\_OBJ

WHERE CDR\_OBJ\_Name = 'Infrastructure Development'

--Creating a procedure to update Mil\_CDR\_OBJ.CDR\_OBJ\_Status from Suspended to Ongoing

GO

CREATE PROCEDURE ApproveCDRStatus (@CDR\_OBJ\_ID int)

AS

BEGIN

UPDATE Mil\_CDR\_OBJ SET CDR\_OBJ\_Status = 'Ongoing'

WHERE Mil\_CDR\_OBJ.CDR\_OBJ\_ID = @CDR\_OBJ\_ID

END

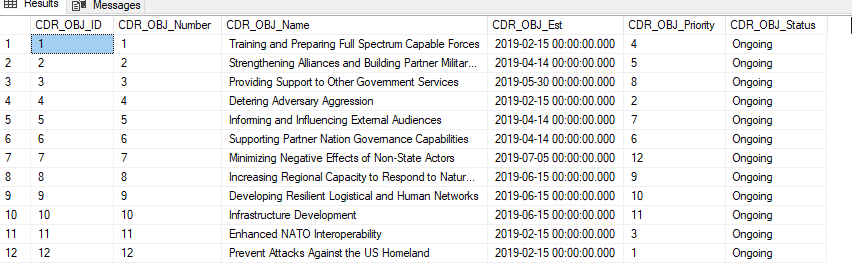
GO

--Testing the update procedure, it did change CDR\_OBJ\_ID ‘7’ to “Ongoing”, now all

--CDR\_OBJs are approved.

EXEC ApproveCDRStatus '7'

SELECT \* FROM Mil\_CDR\_OBJ



--Continuing with Mil\_IO\_OBJ, we have imported 21 rows from excel, not going to show

--result

--Importing the IO TASK TABLE FROM EXCEL, too many rows for manual import

--Creating the table Mil\_Unit\_Skill which will associate implementing units with their special tables

CREATE TABLE Mil\_Unit\_Skill (

Unit\_Special\_Skill\_ID int identity

, Implementing\_Unit\_ID int NOT NULL

, Special\_Skill\_ID int NOT NULL

, Implementing\_Unit\_Name varchar(200)

, Special\_Skill\_Text varchar(200)

, CONSTRAINT PK\_Mil\_Unit\_Skill PRIMARY KEY (Unit\_Special\_Skill\_ID)

, CONSTRAINT FK1\_Mil\_Unit\_Skill FOREIGN KEY (Implementing\_Unit\_ID) REFERENCES

Mil\_Implementing\_Unit(Implementing\_Unit\_ID)

, CONSTRAINT FK2\_Mil\_Unit\_Skill FOREIGN KEY (Special\_Skill\_Id) REFERENCES

Mil\_Special\_Skill(Special\_Skill\_ID)

)

INSERT INTO Mil\_Unit\_Skill (Implementing\_Unit\_ID, Special\_Skill\_ID, Implementing\_Unit\_Name, Special\_Skill\_Text)

VALUES ('1', '1', (SELECT Implementing\_Unit\_Name FROM Mil\_Implementing\_Unit WHERE Implementing\_Unit\_ID = '1'), (SELECT

Special\_Skill\_Text FROM Mil\_Special\_Skill WHERE Special\_Skill\_ID = '1'))

--Altering table to include unique composite

ALTER TABLE Mil\_Unit\_Skill

ADD CONSTRAINT U1\_Mil\_Unit\_Skill UNIQUE (Implementing\_Unit\_ID, Special\_Skill\_ID)

-Continuing with inserts for this table

INSERT INTO Mil\_Unit\_Skill (Implementing\_Unit\_ID, Special\_Skill\_ID)

VALUES ('1', '4'), ('1', '5'), ('1', '12'), ('2','1'), ('2', '9'), ('2', '12'), ('3', '1'), ('3', '3')

--This will bring over the Implementing Unit Names from Mil\_Implementing\_Unit

UPDATE Mil\_Unit\_Skill

SET Implementing\_Unit\_Name = Mil\_Implementing\_Unit.Implementing\_Unit\_Name

FROM Mil\_Unit\_Skill

INNER JOIN Mil\_Implementing\_Unit on Mil\_implementing\_Unit.Implementing\_Unit\_ID = Mil\_Unit\_Skill.Implementing\_Unit\_ID

WHERE Mil\_Implementing\_Unit.Implementing\_Unit\_ID = Mil\_Unit\_Skill.Implementing\_Unit\_ID

--This will bring over the Special Skill text from Mil\_Special\_Skill

UPDATE Mil\_Unit\_Skill

SET Special\_Skill\_Text = Mil\_Special\_Skill.Special\_Skill\_Text

FROM Mil\_Unit\_Skill

RIGHT JOIN Mil\_Special\_Skill on Mil\_Special\_Skill.Special\_Skill\_ID = Mil\_Unit\_Skill.Special\_Skill\_ID

WHERE Mil\_Special\_Skill.Special\_Skill\_Id = Mil\_Unit\_Skill.Special\_Skill\_Id

--Continuing with inserts for this table

INSERT INTO Mil\_Unit\_Skill (Implementing\_Unit\_ID, Special\_Skill\_ID)

VALUES ('3', '4'), ('3', '6'), ('3', '7'), ('4', '1'), ('4', '3'), ('4', '12')

--Rest of Implementing Unit has been pulled from excel

--Excel Insert failed, continuing with the rest of the data manually

INSERT INTO Mil\_Unit\_Skill (Implementing\_Unit\_ID, Special\_Skill\_ID)

VALUES ('5', '1'), ('5', '6'), ('5', '8'), ('5', '9'), ('7', '9'), ('8', '9'), ('8', '10'),

('9', '9'), ('9', '11'), ('10', '3'), ('10', '9'), ('10', '10'), ('10', '11'), ('11', '9'),

('11', '10'), ('11', '11'), ('11', '12'), ('12', '9'), ('13', '3'), ('13', '5'), ('13', '7'),

('13', '9'), ('13', '10'), ('13', '11'), ('13', '12'), ('14', '3'), ('14', '9'), ('14', '10'),

('14', '11'), ('14', '12'), ('15', '2'), ('15', '9'), ('15', '10'), ('15', '11'), ('16', '3'),

('16', '4'), ('16', '5'), ('16', '7'), ('16', '9'), ('16', '11'), ('16', '12'), ('17', '2'),

('17', '3'), ('17', '4'), ('17', '5'), ('17', '7'), ('17', '9'), ('17', '11'), ('17', '12'),

('18', '3'), ('18', '4'), ('18', '5'), ('18', '9'), ('18', '11'), ('19', '2'),

('19', '3'), ('19', '7'), ('19', '11'), ('18', '7'), ('20', '1'), ('20', '2'), ('20', '3'),

('20', '4'), ('20', '5'), ('20', '7'), ('20', '11'), ('21', '3'), ('21', '4'),

('21', '5'), ('21', '7'), ('21', '11'), ('21', '12'), ('22', '1'), ('22', '3'), ('22', '7'),

('22', '11'), ('22', '12'), ('23', '7'), ('23', '11'), ('24', '1'), ('24', '3'), ('24', '7'),

('24', '11'), ('24', '12'), ('25', '2'), ('25', '3'), ('25', '5'), ('26', '2'), ('26', '3'),

('26', '5'), ('26', '10'), ('26', '11'), ('27', '2'), ('27', '3'), ('27', '5'), ('27', '10'),

('27', '11'), ('28', '2'), ('28', '3'), ('28', '5'), ('28', '10'), ('29', '1'), ('29', '2'),

('29', '3'), ('29', '5'), ('29', '10'), ('29', '11'), ('29', '12'), ('30', '2'), ('30', '3'),

('30', '5'), ('30', '10'), ('30', '11'), ('30', '12'), ('31', '2'), ('31', '3'), ('31', '5'),

('32', '1'), ('32', '2'), ('32', '3'), ('32', '5'), ('32', '10'), ('32', '11'), ('32', '12')

--This will bring over the Implementing Unit Names from Mil\_Implementing\_Unit

UPDATE Mil\_Unit\_Skill

SET Implementing\_Unit\_Name = Mil\_Implementing\_Unit.Implementing\_Unit\_Name

FROM Mil\_Unit\_Skill

INNER JOIN Mil\_Implementing\_Unit on Mil\_implementing\_Unit.Implementing\_Unit\_ID = Mil\_Unit\_Skill.Implementing\_Unit\_ID

WHERE Mil\_Implementing\_Unit.Implementing\_Unit\_ID = Mil\_Unit\_Skill.Implementing\_Unit\_ID

SELECT \* FROM Mil\_Unit\_Skill

--This will bring over the Special Skill text from Mil\_Special\_Skill

UPDATE Mil\_Unit\_Skill

SET Special\_Skill\_Text = Mil\_Special\_Skill.Special\_Skill\_Text

FROM Mil\_Unit\_Skill

RIGHT JOIN Mil\_Special\_Skill on Mil\_Special\_Skill.Special\_Skill\_ID = Mil\_Unit\_Skill.Special\_Skill\_ID

WHERE Mil\_Special\_Skill.Special\_Skill\_Id = Mil\_Unit\_Skill.Special\_Skill\_Id

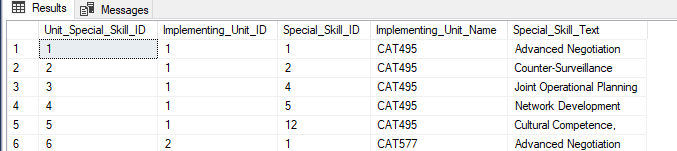
SELECT \* FROM Mil\_Unit\_Skill --Check, the result of our work joining these tables.

---Only the first 6 rows. With a simple Where clauses, we can SELECT \* FROM

--Mil\_Unit\_Skill

--WHERE Mil\_Unit\_Skill.Implementing\_Unit\_ID = ‘1’

--To understand all of the special skills that ‘Implementing Unit CAT495’ possesses



--Create 3 Target audience tables which incorporate beliefs, motivations and grievances

CREATE TABLE Mil\_TA\_Beliefs (

TA\_Belief\_ID int identity

, Target\_Audience\_ID int NOT NULL

, Belief\_ID int NOT NULL

, Belief\_Text varchar(50)

, Target\_Audience\_Text varchar(50)

, CONSTRAINT PK\_Mil\_TA\_Beliefs PRIMARY KEY(TA\_Belief\_ID)

, CONSTRAINT FK1\_Mil\_TA\_Beliefs FOREIGN KEY(Target\_Audience\_ID) REFERENCES

Mil\_Target\_Audience(Target\_Audience\_ID)

, CONSTRAINT FK2\_Mil\_TA\_Beliefs FOREIGN KEY (Belief\_ID) REFERENCES

Mil\_Beliefs(Belief\_ID)

)

--Inserting all rows into the Mil\_TA\_Beliefs table and we will then update the table with

--belief and audience text with an update

--checking first to ensure we have the correct values in the target audience table

SELECT \* FROM Mil\_Target\_Audience

INSERT INTO Mil\_TA\_Beliefs(Target\_Audience\_ID, Belief\_ID)

VALUES ('1','1'), ('2', '2'), ('3', '3'), ('4', '4'), ('5', '6'), ('6', '7'), ('7', '8'), ('8', '9'),

('9','2'), ('10', '4'), ('11', '5'), ('12', '8'), ('13', '3'), ('14', '6'), ('15', '4'), ('16', '1'),

('17','6'), ('18', '4'), ('19', '3'), ('20', '2'), ('21', '1'), ('22', '2'), ('23', '3'), ('24', '4'),

('25','6'), ('26', '2')

--updating Mil\_TA\_Beliefs to include belief text from Mil\_Beliefs

UPDATE Mil\_TA\_Beliefs

SET Belief\_Text = Mil\_Beliefs.Belief\_Text

FROM Mil\_TA\_Beliefs

RIGHT JOIN Mil\_Beliefs ON Mil\_Beliefs.Belief\_ID = Mil\_TA\_Beliefs.Belief\_ID

WHERE Mil\_Beliefs.Belief\_ID = Mil\_TA\_Beliefs.Belief\_ID

SELECT \* FROM Mil\_TA\_Beliefs

WHERE Belief\_Text = 'Conservatism' -- works, returns 4 rows with belief\_text set to conservatism

--updating Mil\_TA\_Beliefs to include target\_audience\_text from mil\_target\_Audience

UPDATE Mil\_TA\_Beliefs

SET Target\_Audience\_Text = Mil\_Target\_Audience.Target\_Audience\_Name

FROM Mil\_TA\_Beliefs

RIGHT JOIN Mil\_Target\_Audience ON Mil\_Target\_Audience.Target\_Audience\_Id = Mil\_TA\_Beliefs.Target\_Audience\_ID

WHERE Mil\_Target\_Audience.Target\_Audience\_ID = Mil\_TA\_Beliefs.Target\_Audience\_ID

--Try a join to select the city (location from Mil\_Target\_Audience) with the option to filter by city name

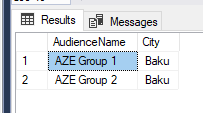
Select Mil\_TA\_Beliefs.Target\_Audience\_Text AS AudienceName, Mil\_Target\_Audience.Target\_Audience\_Location AS City

FROM Mil\_TA\_Beliefs

RIGHT JOIN Mil\_Target\_Audience ON Mil\_TA\_Beliefs.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

WHERE Mil\_Target\_Audience.Target\_Audience\_Location = 'Baku'

--The above returns two columns and values with the WHERE CLAUSE, otherwise all values



--Going to Create a table called Mil\_TA\_Motivations, we will insert 5 rows and then import the rest from excel

CREATE TABLE Mil\_TA\_Motivations (

Mil\_TA\_Motivations\_ID int identity

, Target\_Audience\_ID int NOT NULL

, Motivation\_ID int NOT NULL

, Motivation\_Text varchar(30)

, CONSTRAINT PK\_Mil\_TA\_Motivations PRIMARY KEY (Mil\_TA\_Motivations\_ID)

, CONSTRAINT FK1\_Mil\_TA\_Motivations FOREIGN KEY (Target\_Audience\_ID) REFERENCES

Mil\_Target\_Audience(Target\_Audience\_ID)

, CONSTRAINT FK2\_Mil\_TA\_Motivations FOREIGN KEY (Motivation\_ID) REFERENCES

Mil\_Motivations(Motivation\_ID)

)

INSERT INTO Mil\_TA\_Motivations (Target\_Audience\_ID, Motivation\_ID)

VALUES ('1', '1'), ('1', '2'), ('1', '4'), ('2', '2'), ('2', '3') -- 5 rows added, motivation text is still null, exporting rest from excel

--Import error, checking - Had to delete blank row somewhere in XL

--updating motivation texts

UPDATE Mil\_TA\_Motivations

SET Motivation\_Text = Mil\_Motivations.Motivation\_Text

FROM Mil\_TA\_Motivations

RIGHT JOIN MIl\_Motivations on Mil\_TA\_Motivations.Motivation\_ID = Mil\_Motivations.Motivation\_ID

WHERE Mil\_TA\_Motivations.Motivation\_ID = Mil\_Motivations.Motivation\_ID

--Trying some select statements to show which Target Audiences have which motivations

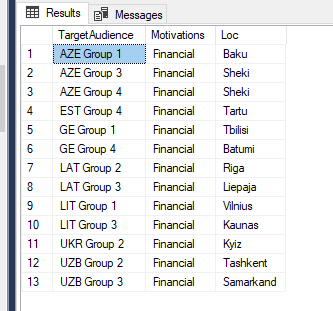
SELECT Mil\_Target\_audience.Target\_Audience\_Name AS TargetAudience, Motivation\_Text as Motivations, Mil\_Target\_Audience.Target\_Audience\_Location AS Loc

-- did not return what we wanted, so we delete. (SELECT Count (Target\_Audience\_Location) --FROM Mil\_Target\_Audience WHERE Target\_Audience\_Location = 'Baku')

FROM Mil\_TA\_Motivations

RIGHT JOIN Mil\_Target\_Audience ON Mil\_Target\_Audience.Target\_Audience\_ID = Mil\_TA\_Motivations.Target\_Audience\_ID

Where Motivation\_Text = 'Financial'



--Creating and importing Mil\_TA\_grievance table

CREATE TABLE Mil\_TA\_Grievances (

TA\_Grievance\_ID int identity

, Target\_Audience\_ID int NOT NULL

, Grievance\_ID int NOT NULL

, Grievance\_Text varchar(30)

, CONSTRAINT PK\_Mil\_TA\_Grievances PRIMARY KEY (TA\_Grievance\_ID)

, CONSTRAINT FK1\_Mil\_TA\_Grievances FOREIGN KEY (Target\_audience\_ID)

REFERENCES Mil\_Target\_Audience(Target\_Audience\_ID)

, CONSTRAINT FK2\_Mil\_TA\_Grievances FOREIGN KEY (Grievance\_ID)

REFERENCES Mil\_Grievances(Grievance\_ID)

)

--Importing values via excel

--Import grievances text based on Mil\_Grievance table.

UPDATE Mil\_TA\_Grievances

SET Grievance\_Text = Mil\_Grievances.Grievance\_Text

FROM Mil\_TA\_Grievances

RIGHT JOIN Mil\_Grievances ON Mil\_TA\_Grievances.Grievance\_ID = Mil\_Grievances.Grievance\_Id

WHERE Mil\_TA\_Grievances.Grievance\_ID = Mil\_Grievances.Grievance\_Id

--See what kind of join we can do with this newly created table,

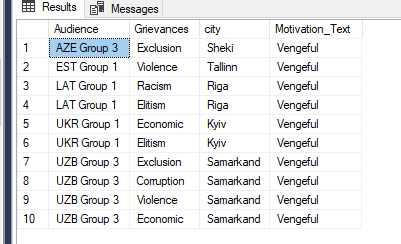
SELECT Mil\_Target\_Audience.Target\_Audience\_Name AS Audience, Mil\_TA\_Grievances.Grievance\_Text as Grievances, Mil\_Target\_Audience.Target\_Audience\_Location AS city, Mil\_TA\_Motivations.Motivation\_Text

FROM Mil\_TA\_Grievances

RIGHT JOIN Mil\_Target\_Audience ON Mil\_TA\_Grievances.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

RIGHT JOIN Mil\_TA\_Motivations ON Mil\_TA\_Motivations.Target\_Audience\_ID = Mil\_TA\_Grievances.Target\_Audience\_ID

WHERE Motivation\_Text = 'vengeful' –optional



/\*What does this ultimate do for us? It allows us to see which target audiences in which cities are motivated for “Vengeance” while also displaying their chief grievances towards a government or oppressive group. This will be helpful in illustrating which conditions cause our information operations to be more successful in a particular city, society or against a certain type of group.

\*/

CREATE TABLE Mil\_CDR\_USG\_XWalk (

USG\_CDR\_OBJ\_ID int identity

, USG\_OBJ\_ID int NOT NULL

, CDR\_OBJ\_ID int NOT NULL

, USG\_OBJ\_Name varchar(200)

, CDR\_OBJ\_Name varchar(200)

, USG\_CDR\_Ver\_Date datetime

, USG\_CDR\_SPT\_Ind int

, CONSTRAINT PK\_Mil\_CDR\_USG\_XWalk PRIMARY KEY (USG\_CDR\_OBJ\_ID)

, CONSTRAINT FK1\_Mil\_CDR\_USG\_XWalk FOREIGN KEY (USG\_OBJ\_ID)

REFERENCES Mil\_USG\_OBJ(USG\_OBJ\_ID)

, CONSTRAINT FK2\_Mil\_CDR\_USG\_XWalk FOREIGN KEY (CDR\_OBJ\_ID)

REFERENCES Mil\_CDR\_OBJ(CDR\_OBJ\_ID)

, CONSTRAINT U1\_Mil\_CDR\_USG\_Xwalk UNIQUE (USG\_OBJ\_ID, CDR\_OBJ\_ID)

)

--Inserting from Xl USG OBJ ID, CDR OBJ ID, Verification Date and Supportability index 42 rows

--Updating the Mil\_USG\_CDR\_Xwalk table to include USG\_OBJ\_Names and CDR\_OBJ\_Names

UPDATE Mil\_CDR\_USG\_XWalk

SET USG\_OBJ\_Name = Mil\_USG\_OBJ.USG\_OBJ\_Name

FROM Mil\_CDR\_USG\_XWalk

RIGHT JOIN Mil\_USG\_OBJ ON Mil\_CDR\_USG\_XWalk.USG\_OBJ\_ID = Mil\_USG\_OBJ.USG\_OBJ\_ID

WHERE Mil\_CDR\_USG\_XWalk.USG\_OBJ\_ID = Mil\_USG\_OBJ.USG\_OBJ\_ID

UPDATE Mil\_CDR\_USG\_XWalk

SET CDR\_OBJ\_Name = Mil\_CDR\_OBJ.CDR\_OBJ\_Name

FROM Mil\_CDR\_USG\_XWalk

RIGHT JOIN Mil\_CDR\_OBJ ON Mil\_CDR\_USG\_XWalk.CDR\_OBJ\_ID = Mil\_CDR\_OBJ.CDR\_OBJ\_ID

WHERE Mil\_CDR\_USG\_XWalk.CDR\_OBJ\_ID = Mil\_CDR\_OBJ.CDR\_OBJ\_ID

--check to see if values inserted properly

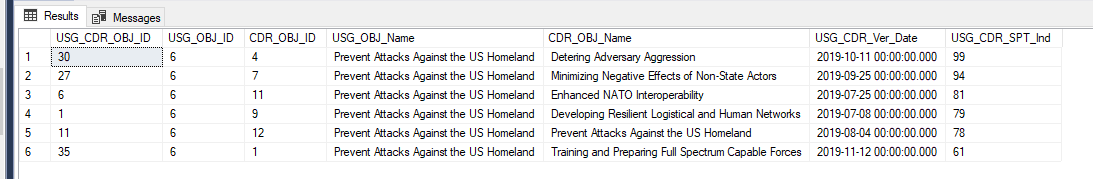
SELECT \* FROM Mil\_CDR\_USG\_XWalk

WHERE Mil\_CDR\_USG\_XWalk.USG\_OBJ\_ID = '6'

ORDER BY USG\_CDR\_SPT\_Ind DESC

/\*this statement informs us of how many of the commander’s objectives directly support USG objectives and to what extent we believe that they will support that objective moving forward. We chose Mil\_CDR\_USG\_Xwalk.USG\_OBJ\_ID = ‘6’ because that is the #1 priority for the USG government according to our table display early for Mil\_USG\_OBJ. This view shows that 6 of our CDR\_OBJs will be supporting this #1 USG priority and they range from a ‘99’ to a ‘61’ – which denotes how much we believe achieving these commander objectives will support achieving a USG objective. (Note – this is somewhat randomly generated data which has led to a few contradictions in the above claim throughout the dataset, but it illustrates the point well enough. I believe if this were to be implemented in reality, we would leave the supportability index blank and have actual activities inform the column value. We would assess how effectively the CDR objectives support the USG Objectives through other DML, but I don’t think we have the data to execute it right now and we are not there yet in terms of programming skill.

\*/



--Create Mil\_USG\_IO\_OBJ\_XWalk table

CREATE TABLE Mil\_CDR\_IO\_OBJ\_XWalk (

CDR\_IO\_OBJ int identity

, CDR\_OBJ\_ID int not null

, IO\_OBJ\_ID int not null

, CDR\_OBJ\_Name varchar(200)

, IO\_OBJ\_Name varchar(200)

, CDR\_IO\_Ver\_Date datetime

, CDR\_IO\_Spt\_Ind int

, CONSTRAINT PK\_Mil\_CDR\_IO\_OBJ\_XWalk PRIMARY KEY (CDR\_IO\_OBJ)

, CONSTRAINT FK1\_Mil\_CDR\_IO\_OBJ\_XWalk FOREIGN KEY (CDR\_OBJ\_ID)

REFERENCES Mil\_CDR\_OBJ(CDR\_OBJ\_ID)

, CONSTRAINT FK2\_Mil\_CDR\_IO\_OBJ\_XWalk FOREIGN KEY (IO\_OBJ\_ID)

REFERENCES Mil\_IO\_OBJ(IO\_OBJ\_ID)

, CONSTRAINT U1\_Mil\_CDR\_IO\_OBJ\_XWalk UNIQUE (CDR\_OBJ\_ID, IO\_OBJ\_ID)

)

--Excel values insert for Mil\_CDR\_IO\_OBJ\_Xwalk

--Updates on CDR\_OBJ\_Name and IO\_OBJ\_Name

UPDATE Mil\_CDR\_IO\_OBJ\_XWalk

SET CDR\_OBJ\_Name = Mil\_CDR\_OBJ.CDR\_OBJ\_Name

FROM Mil\_CDR\_IO\_OBJ\_XWalk

RIGHT JOIN Mil\_CDR\_OBJ on Mil\_CDR\_IO\_OBJ\_XWalk.CDR\_OBJ\_ID = Mil\_CDR\_OBJ.CDR\_OBJ\_ID

WHERE Mil\_CDR\_IO\_OBJ\_XWalk.CDR\_OBJ\_ID = Mil\_CDR\_OBJ.CDR\_OBJ\_ID

UPDATE Mil\_CDR\_IO\_OBJ\_XWalk

SET IO\_OBJ\_Name = Mil\_IO\_OBJ.IO\_OBJ\_Name

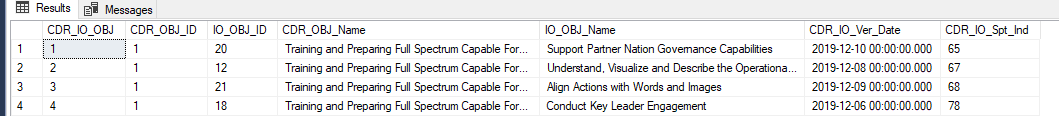
FROM Mil\_CDR\_IO\_OBJ\_XWalk

RIGHT JOIN Mil\_IO\_OBJ ON Mil\_CDR\_IO\_OBJ\_XWalk.IO\_OBJ\_ID = Mil\_IO\_OBJ.IO\_OBJ\_ID

WHERE Mil\_CDR\_IO\_OBJ\_XWalk.IO\_OBJ\_ID = Mil\_IO\_OBJ.IO\_OBJ\_ID

/\*Checking columns, this is the same as the above screenshot except now we are looking at CDR\_OBJ and IO OBJ. We are doing this so that we can illustrate how USG objectives drive commander objectives drive IO objectives. First, this ensures we “nest” our activities based on what our higher commander is telling us is important to him or her. Second, we know that we should be executing activities which achieve those priorities. It makes no sense to conduct activities or achieve objectives that have little importance to the USG.

\*/



--Create Table Mil\_IOOBJ\_Task\_XWalk

CREATE TABLE Mil\_IOOBJ\_Task\_XWalk (

IOOBJ\_Task\_ID int identity

, IO\_OBJ\_ID int

, IO\_Task\_ID int

, IO\_OBJ\_Name varchar(200)

, IO\_Task\_Name varchar(200)

, OBJ\_Task\_Ver\_Date datetime

, CONSTRAINT PK\_Mil\_IOOBJ\_Task\_XWalk PRIMARY KEY (IOOBJ\_Task\_ID)

, CONSTRAINT FK1\_Mil\_IOOBJ\_Task\_XWalk FOREIGN KEY (IO\_OBJ\_ID)

REFERENCES Mil\_IO\_OBJ(IO\_OBJ\_ID)

, CONSTRAINT FK2\_Mil\_IOOBJ\_Task\_XWalk FOREIGN KEY (IO\_Task\_ID)

REFERENCES Mil\_IO\_Task(IO\_Task\_ID)

, CONSTRAINT U1\_Mil\_IOOBJ\_Task\_Xwalk Unique (IO\_OBJ\_ID, IO\_Task\_ID)

)

--Inserting From XL, then adding the IO OBJ Name and IO Task Name

UPDATE Mil\_IOOBJ\_Task\_XWalk

SET IO\_OBJ\_Name = Mil\_IO\_OBJ.IO\_OBJ\_Name

FROM Mil\_IOOBJ\_Task\_XWalk

RIGHT JOIN Mil\_IO\_OBJ on Mil\_IOOBJ\_Task\_XWalk.IO\_OBJ\_ID = Mil\_IO\_OBJ.IO\_OBJ\_ID

WHERE Mil\_IOOBJ\_Task\_XWalk.IO\_OBJ\_ID = Mil\_IO\_OBJ.IO\_OBJ\_ID

UPDATE Mil\_IOOBJ\_Task\_XWalk

SET IO\_Task\_Name = Mil\_IO\_Task.IO\_Task\_Name

FROM Mil\_IOOBJ\_Task\_XWalk

RIGHT JOIN Mil\_IO\_Task on Mil\_IOOBJ\_Task\_XWalk.IO\_Task\_ID = Mil\_IO\_Task.IO\_Task\_ID

WHERE Mil\_IOOBJ\_Task\_XWalk.IO\_Task\_ID = Mil\_IO\_Task.IO\_Task\_ID

SELECT \* FROM Mil\_IOOBJ\_Task\_XWalk

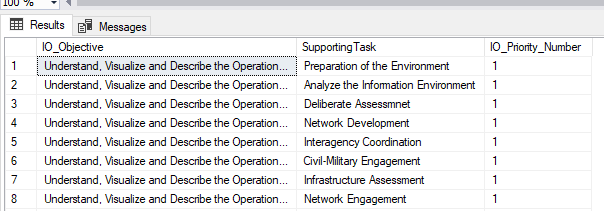
--If commander wanted to know how many of the supporting tasks were targeting his #1 IO priority, he could have his DBA pull the following:

Select Mil\_IOOBJ\_Task\_XWalk.IO\_OBJ\_Name AS IO\_Objective, IO\_Task\_Name as SupportingTask, Mil\_IO\_OBJ.IO\_OBJ\_Priority as IO\_Priority\_Number

FROM Mil\_IOOBJ\_Task\_XWalk

RIGHT JOIN Mil\_IO\_OBJ on Mil\_IOOBJ\_Task\_XWalk.IO\_OBJ\_ID = Mil\_IO\_OBJ.IO\_OBJ\_ID

ORDER BY Mil\_IOOBJ\_Task\_XWalk.IO\_Task\_Name



--If these are too numerous, he could then do the following below to count the number of supporting tasks which support the IO OBJ.

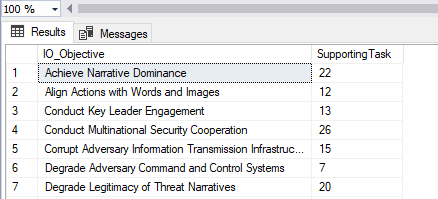
--Count of IO Tasks that support IO Objectives

Select Mil\_IOOBJ\_Task\_XWalk.IO\_OBJ\_Name AS IO\_Objective, count (IO\_Task\_Name) as SupportingTask--, Mil\_IO\_OBJ.IO\_OBJ\_Priority as IO\_Priority\_Number

FROM Mil\_IOOBJ\_Task\_XWalk

RIGHT JOIN Mil\_IO\_OBJ on Mil\_IOOBJ\_Task\_XWalk.IO\_OBJ\_ID = Mil\_IO\_OBJ.IO\_OBJ\_ID

GROUP BY Mil\_IOOBJ\_Task\_XWalk.IO\_OBJ\_Name



--There are 22 tasks supporting the #1 IO Priority, of 42 total IO\_Tasks

--Creating the Operations Table, this table will include IO tasks, IO OBJs and CDR OBJs

CREATE TABLE MIl\_Operations (

Operation\_ID int Identity

, Implementing\_Unit\_ID int not null

, Target\_Audience\_ID int not null

, IO\_Task\_ID int not null

, Implementing\_Unit\_Name varchar(30)

, Target\_Audience\_Name varchar(30)

, IO\_Task\_Name varchar(200)

, Date\_Begin datetime

, Date\_Complete datetime

, Opn\_Result varchar(30)

, Opn\_Effect int

, Opn\_Rec varchar(30)

, CONSTRAINT PK\_Mil\_Operations PRIMARY KEY (Operation\_ID)

, CONSTRAINT FK1\_Mil\_Operations FOREIGN KEY (Implementing\_Unit\_ID)

REFERENCES Mil\_Implementing\_Unit(Implementing\_Unit\_ID)

, CONSTRAINT FK2\_Mil\_Operatoins FOREIGN KEY (Target\_Audience\_ID)

REFERENCES Mil\_Target\_Audience(Target\_Audience\_ID)

, CONSTRAINT FK3\_Mil\_Operations FOREIGN KEY (IO\_Task\_ID)

REFERENCES Mil\_IO\_Task(IO\_Task\_ID)

)

--Inserting all foreign keys into Mil\_Operations via XL

--Update the FK values

UPDATE MIl\_Operations

SET Implementing\_Unit\_Name = Mil\_Implementing\_Unit.Implementing\_Unit\_Name

, Target\_Audience\_Name = Mil\_Target\_Audience.Target\_Audience\_Name

, IO\_Task\_Name = Mil\_IO\_Task.IO\_Task\_Name

FROM MIl\_Operations

RIGHT JOIN Mil\_Implementing\_Unit ON Mil\_Operations.Implementing\_Unit\_ID = Mil\_Implementing\_Unit.Implementing\_Unit\_ID

RIGHT JOIN Mil\_Target\_Audience ON MIl\_Operations.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

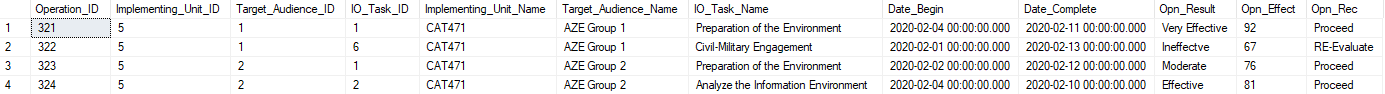
RIGHT JOIN Mil\_IO\_Task on MIl\_Operations.IO\_Task\_ID = Mil\_IO\_Task.IO\_Task\_ID

WHERE Mil\_Operations.Implementing\_Unit\_ID = Mil\_Implementing\_Unit.Implementing\_Unit\_ID

AND MIl\_Operations.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

AND MIl\_Operations.IO\_Task\_ID = Mil\_IO\_Task.IO\_Task\_ID

SELECT \* FROM MIl\_Operations



/\*This table is a list of actual IO\_tasks being conducted against a target audience by an implementing unit and the subsequent result and recommendation of that operations. This is just a few of these rows.

To illustrate the activities of a certain implementing unit, we can execute the following code which will

Select the unit, audience, display the task, location and assessed effectiveness. In this particular select statement, we are going to look at just one team and only display the operations which had an assessed effectiveness >80.

\*/

SELECT MIl\_Operations.Implementing\_Unit\_Name as Executing\_Unit

, MIl\_Operations.Target\_Audience\_Name as TargetAudience

, MIl\_Operations.IO\_Task\_Name as Task\_Executed

, Mil\_Target\_Audience.Target\_Audience\_Location as Location\_occurred

-- , Mil\_IO\_OBJ.IO\_OBJ\_Priority as IO\_Priority

, (MIl\_Operations.Opn\_Effect) as Assessed\_Effectiveness

, MIl\_Operations.Operation\_ID

FROM MIl\_Operations

JOIN Mil\_Target\_Audience on MIl\_Operations.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

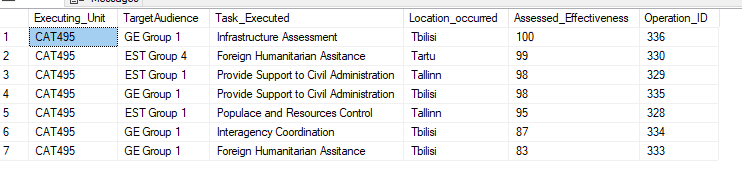
-- JOIN Mil\_IOOBJ\_Task\_XWalk on MIl\_Operations.IO\_Task\_ID = Mil\_IOOBJ\_Task\_XWalk.IO\_Task\_ID

--JOIN Mil\_IO\_OBJ on Mil\_IO\_OBJ.IO\_OBJ\_ID = Mil\_IOOBJ\_Task\_XWalk.IO\_OBJ\_ID

WHERE MIl\_Operations.Opn\_Effect > 80

and MIl\_Operations.Implementing\_Unit\_Name = 'CAT495'

Order by Opn\_Effect DESC



--A more in depth select below which will display all of the activities that a particular team has executed against number 1 IO priority.

SELECT MIl\_Operations.Implementing\_Unit\_Name as Executing\_Unit

, MIl\_Operations.Target\_Audience\_Name as TargetAudience

, MIl\_Operations.IO\_Task\_Name as Task\_Executed

, Mil\_Target\_Audience.Target\_Audience\_Location as Location\_occurred

, Mil\_IO\_OBJ.IO\_OBJ\_Priority as IO\_Priority

, (MIl\_Operations.Opn\_Effect) as Assessed\_Effectiveness

, MIl\_Operations.Operation\_ID

FROM MIl\_Operations

RIGHT JOIN Mil\_Target\_Audience on MIl\_Operations.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

RIGHT JOIN Mil\_IOOBJ\_Task\_XWalk on MIl\_Operations.IO\_Task\_ID = Mil\_IOOBJ\_Task\_XWalk.IO\_Task\_ID

RIGHT JOIN Mil\_IO\_OBJ on Mil\_IO\_OBJ.IO\_OBJ\_ID = Mil\_IOOBJ\_Task\_XWalk.IO\_OBJ\_ID

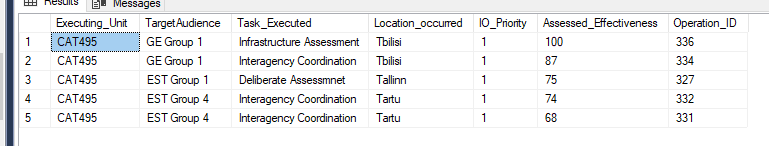
WHERE MIl\_Operations.Opn\_Effect > 55

and MIl\_Operations.Implementing\_Unit\_Name = 'CAT495'

and IO\_OBJ\_Priority = '1'

--GROUP BY Mil\_IO\_OBJ.IO\_OBJ\_Priority

Order by Assessed\_Effectiveness DESC



--Going further, we can assess the average effectiveness of our activities for a single implementing unit (CAT 495) with the below code.

SELECT MIl\_Operations.Implementing\_Unit\_Name as Executing\_Unit

, avg (MIl\_Operations.Opn\_Effect) as Assessed\_Effectiveness

FROM MIl\_Operations

RIGHT JOIN Mil\_Target\_Audience on MIl\_Operations.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

RIGHT JOIN Mil\_IOOBJ\_Task\_XWalk on MIl\_Operations.IO\_Task\_ID = Mil\_IOOBJ\_Task\_XWalk.IO\_Task\_ID

RIGHT JOIN Mil\_IO\_OBJ on Mil\_IO\_OBJ.IO\_OBJ\_ID = Mil\_IOOBJ\_Task\_XWalk.IO\_OBJ\_ID

WHERE MIl\_Operations.Opn\_Effect > 55

and MIl\_Operations.Implementing\_Unit\_Name = 'CAT495'

and IO\_OBJ\_Priority = '1'

GROUP BY MIl\_Operations.Implementing\_Unit\_Name



-- Works Can we do the same thing for all of the teams? Can we add layers of complexity?

--This statement will select all of the implementing units, their tasks and the resulting --effectiveness, ordering them by descending effectiveness

SELECT MIl\_Operations.Implementing\_Unit\_Name as Executing\_Unit

, MIl\_Operations.IO\_Task\_Name

, (MIl\_Operations.Opn\_Effect) as Assessed\_Effectiveness

FROM MIl\_Operations

RIGHT JOIN Mil\_Target\_Audience on MIl\_Operations.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

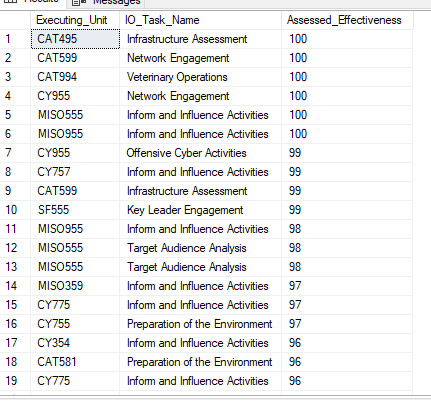
RIGHT JOIN Mil\_IOOBJ\_Task\_XWalk on MIl\_Operations.IO\_Task\_ID = Mil\_IOOBJ\_Task\_XWalk.IO\_Task\_ID

RIGHT JOIN Mil\_IO\_OBJ on Mil\_IO\_OBJ.IO\_OBJ\_ID = Mil\_IOOBJ\_Task\_XWalk.IO\_OBJ\_ID

WHERE MIl\_Operations.Opn\_Effect > 55

and IO\_OBJ\_Priority = '1'

ORDER BY MIl\_Operations.Opn\_Effect DESC



--This statement assesses the average effectiveness of each team against all objectives.

SELECT MIl\_Operations.Implementing\_Unit\_Name as Executing\_Unit

, avg (MIl\_Operations.Opn\_Effect) as Assessed\_Effectiveness

FROM MIl\_Operations

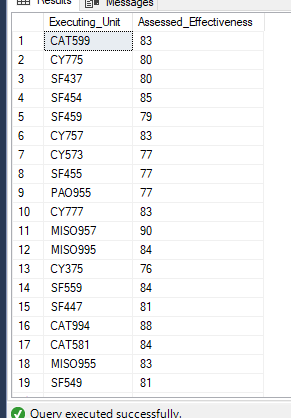
RIGHT JOIN Mil\_Target\_Audience on MIl\_Operations.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

RIGHT JOIN Mil\_IOOBJ\_Task\_XWalk on MIl\_Operations.IO\_Task\_ID = Mil\_IOOBJ\_Task\_XWalk.IO\_Task\_ID

RIGHT JOIN Mil\_IO\_OBJ on Mil\_IO\_OBJ.IO\_OBJ\_ID = Mil\_IOOBJ\_Task\_XWalk.IO\_OBJ\_ID

WHERE MIl\_Operations.Opn\_Effect > 55

GROUP BY MIl\_Operations.Implementing\_Unit\_Name



--Selects a target audience and the relevant beliefs, motivations and grievances for that group

SELECT Mil\_Target\_Audience.Target\_Audience\_Name as Target\_Audience, Mil\_TA\_Beliefs.Belief\_Text as Beliefs, Mil\_TA\_Motivations.Motivation\_Text as Motivations, Mil\_TA\_Grievances.Grievance\_Text as Grievances

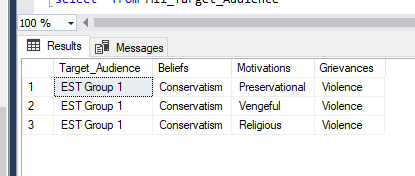
FROM Mil\_Target\_Audience

RIGHT JOIN Mil\_TA\_Beliefs on Mil\_TA\_Beliefs.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

RIGHT JOIN Mil\_TA\_Motivations on Mil\_Target\_Audience.Target\_Audience\_ID = Mil\_TA\_Motivations.Target\_Audience\_ID

RIGHT JOIN Mil\_TA\_Grievances on Mil\_Target\_Audience.Target\_Audience\_ID = Mil\_TA\_Grievances.Target\_Audience\_ID

where Target\_Audience\_Name = 'est group 1'



--This code displays special skills attached to particular units

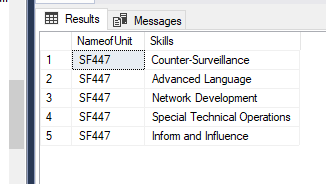
Select Mil\_Implementing\_Unit.Implementing\_Unit\_Name as NameofUnit, Mil\_Unit\_Skill.Special\_Skill\_Text as Skills

FROM Mil\_Implementing\_Unit

Right join Mil\_Unit\_Skill on mil\_Implementing\_Unit.Implementing\_Unit\_ID = Mil\_Unit\_Skill.Implementing\_Unit\_ID

WHERE Implementing\_Unit\_Type = 'Special Forces'

And Mil\_Implementing\_Unit.Implementing\_Unit\_Name = 'SF447'



Business Questions

1. Which Countries are most receptive to USG Information Operations?

Our Strategy for answering this question is to group the countries by average effectiveness of our operations against those country. While this is a rather simplistic way to determine success, we are averaging the effectiveness of about 350 distinct operations and using that as our metric for how receptive a country is to our activities.

/\*Seeking to determine which countries are most receptive to USG information operations, our plan in answering this is to join two tables Mil\_Target\_Audience and Mil\_Operations, select the Mil\_Target\_Audience ethnicity column as our Country and then select the average effectiveness of our Mil\_Operations.Opn\_Effect to show how well our teams have done overall in each country.

\*/

SELECT

Mil\_Target\_Audience.Target\_Audience\_Ethnicity AS TA\_Country

, avg (MIl\_Operations.Opn\_Effect) as Effectiveness

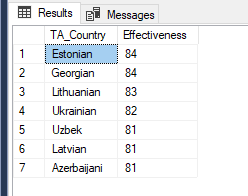
-- , MIl\_Operations.Operation\_ID as Operations\_Number

FROM MIl\_Operations

RIGHT JOIN Mil\_Target\_Audience on Mil\_Target\_Audience.Target\_Audience\_ID = MIl\_Operations.Target\_Audience\_ID

GROUP BY Mil\_Target\_Audience.Target\_Audience\_Ethnicity

ORDER BY avg (MIl\_Operations.Opn\_Effect) DESC



After completing this, we have determined that Estonia is the most receptive country, answering our business question #1.

We must determine if the particular activities being conducted in Estonia contrast greatly with a country like Azerbaijan. Do the certain activities in Estonia bring up the average effectiveness of all activities?

--Can we count the number of operations in estonia?

SELECT Mil\_IO\_Task.IO\_Task\_Name as TaskExecuted

, count (MIl\_Operations.IO\_Task\_Name) as NumberofTimesExecuted

-- , Mil\_Target\_Audience.Target\_Audience\_Name as TargetAudience

FROM Mil\_IO\_Task

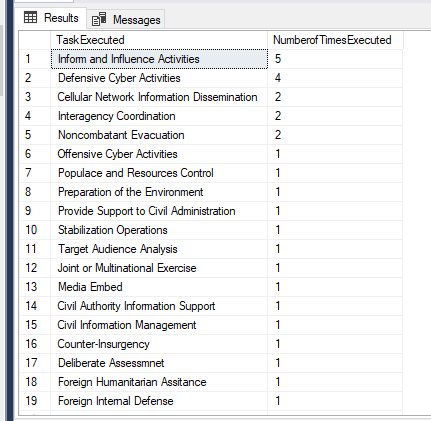
RIGHT JOIN Mil\_operations on MIl\_Operations.IO\_Task\_ID = Mil\_IO\_Task.IO\_task\_ID

RIGHT JOIN Mil\_Target\_Audience on Mil\_Target\_Audience.Target\_Audience\_ID = MIl\_Operations.Target\_Audience\_ID

WHERE Mil\_Target\_Audience.Target\_Audience\_Ethnicity = 'Estonian'

GROUP BY Mil\_IO\_Task.IO\_Task\_Name

order by count (mil\_operations.io\_task\_Name) DESC



--We need to compare this to the ‘least receptive’ country of Azerbaijan to determine any glaring differences between the two which may account for “receptiveness”

--Can we count the number of operations in Azerbaijan?

SELECT Mil\_IO\_Task.IO\_Task\_Name as TaskExecuted

, count (MIl\_Operations.IO\_Task\_Name) as NumberofTimesExecuted

-- , Mil\_Target\_Audience.Target\_Audience\_Name as TargetAudience

FROM Mil\_IO\_Task

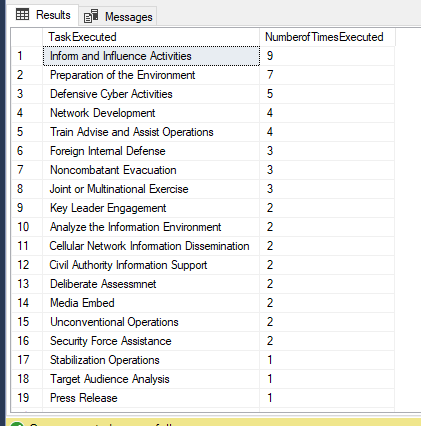
RIGHT JOIN Mil\_operations on MIl\_Operations.IO\_Task\_ID = Mil\_IO\_Task.IO\_task\_ID

RIGHT JOIN Mil\_Target\_Audience on Mil\_Target\_Audience.Target\_Audience\_ID = MIl\_Operations.Target\_Audience\_ID

WHERE Mil\_Target\_Audience.Target\_Audience\_Ethnicity = 'Azerbaijani'

GROUP BY Mil\_IO\_Task.IO\_Task\_Name

order by count (mil\_operations.io\_task\_Name) DESC



From these two displays, it is very difficult to determine anything other than the fact that a lot more activities are being conducted in Azerbaijan compared to Estonia. Are there reasons for this that is not being captured in our data? What are the conditions in Azerbaijan? Perhaps this is outside the scope of our project or require a multi-disciplinary approach / more objective assessments.

We must dive deeper into what it means to “be Estonian.” In the following processes, we will write code to count the number of motivations, grievances and beliefs that Estonian target audiences possess and compare them to Azerbaijani audiences to see if there is a correlation between the number of grievances, beliefs, motivations and the effectiveness of our teams.

1. What socio-economic, diplomatic and developmental conditions are necessary for USG IO to thrive in a particular country?

--drill down on beliefs, motivations and grievances

SELECT Mil\_Target\_Audience.Target\_Audience\_Name as TA\_Name

, Mil\_TA\_Beliefs.Belief\_Text as Beliefs

, Mil\_TA\_Grievances.Grievance\_Text as Grievances

, Mil\_TA\_Motivations.Motivation\_Text as Motivations

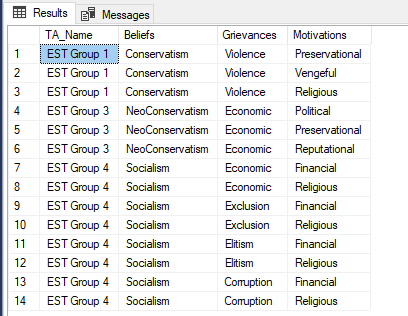
FROM Mil\_Target\_Audience

RIGHT JOIN Mil\_TA\_Grievances on Mil\_TA\_Grievances.target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

RIGHT JOIN Mil\_TA\_Beliefs on Mil\_TA\_Beliefs.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

RIGHT JOIN Mil\_TA\_Motivations on Mil\_TA\_Motivations.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

WHERE Mil\_Target\_Audience.Target\_Audience\_Ethnicity = 'estonian'



/\* In this particular case, it is easy to determine that there are only 3 beliefs for

groups in Estonia, however, if we had a much larger dataset, we would be able to

write code which lists the beliefs and counts the number of target audiences with a particular ethnicity

\*/

/\* We will now take the top country for further analysis, Estonia. We want to determine what operations are occurring in that countries and what the beliefs, motivations and grievances of the target audiences in that countries consist of.

\*/

--estonian motivations

SELECT Mil\_Motivations.Motivation\_Text as motivations

, count (Mil\_TA\_Motivations.Motivation\_Text) as CountMotivations

FROM Mil\_Motivations

RIGHT JOIN Mil\_TA\_Motivations on Mil\_Motivations.Motivation\_ID = Mil\_TA\_Motivations.Motivation\_ID

RIGHT JOIN Mil\_Target\_Audience on Mil\_Target\_Audience.Target\_Audience\_ID = Mil\_TA\_Motivations.Target\_Audience\_ID

WHERE Mil\_Target\_Audience.Target\_Audience\_Ethnicity = 'estonian'

GROUP BY Mil\_Motivations.Motivation\_Text

--estonian beliefs

SELECT Mil\_Beliefs.Belief\_Text as beliefs

, count (Mil\_TA\_Beliefs.Belief\_Text) as CountBeliefs

FROM Mil\_Beliefs

RIGHT JOIN Mil\_TA\_Beliefs on Mil\_Beliefs.Belief\_ID = Mil\_TA\_Beliefs.Belief\_ID

RIGHT JOIN Mil\_Target\_Audience on Mil\_Target\_Audience.Target\_Audience\_ID = Mil\_TA\_Beliefs.Target\_Audience\_ID

WHERE Mil\_Target\_Audience.Target\_Audience\_Ethnicity = 'estonian'

GROUP BY Mil\_Beliefs.Belief\_Text

--estonian grievances

SELECT Mil\_Grievances.Grievance\_Text as Grievances

, count (Mil\_TA\_Grievances.Grievance\_ID) as Countgrievances

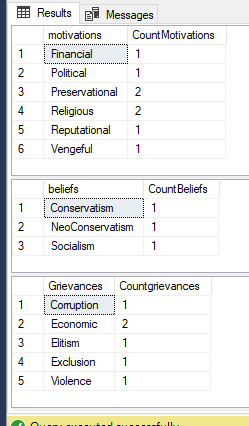
FROM Mil\_Grievances

RIGHT JOIN Mil\_TA\_Grievances on Mil\_Grievances.Grievance\_Id = Mil\_TA\_Grievances.Grievance\_ID

RIGHT JOIN Mil\_Target\_Audience on Mil\_Target\_Audience.Target\_Audience\_ID = Mil\_TA\_Grievances.Target\_Audience\_ID

WHERE Mil\_Target\_Audience.Target\_Audience\_Ethnicity = 'estonian'

GROUP BY Mil\_Grievances.Grievance\_Text



We can see from the above screenshot that our selected Estonian target audience have the following beliefs: Conservatism, NeoConservatism and socialism. These target audiences have Preservation, vengeful, religious, political, reputational and financial motivations. Finally, their grievances are corruption, elitism, exclusion, economic and violence. We are trying to determine if any of these particular traits are more abundant than the others. While we see that two motivations (preservation and religious) are represented across two different groups, there is not a very significant disparity.

Let’s contrast this against the Azerbaijani target audience.

--azerbaijani motivations

SELECT Mil\_Motivations.Motivation\_Text as motivations

, count (Mil\_TA\_Motivations.Motivation\_Text) as CountMotivations

FROM Mil\_Motivations

RIGHT JOIN Mil\_TA\_Motivations on Mil\_Motivations.Motivation\_ID = Mil\_TA\_Motivations.Motivation\_ID

RIGHT JOIN Mil\_Target\_Audience on Mil\_Target\_Audience.Target\_Audience\_ID = Mil\_TA\_Motivations.Target\_Audience\_ID

WHERE Mil\_Target\_Audience.Target\_Audience\_Ethnicity = 'azerbaijani'

GROUP BY Mil\_Motivations.Motivation\_Text

--azerbaijani beliefs

SELECT Mil\_Beliefs.Belief\_Text as beliefs

, count (Mil\_TA\_Beliefs.Belief\_Text) as CountBeliefs

FROM Mil\_Beliefs

RIGHT JOIN Mil\_TA\_Beliefs on Mil\_Beliefs.Belief\_ID = Mil\_TA\_Beliefs.Belief\_ID

RIGHT JOIN Mil\_Target\_Audience on Mil\_Target\_Audience.Target\_Audience\_ID = Mil\_TA\_Beliefs.Target\_Audience\_ID

WHERE Mil\_Target\_Audience.Target\_Audience\_Ethnicity = 'azerbaijani'

GROUP BY Mil\_Beliefs.Belief\_Text

--azerbaijani grievances

SELECT Mil\_Grievances.Grievance\_Text as Grievances

, count (Mil\_TA\_Grievances.Grievance\_ID) as Countgrievances

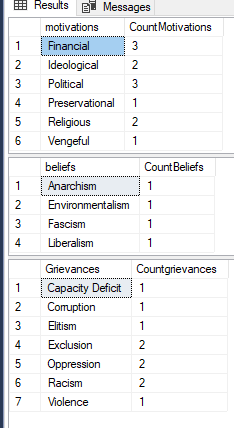
FROM Mil\_Grievances

RIGHT JOIN Mil\_TA\_Grievances on Mil\_Grievances.Grievance\_Id = Mil\_TA\_Grievances.Grievance\_ID

RIGHT JOIN Mil\_Target\_Audience on Mil\_Target\_Audience.Target\_Audience\_ID = Mil\_TA\_Grievances.Target\_Audience\_ID

WHERE Mil\_Target\_Audience.Target\_Audience\_Ethnicity = 'azerbaijani'

GROUP BY Mil\_Grievances.Grievance\_Text



I think that when we look at these two sets, something becomes clear: This approach to problem solving does not in itself answer our business questions. Azerbaijan seems to have more motivating factors and grievances across the target audience group. Azerbaijani target audiences are motivated by multiple factors. Does this suggest that it might be more difficult to truly mobilize this population because there is a lack of focused motivation which is driving them to action? I believe that with this data, all we can do as a data scientist is inform a multi-disciplinary team of our findings and allow them to delve into the nuance of what each of these factors mean and how they might interplay against each other. If we were to have hundreds or possibly thousands of operations, or a larger set of target audiences, we could possibly draw a more informed conclusion. Additionally, since this data was randomly generated, perhaps there is a nonsensical relation between these characteristics which we are trying to form an opinion on based on our understanding of what each characteristic’s value means.

SELECT Mil\_Target\_Audience.Target\_Audience\_Name as TA\_Name

, Mil\_TA\_Beliefs.Belief\_Text as Beliefs

, Mil\_TA\_Grievances.Grievance\_Text as Grievances

, Mil\_TA\_Motivations.Motivation\_Text as Motivations

FROM Mil\_Target\_Audience

RIGHT JOIN Mil\_TA\_Grievances on Mil\_TA\_Grievances.target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

RIGHT JOIN Mil\_TA\_Beliefs on Mil\_TA\_Beliefs.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

RIGHT JOIN Mil\_TA\_Motivations on Mil\_TA\_Motivations.Target\_Audience\_ID = Mil\_Target\_Audience.Target\_Audience\_ID

WHERE Mil\_Target\_Audience.Target\_Audience\_Ethnicity = 'Azerbaijan’

1. Which type of information related capability is having the most general success and to what can we attribute this success?

--count of how many tasks were conducted, by task

select MIl\_Operations.IO\_Task\_Name as TaskConducted

, count (Mil\_operations.IO\_Task\_Name) as NumberofTimesExecuted

FROM MIl\_Operations

GROUP BY MIl\_Operations.IO\_Task\_Name

order by count (mil\_operations.io\_task\_name) desc

--average effectiveness of tasks

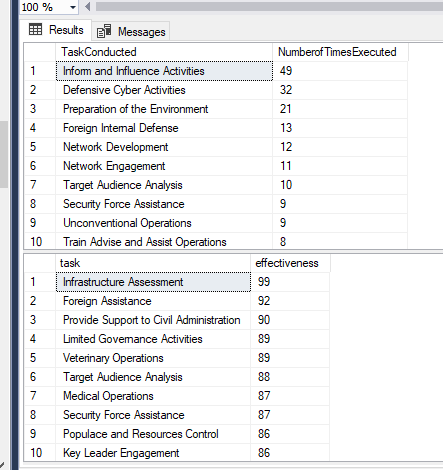
select MIl\_Operations.IO\_Task\_Name as task

, avg (mil\_operations.Opn\_Effect) as effectiveness

FROM MIl\_Operations

group by MIl\_Operations.IO\_Task\_Name

order by avg (Mil\_operations.opn\_effect) DESC



From this selection, we see that our most conducted activity is “Inform and Influence Activities.” This is not, however, our highest average effectiveness. Is this because the other tasks are conducted to a lesser extent and happened to get ‘lucky’ with their high effectiveness rating? A simple select on the three top tasks shows us that out of 320+ activities, 2 were infrastructure assessment, 3 were foreign assistance and 3 were provide support to civil administration. Is it by chance that these have a high effectiveness rating? This is possible. The bigger question is why are we conducting Inform and influence activities if another select will show us that their average effectiveness is only 81?

--average effectiveness of tasks

select MIl\_Operations.IO\_Task\_Name as task

, avg (mil\_operations.Opn\_Effect) as effectiveness

FROM MIl\_Operations

WHERE MIl\_Operations.IO\_Task\_name = 'inform and influence activities'

group by MIl\_Operations.IO\_Task\_Name

order by avg (Mil\_operations.opn\_effect) DESC



1. Are our priority information operations objectives sufficiently supporting the commander’s priority objectives?

To do this, we first identified our commander’s number one priority, used that obj id to determine which IO objectives best support that commander priority, selected those OBJ Ids and counted the number of distinct operations specifically targeted that objective.

--BQ4

Select \* from Mil\_CDR\_OBJ

where CDR\_OBJ\_Priority = '1'

SELECT \* from Mil\_CDR\_IO\_OBJ\_XWalk

WHERE CDR\_OBJ\_ID = '12'

SELECT \* from Mil\_IOOBJ\_Task\_XWalk

WHERE IO\_OBJ\_ID = '16'

or IO\_OBJ\_ID = '7'

or IO\_OBJ\_ID = '14'

or IO\_OBJ\_ID = '3'

SELECT count (Mil\_IOOBJ\_Task\_XWalk.io\_task\_id) as numberofoperations

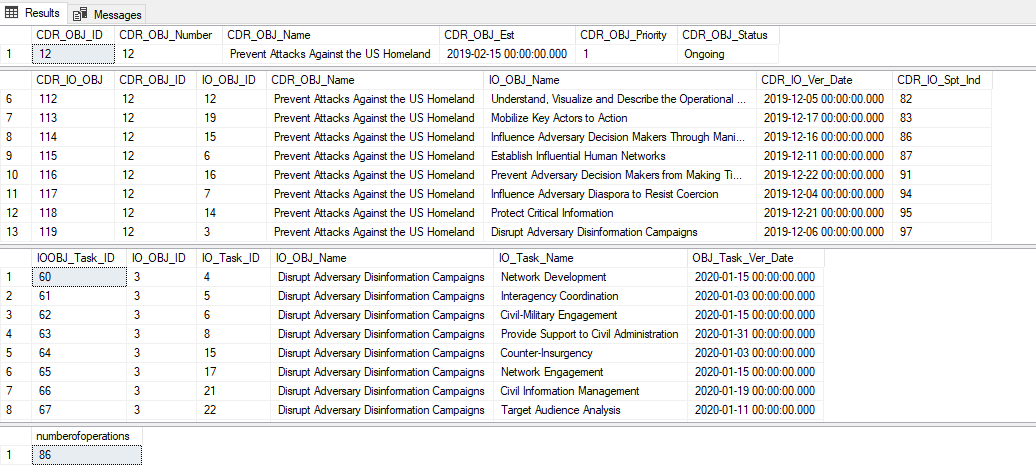
From Mil\_IOOBJ\_Task\_XWalk

WHERE IO\_OBJ\_ID = '16'

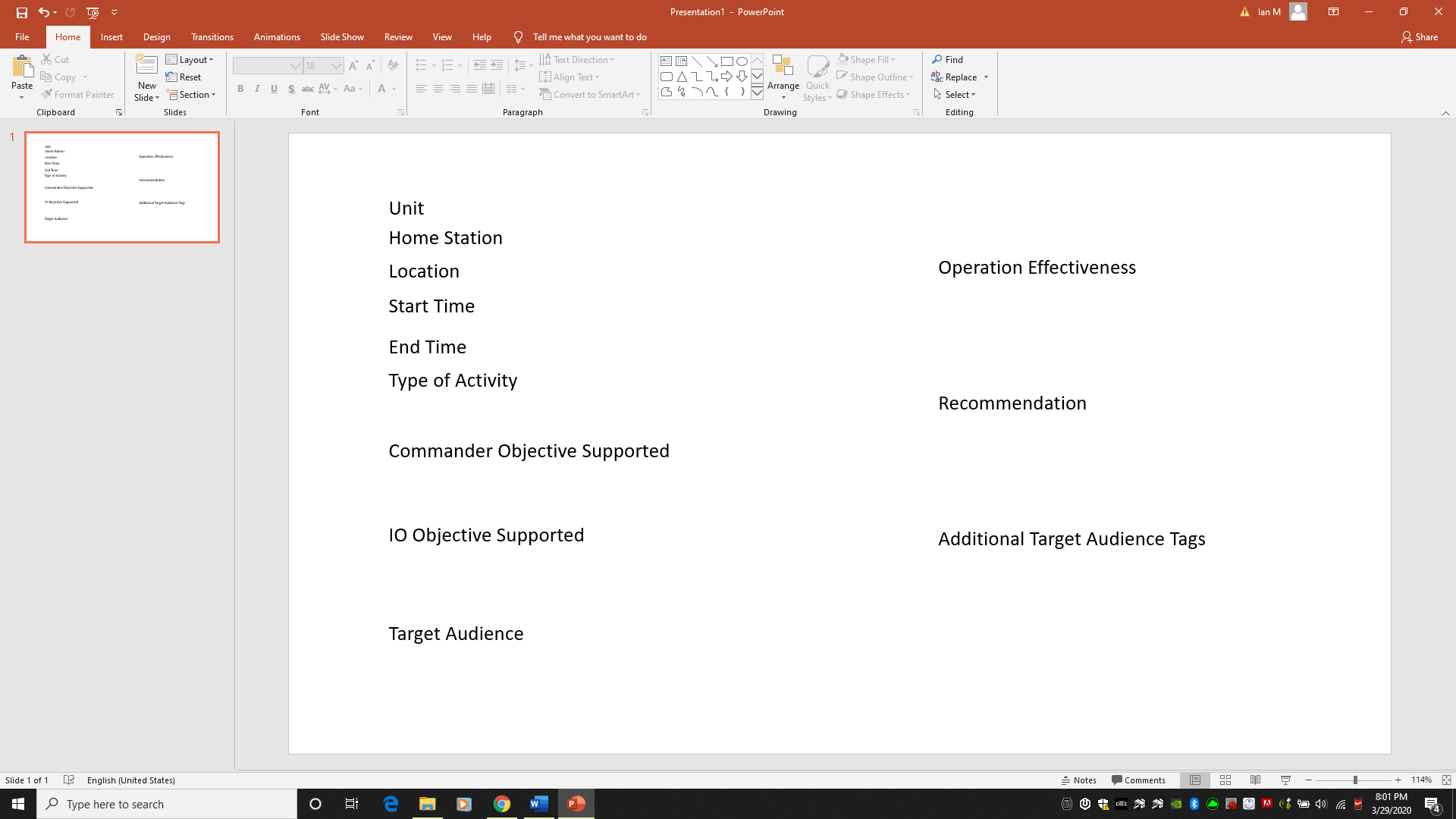
or IO\_OBJ\_ID = '7'

or IO\_OBJ\_ID = '14'

or IO\_OBJ\_ID = '3'



This process illustrates that we are conducting 86 out of 320+ total operations which should have a direct effect on achieving the Commander’s number one priority of “Preventing Attacks Against the US Homeland.” Given that every task is nested in some way to one of the 12 commander’s objectives, and knowing that approximately ¼ of all total operations are targeting the number 1 priority one could say that our operations are properly targeting the commander’s priorities. This is information for the stakeholders involved. If it was determined that the commander required a higher percentage of all activities to target specific priorities, he could use this information above to instruct his staff as to why they need to adjust course in planning and operations.



Above is a simple view of how a team could enter information on the go to inform the database. The intent is not for this to be a flashy application as units are required to access the front-end application in austere environments on encrypted Blackberry-type phones with limited data. A similar version should be able to be accessed from a more permanent structure an exported for further analysis by intelligence analysts or other staff planners.

**Reflections**

I will never undertake another project that requires me to create my own data entirely. There are several problems with taking this approach: it is time consuming, it can at times be contradictory, it may be difficult to make sense of your data. I used Mockaroo.com to generate some data and the biggest problem that I found in doing this was that my target audiences had random characteristics that contradicted themselves. For example, it does not make sense for an environmentalist group to have financial motivations that are driven by persecution. Perhaps that could make sense in some reality, but aside from taking note that this is what the data had presented me, it made it difficult to answer my business questions.

The Logical ERD was more important than I could have imagined. I wanted to begin my project by creating the ERD and then generate the data. It is very easy to overlook relationships, fail to normalize or create relationships that might not make sense in a physical database. I tried to generate my data based on the ERD, but had to go back in the middle of the project to adjust the ERD, create new tables and completely new relationships when I ran into problems. This is not ideal. In the future, I will spend much more time thinking about how my tables will relate between each other and ensuring everything is normalized so that I don’t have difficulties updating or joining tables.

I ran into many problems trying to show how low-level (tactical) “IO Tasks” supported broader (strategic-national) objectives. The intent was that USG objectives would inform commander objectives, would inform IO objectives and drive IO Tasks. Perhaps due to programming inexperience, or other circumstances, I found it very difficult to join these tables together to display which IO Task best supported the number one priority commander or USG objective. I also found that when I joined these tables together, that if I had IO Tasks that supported multiple IO objectives, I would get displays with many duplicate rows. For example, an operation might have 1 IO task. That task might support 5 different IO objectives and transitively 10 different commander objectives. Therefore, my select statements would show the same IO task 5 times, 10 different IO objectives. It was very confusing and it drew my attention away from generating procedures when I could not get the select statements to work properly. This is why my project is mostly comprised of select statements.

I think I had a lot of “scope-creep” in this project or possibly just undertook something that was a bit more complicated than I was ready to handle.