Joining CDW Back Together

Joining CDW Tables Continued

By Margaret Gonsoulin

October 24, 2016

Thanks!

- Richard Pham
- Mark Dean
- Andy Kelly

By the end of this talk,

We hope that you will:

- Be able to identify the correct linking keys
- Be able to incorporate some of the “best practices” for working with CDW into their queries
- Understand the most common types of joins one can use in Structured Query Language (SQL)
- Apply that logic to joining tables in CDW within and across CDW Domains

VIReC
RESEARCHERS' GUIDE TO VA DATA

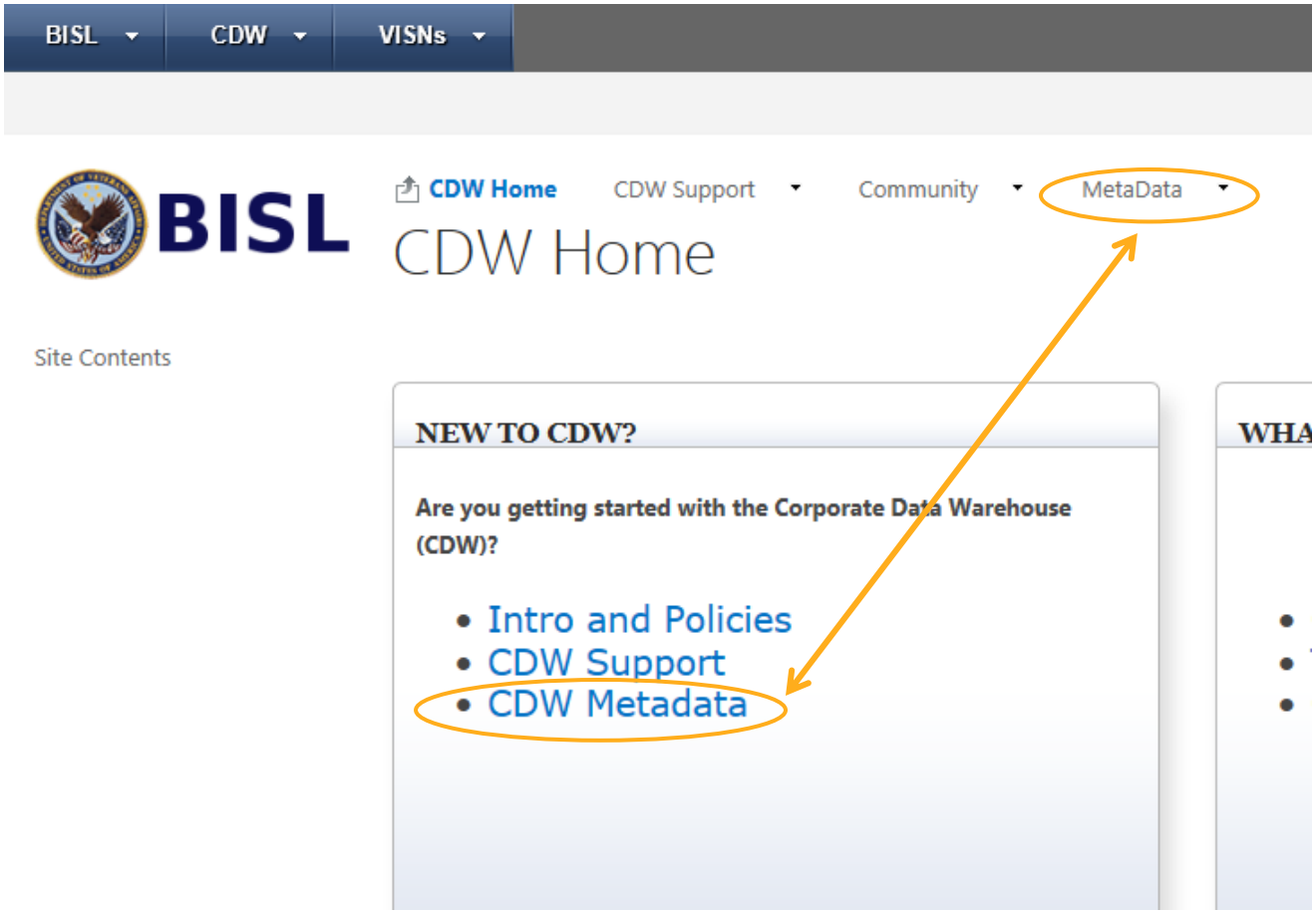
By the end of this talk,

We hope that a new CDW user will:

- **<u>Be able to identify the correct linking keys</u>**
- Be able to incorporate some of the "best practices" for working with CDW into their queries
- Understand the most common types of joins one can use in Structured Query Language (SQL)
- Apply that logic to joining tables in CDW within and across CDW Domains

VIReC
RESEARCHERS' GUIDE TO VA DATA

3 methods of identifying joining keys



https://vaww.cdw.va.gov/Pages/CDWHome.aspx

Click “execute the metadata report”

Site Actions | Browse | Page

United States
Department of Veterans Affairs

MetaData

CDW Home | CDW Support | Community | Domain Teams | MetaData | TechTeam

MetaData Home
Libraries
Metadata Documents
Reports - SSRS
Data Sources

Reports
Schemas
VINCI Central

All Site Content

Execute the MetaData Report

Metadata Documents

Type	Name	Modified	Modified By
	Allergy 1.0	2/19/2014 6:15 PM	Anderson, Stephen M. (CDW)
	Appointment 2.0	2/19/2014 4:25 PM	Anderson, Stephen M. (CDW)
	Consult 2.1	10/1/2014 2:36 PM	Dean, Mark A.
	Dental 1.0	2/19/2014 5:52 PM	Anderson, Stephen M. (CDW)
	DSS	2/19/2014 5:59 PM	Anderson, Stephen M. (CDW)
	ICD-10	2/19/2014 5:51 PM	Anderson, Stephen M. (CDW)
	Inpatient 1.0	2/19/2014 4:22 PM	Anderson, Stephen M. (CDW)
	Inpatient 2.0	2/19/2014 4:22 PM	Anderson, Stephen M. (CDW)

VIReC
RESEARCHERS' GUIDE TO VA DATA

CDW Home > MetaData > Reports

Actions ▾ 1 of 1 Find Next 100%

CDW Metadata

Contains a grouped list of available CDW ER Diagrams and members.

ImageDescription	
⊞ Allergy 1.0	Image Date: 01 Feb 2014
⊞ Appointment 2.0	Image Date: 03 Jun 2015
⊞ Consult 2.1	Image Date: 24 Sep 2015
⊞ CPRSOrder 1.0	Image Date: 11 Aug 2014
⊞ Data Profiling 1.0	Image Date: 21 Feb 2014
⊞ Dental 1.0 Diagram 1 of 2	Image Date: 28 Oct 2015
⊞ Dental 1.0 Diagram 2 of 2 for Analytics	Image Date: 28 Oct 2015
⊞ Dimensions A Through D 7/8/2015	Image Date: 08 Jul 2015
⊞ Dimensions E Through K 7/8/2015	Image Date: 08 Jul 2015
⊞ Dimensions L Through O 9/24/2015	Image Date: 24 Sep 2015
⊞ Dimensions P Through R 6/30/2015	Image Date: 30 Jun 2015
⊞ Dimensions S Through Z 4/17/2015	Image Date: 17 Apr 2015
⊞ Dimensions, Place	Image Date: 09 Oct 2015
⊞ Encounter 1.0	Image Date: 29 Oct 2013
⊞ Health Factor 2.0	Image Date: 11 Mar 2015
⊞ Health Factor 2.1	Image Date: 15 Sep 2015
⊞ ICD-9-CM and ICD-10-CM	Image Date: 24 Sep 2015
⊞ ICD-9-PCS and ICD-10-PCS	Image Date: 10 Aug 2015
⊞ Immunization 2.1	Image Date: 03 Jun 2015

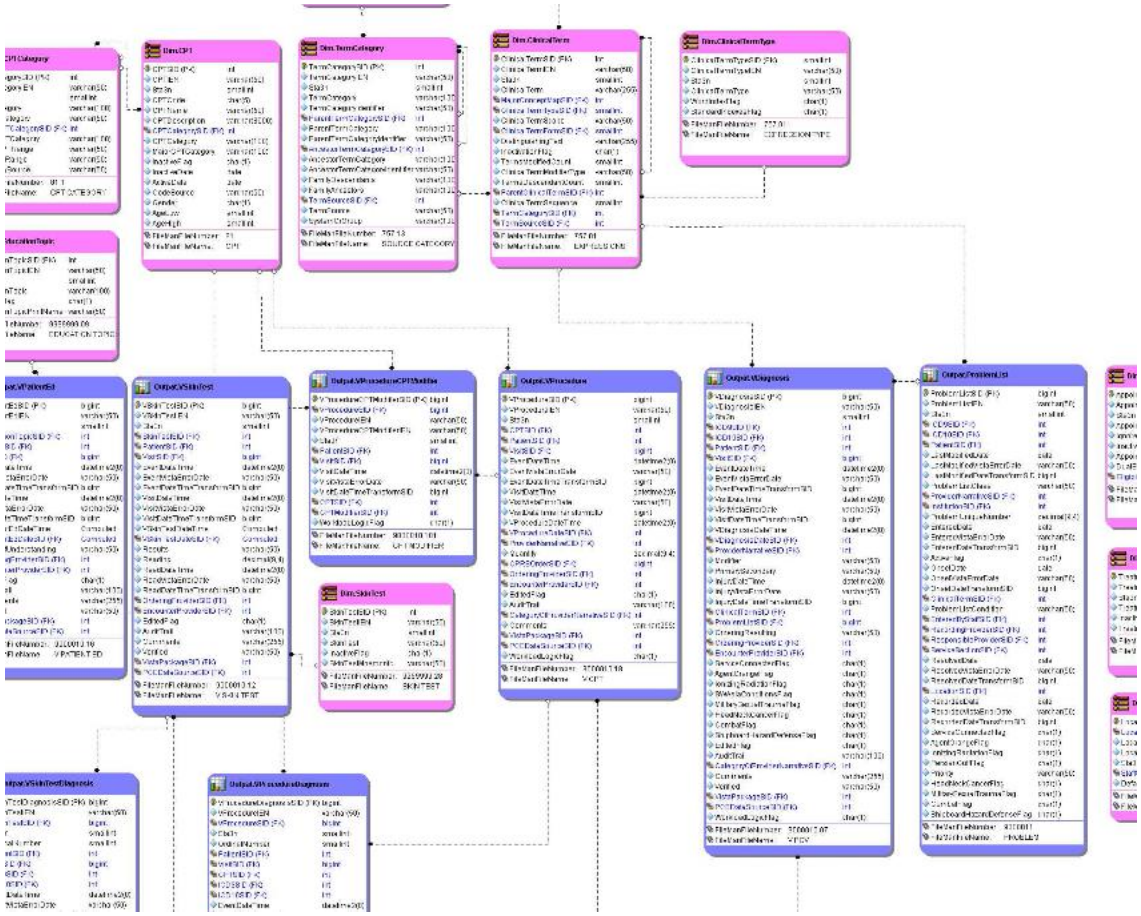
CDW Metadata Report

(Click on the domain name to open ER Diagram)

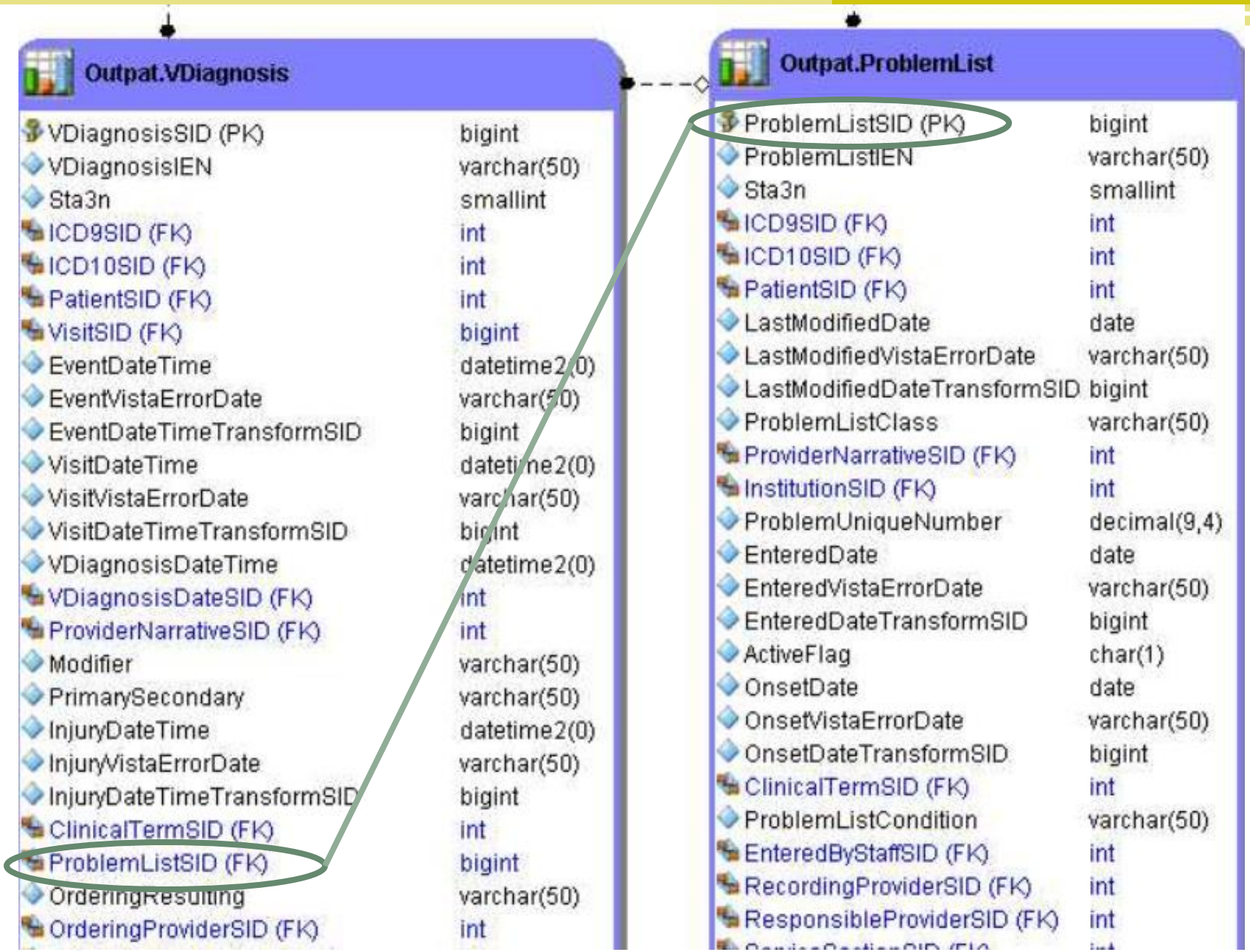
VIReC RESEARCHERS' GUIDE TO VA DATA

1. Use the Entity Relationship (ER) Diagram

(click on image to enlarge)



VIReC
RESEARCHERS' GUIDE TO VA DATA


Outpat.VDiagnosis
VDiagnosisSID (PK) bigint
VDiagnosisIEN varchar(50)
Sta3n smallint
ICD9SID (FK) int
ICD10SID (FK) int
PatientSID (FK) int
VisitSID (FK) bigint
EventDateTime datetime2(0)
EventVistaErrorDate varchar(50)
EventDateTimeTransformSID bigint
VisitDateTime datetime2(0)
VisitVistaErrorDate varchar(50)
VisitDateTimeTransformSID bigint
VDiagnosisDateTime datetime2(0)
VDiagnosisDateSID (FK) int
ProviderNarrativeSID (FK) int
Modifier varchar(50)
PrimarySecondary varchar(50)
InjuryDateTime datetime2(0)
InjuryVistaErrorDate varchar(50)
InjuryDateTimeTransformSID bigint
ClinicalTermSID (FK) int
ProblemListSID (FK) bigint
OrderingResulting varchar(50)
OrderingProviderSID (FK) int
Outpat.ProblemList
ProblemListSID (PK) bigint
ProblemListIEN varchar(50)
Sta3n smallint
ICD9SID (FK) int
ICD10SID (FK) int
PatientSID (FK) int
LastModifiedDate date
LastModifiedVistaErrorDate varchar(50)
LastModifiedDateTransformSID bigint
ProblemListClass varchar(50)
ProviderNarrativeSID (FK) int
InstitutionSID (FK) int
ProblemUniqueNumber decimal(9,4)
EnteredDate date
EnteredVistaErrorDate varchar(50)
EnteredDateTransformSID bigint
ActiveFlag char(1)
OnsetDate date
OnsetVistaErrorDate varchar(50)
OnsetDateTransformSID bigint
ClinicalTermSID (FK) int
ProblemListCondition varchar(50)
EnteredByStaffSID (FK) int
RecordingProviderSID (FK) int
ResponsibleProviderSID (FK) int

2.Use the CDW Metadata Report

Outpatient 2.1

DWViewName	Field Count	FileMan File Data Source	View Version	Relevant Dates	Relationship
Dim.AppointmentStatus	9	APPOINTMENT STATUS (409.63)	DWViewDeployed: xDWWork View Version: 8		
Dim.AppointmentType	9	APPOINTMENT TYPE (409.1)	DWViewDeployed: xDWWork View Version: 13		
Dim.ClinicalTerm	17	EXPRESSIONS (757.01)	DWViewDeployed: xDWWork View Version: 8		
Dim.ClinicalTermType	6	EXPRESSION TYPE (757.011)	DWViewDeployed: xDWWork View Version: 4		
Dim.CPT	16	CPT (81)	DWViewDeployed: xDWWork View Version: 19		
Dim.CPTCategory	10	CPT CATEGORY (81.1)	DWViewDeployed: xDWWork View Version: 2		
Dim.EducationTopic	6	EDUCATION TOPICS (9999999.09)	DWViewDeployed: xDWWork View Version: 7		
Dim.Exam	7	EXAM (9999999.15)	DWViewDeployed: xDWWork View Version: 2		
Dim.LocationProvider	7	PROVIDER (44.1)	DWViewDeployed: xDWWork View Version: 1		
Dim.Protocol	70	PROTOCOL (101)	DWViewDeployed: xDWWork View Version: 4		

VIReC RESEARCHERS' GUIDE TO VA DATA

Click on the Relationships Link…

(for Outpat.ProblemList)

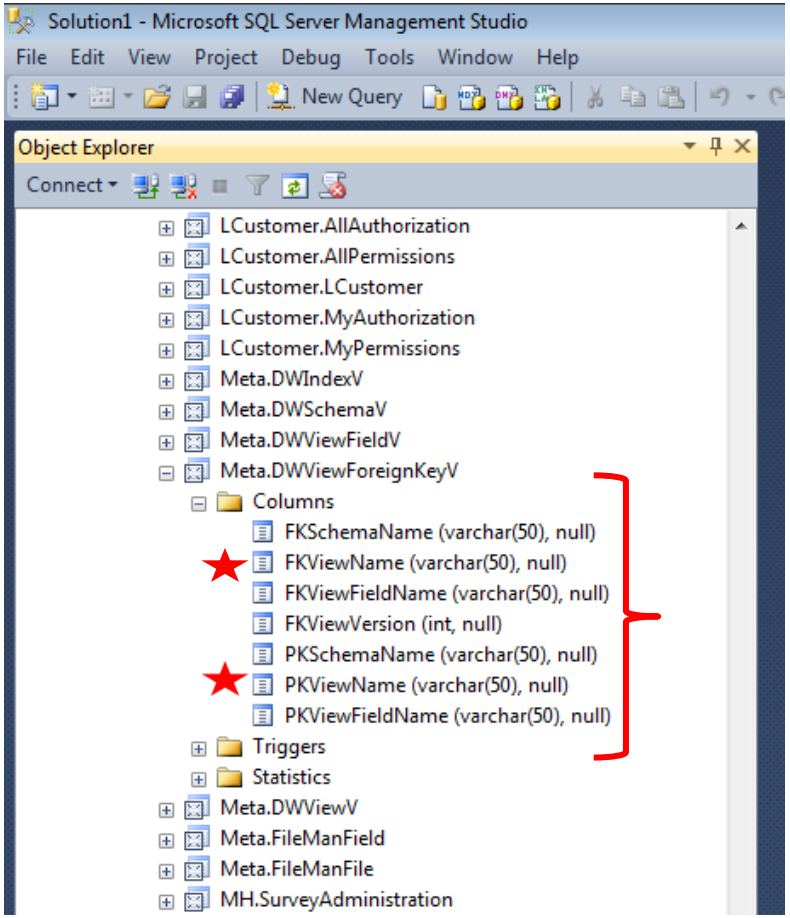
CDW Foreign Keys

Provides a listing of the foreign and primary keys for CDW views.

FKSchemaName	FKViewName	FKViewFieldName	FKViewVersion	PKSchemaName	PKViewName	PKViewFieldName	FKView Status
Outpat	ProblemList	ClinicalTermSID	6	Dim	ClinicalTerm	ClinicalTermSID	ViewPDWWorkComplete
Outpat	ProblemList	EnteredByStaffSID	6	Staff	Staff	StaffSID	ViewPDWWorkComplete
Outpat	ProblemList	ICD10SID	6	Dim	ICD10	ICD10SID	ViewPDWWorkComplete
Outpat	ProblemList	ICD9SID	6	Dim	ICD9	ICD9SID	ViewPDWWorkComplete
Outpat	ProblemList	InstitutionSID	6	Dim	Institution	InstitutionSID	ViewPDWWorkComplete
Outpat	ProblemList	LocationSID	6	Dim	Location	LocationSID	ViewPDWWorkComplete
Outpat	ProblemList	PatientSID	6	Patient	Patient	PatientSID	ViewPDWWorkComplete
Outpat	ProblemList	ProviderNarrativeSID	6	Dim	ProviderNarrative	ProviderNarrativeSID	ViewPDWWorkComplete
Outpat	ProblemList	RecordingProviderSID	6	Staff	Staff	StaffSID	ViewPDWWorkComplete
Outpat	ProblemList	ResponsibleProviderSID	6	Staff	Staff	StaffSID	ViewPDWWorkComplete
Outpat	ProblemList	ServiceSectionSID	6	Dim	ServiceSection	ServiceSectionSID	ViewPDWWorkComplete
Outpat	VDiagnosis	ProblemListSID	24	Outpat	ProblemList	ProblemListSID	ViewPDWWorkComplete

VIReC
RESEARCHERS' GUIDE TO VA DATA

3. Use Meta.DWViewForeignKey



- Use Meta.DWViewForeignKeyV to search for all other tables/views that directly connect to our view of interest.
- The column called FKViewName will contain information about the name of the view(s) containing foreign keys that connect to the view.
- The PKViewName will contain information the names of views that contain primary keys that connect to the view.

VIReC
RESEARCHERS' GUIDE TO VA DATA

Query the Metadata View

- So, we will write a query that looks for connections to the Outpat.ProblemList view.
- We will select all of the columns in this "meta" view with SELECT *
- And, we will use a WHERE to search for information about the view (a.k.a., FKViewName or PKViewName)

```
SELECT *
FROM Database.Schema.Table
WHERE column1 LIKE '------'  OR column2 LIKE '-----' ;
```

VIReC
RESEARCHERS' GUIDE TO VA DATA

Run Query to look at Linking Keys

SQLQuery5.sql - V...hahingonsom (341))*

```sql
SELECT *
FROM CDWWork.Meta.DWViewForeignKey
WHERE FKViewName LIKE 'ProblemList'  OR PKViewName LIKE 'ProblemList' ;
```

100 %

Results | Messages

	FKSchemaName	FKViewName	FKViewFieldName	FKViewVersion	PKSchemaName	PKViewName	PKViewFieldName
1	Outpat	ProblemList	ClinicalTermSID	6	Dim	ClinicalTerm	ClinicalTermSID
2	Outpat	ProblemList	EnteredByStaffSID	6	Staff	Staff	StaffSID
3	Outpat	ProblemList	ICD10SID	6	Dim	ICD10	ICD10SID
4	Outpat	ProblemList	ICD9SID	6	Dim	ICD9	ICD9SID
5	Outpat	ProblemList	InstitutionSID	6	Dim	Institution	InstitutionSID
6	Outpat	ProblemList	LocationSID	6	Dim	Location	LocationSID
7	Outpat	ProblemList	PatientSID	6	Patient	Patient	PatientSID
8	Outpat	ProblemList	ProviderNarrativeSID	6	Dim	ProviderNarrative	ProviderNarrativeSID
9	Outpat	ProblemList	RecordingProviderSID	6	Staff	Staff	StaffSID
10	Outpat	ProblemList	ResponsibleProviderSID	6	Staff	Staff	StaffSID
11	Outpat	ProblemList	ServiceSectionSID	6	Dim	ServiceSection	ServiceSectionSID

Query executed successfully. | VHACDWa01.vha.med.va.gov (1... | VHA12\vhahingonsom (341) | CDWWork | 00:00:00 | 12 rows

By the end of this talk,

We hope that a new CDW user will:

- Be able to identify the correct linking keys
- **Be able to incorporate some of the "best practices" for working with CDW into their queries**
- Understand the most common types of joins one can use in Structured Query Language (SQL)
- Apply that logic to joining tables in CDW within and across CDW Domains

VIReC
RESEARCHERS' GUIDE TO VA DATA

"Best Practices" Reminders…

- When working with large fact tables in CDW, you will want limit the size of your requests for information.
  - SELECT TOP… choose a number
  - WHERE… ask for a specific condition to be met
  - IS NOT NULL… allows you to eliminate any rows where your column of interest has a null value
- Join the dimension tables to the fact tables when possible… put the fact table into the FROM statement and the dimension table into the JOIN statement

VIReC
RESEARCHERS' GUIDE TO VA DATA

Use an Alias (Shortened Table Name)

- You may provide a shortened name to substitute for the table by assigning an "alias" using the AS function in SQL
  - Outpat.ProblemList AS A
  - Dim.ICD9 AS B
- Use that alias the columns and joining keys instead
  - A.ICD9SID , B.ICD9SID
  - A.OnsetDate , B.ICD9Code

USE

- Another useful shortcut is the USE command.
- It allows the user to choose their database at the beginning of the query, so there is not need to repeat it throughout the query.

```
USE Database GO
SELECT column1, column2, column3
FROM Database.Schema.View1
INNER JOIN Database.Schema.View2
                ON LinkingKey1 = LinkingKey2;
```

Read about execution plans…

16 OCTOBER 2012

SQL Server Execution Plans, Second Edition, by Grant Fritchey

Every Database Administrator, developer, report writer, and anyone else who writes T-SQL to access SQL Server data, must understand how to read and interpret execution plans. My book leads you right from the basics of capturing plans, through how to interrupt them in their various forms, graphical or XML, and then how to use the information you find there to diagnose the most common causes of poor query performance, and so optimize your SQL queries, and improve your indexing strategy.

Grant Fritchey ★★★★☆ 27 11

Free eBook download (PDF): Download here.
Buy the printed book: $29.99

Every day, out in the various online forums devoted to SQL Server, and on Twitter, the same types of questions come up repeatedly: Why is this query running slowly? Why is SQL Server ignoring my index? Why does this query run quickly sometimes and slowly at others? My response is the same in each case: have you looked at the execution plan?

https://www.simple-talk.com/books/sql-books/sql-server-execution-plans-second-edition-by-grant-fritchey/

New "Best Practice" – Execution Plans

Click

Hover

Big #?

Thick lines or red "X"s?

```
SELECT *
FROM CDWWork.SPatient.SPatient as a
INNER JOIN CDWWork.Outpat.Visit as b ON a.PatientSID = b.PatientSID;
```

Query 1: Query cost (relative to the batch): 100%

SELECT * FROM CDWWork.SPatient.SPatient as a INNER JOIN CDWWork.Outpat.Visit as b ON a.Patien

Hash Match (Right Outer Join) Cost: 2 %

Clustered Index Scan (Clustered) [Country_v002].[pk_Country__Country… Cost: 0 %

Nested Loops (Left Outer Join) Cost: 1 %

SELECT

Cached plan size	2600 KB
Estimated Operator Cost	0 (0%)
Estimated Subtree Cost	1055570
Estimated Number of Rows	2505970000

Statement
SELECT *
FROM CDWWork.SPatient.SPatient as a
INNER JOIN CDWWork.Outpat.Visit as b ON
a.PatientSID = b.PatientSID

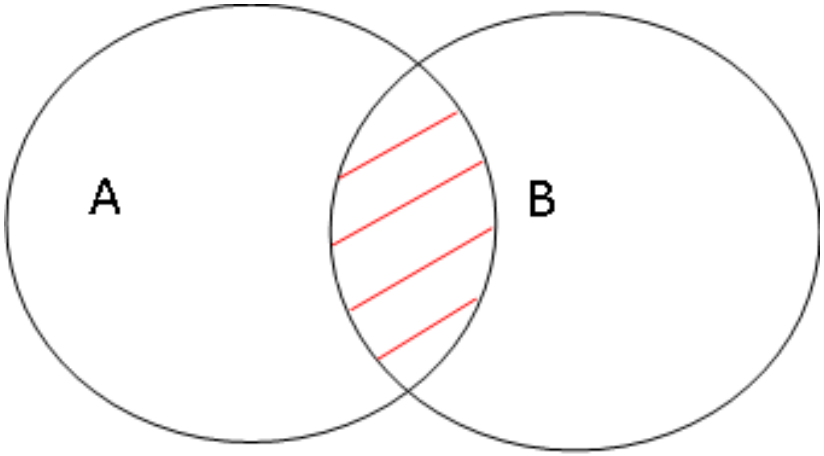
By the end of this talk,

We hope that a new CDW user will:

- Be able to identify the correct linking keys
- Be able to incorporate some of the "best practices" for working with CDW into their queries
- **Understand the most common types of joins one can use in Structured Query Language (SQL)**
- Apply that logic to joining tables in CDW within and across CDW Domains

VIReC
RESEARCHERS' GUIDE TO VA DATA

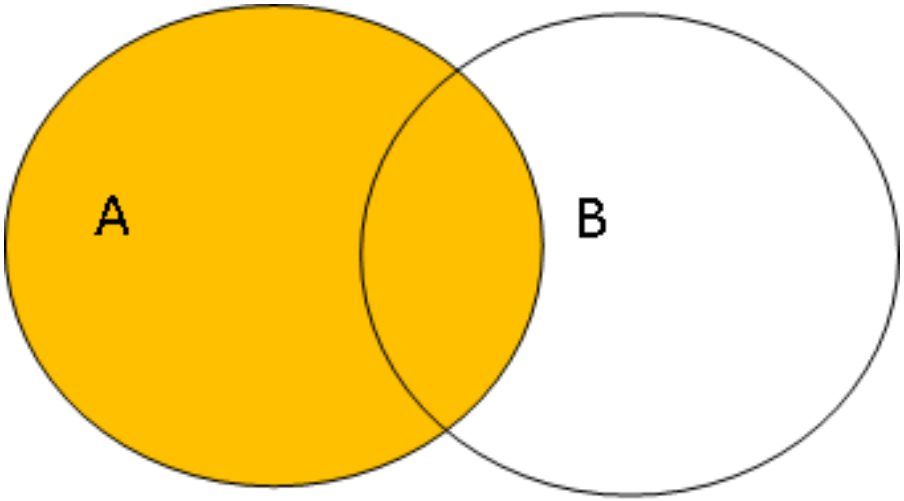
An Inner Join



INNER JOIN

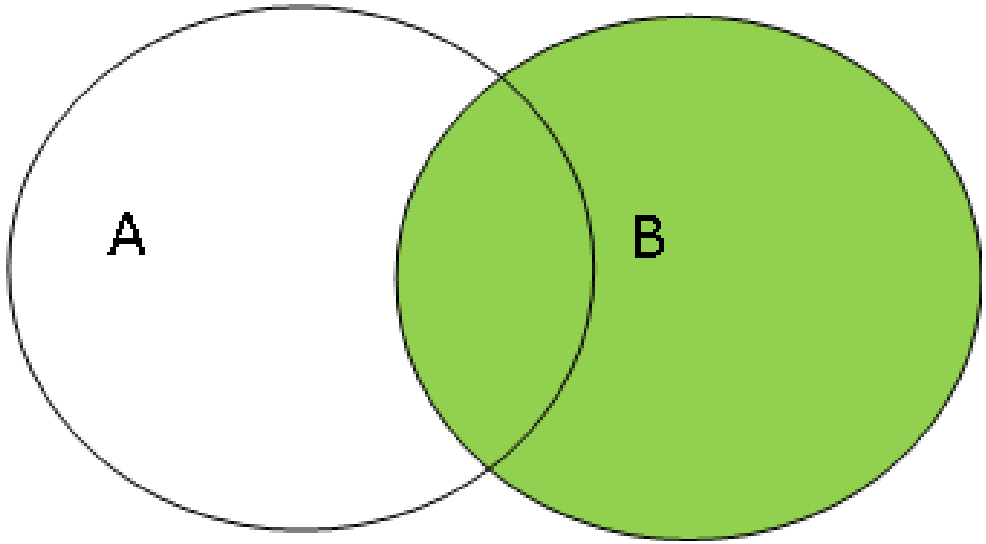
```
SELECT column1, column2
FROM Table 1 AS A
INNER JOIN  Table2 AS B
ON A.Key = B.Key
```

Left Join




```
LEFT JOIN

SELECT column1, column2

FROM Table 1 AS A

LEFT JOIN  Table 2 AS B

ON A.Key = B.Key
```

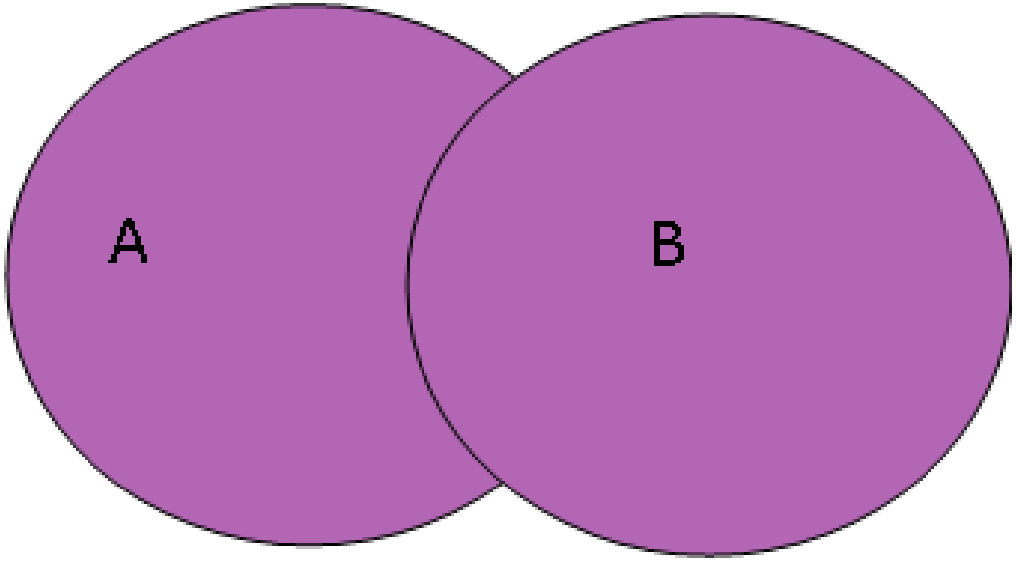
Right Join



RIGHT JOIN

```
SELECT column1, column2
FROM Table 1 AS A
RIGHT JOIN  Table2 AS B
ON A.Key = B.Key
```

Full Outer Join




```
FULL OUTER JOIN

SELECT column1, column2

FROM Table 1 AS A

FULL OUTER JOIN  Table 2 AS B

ON A.Key = B.Key
```

By the end of this talk,

We hope that a new CDW user will:

- Be able to identify the correct linking keys
- Be able to incorporate some of the "best practices" for working with CDW into their queries
- Understand the most common types of joins one can use in Structured Query Language (SQL)
- **Apply that logic to joining tables in CDW within and across CDW Domains**

VIReC
RESEARCHERS' GUIDE TO VA DATA

Practice Problem 1…

- Let's examine the problems that patients report by using:
  - Outpat.ProblemList
  - Dim.ClinicalTerm

Documentation for Outpatient Domain

http://vaww.virec.research.va.gov/CDW/Documentation.htm

Data Documentation

Expand each type of documentation below to view these resources.

[+] **Getting Started with Using CDW**

[−] **NEW! Factbooks**

This product provides descriptions of tables, columns, and values in select CDW Domains including domain-specific SQL "starter language" and sample SQL code.

Domain		Published	Factbooks
Consult	2.1	2016/02	
Inpatient	2.1	2015/10	
Mental Health	1.0	2014/11	
Non-VA Meds	1.0	2016/02	
NEW! Outpatient	2.1	2016/09	
Patient	2.0	2016/05	
Patient Enrollment (with EWL)	1.0	2015/07	

VIReC
RESEARCHERS' GUIDE TO VA DATA

What's in Outpat.ProblemList?

- Outpat.ProblemList
  - Columns
    - ProblemListSID (bigint, not null)
    - ProblemListIEN (varchar(50), not null)
    - Sta3n (smallint, not null)
    - ICD9SID (int, null)
    - ICD10SID (int, null)
    - PatientSID (int, null)
    - LastModifiedDate (date, null)
    - LastModifiedVistaErrorDate (varchar(50), null)
    - LastModifiedDateTransformSID (bigint, null)
    - ProblemListClass (varchar(50), null)
    - ProviderNarrativeSID (int, null)
    - InstitutionSID (int, null)
    - ProblemUniqueNumber (decimal(9,4), null)
    - EnteredDate (date, null)
    - EnteredVistaErrorDate (varchar(50), null)
    - EnteredDateTransformSID (bigint, null)
    - ★ ActiveFlag (char(1), null)
    - ★ OnsetDate (date, null)
    - OnsetVistaErrorDate (varchar(50), null)
    - OnsetDateTransformSID (bigint, null)
    - ★ ClinicalTermSID (int, null)
    - ProblemListCondition (varchar(50), null)
    - EnteredByStaffSID (int, null)
    - RecordingProviderSID (int, null)
    - ResponsibleProviderSID (int, null)
    - ServiceSectionSID (int, null)
    - ResolvedDate (date, null)
    - ResolvedVistaErrorDate (varchar(50), null)
    - ResolvedDateTransformSID (bigint, null)
    - LocationSID (int, null)
    - ★ RecordedDate (date, null)
    - RecordedVistaErrorDate (varchar(50), null)
    - RecordedDateTransformSID (bigint, null)
    - ServiceConnectedFlag (char(1), null)
    - AgentOrangeFlag (char(1), null)
    - IonizingRadiationFlag (char(1), null)
    - PersianGulfFlag (char(1), null)
    - Priority (varchar(50), null)

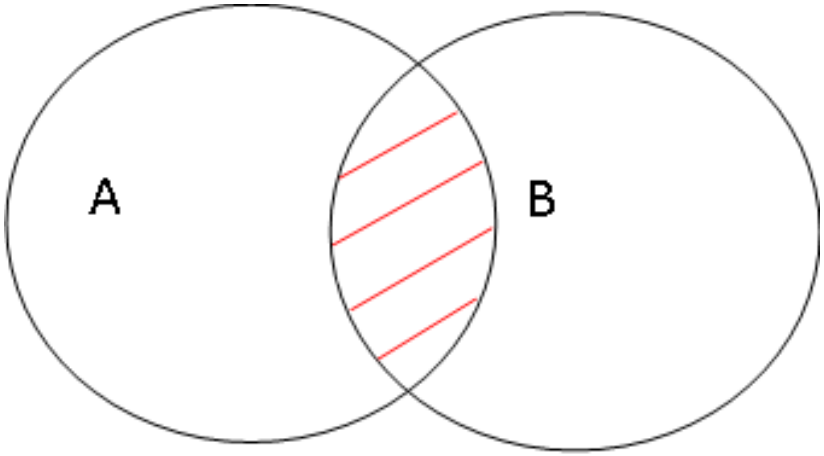
- It contains information about the problems being experienced or reported by a patient including:
  - ✓ Whether or not it is an active problem in the column ActiveFlag
  - ✓ The date of onset of the problem in the column OnsetDate
  - ✓ Date the problem was recorded in the column RecordedDate
  - ✓ A linking key to a Dim.ClinicalTerm, where you will find in the column ClinicalTermSID

What's in Dim.ClinicalTerm

- Dim.ClinicalTerm
  - Columns
    - ★ ClinicalTermSID (int, not null)
    - ClinicalTermIEN (varchar(50), not null)
    - Sta3n (smallint, not null)
    - ClinicalTerm (varchar(255), null)
    - ★ MajorConceptMapSID (int, null)
    - ClinicalTermTypeSID (smallint, null)
    - ClinicalTermScope (varchar(50), null)
    - ClinicalTermFormSID (smallint, null)
    - DistinguishingText (varchar(255), null)
    - InactivationFlag (char(1), null)
    - TermsModifiedCount (smallint, null)
    - ClinicalTermModifierType (varchar(50), n
    - TermsDescendantCount (smallint, null)
    - ParentClinicalTermSID (int, null)
    - ClinicalTermSequence (smallint, null)
    - TermCategorySID (int, null)
    - TermSourceSID (int, null)
  - Triggers

- It contains ~1.3 million clinical terms that are made available to end users to search when entering information about a problem or diagnosis during an outpatient visit.
  - ✓ I'm here to collect the problem reported by the patient that is stored in the column called Clinical Term
  - ✓ I will use the primary key ClinicalTermSID to link to Outpat.ProblemList

An Inner Join



INNER JOIN

```
SELECT column1, column2
FROM Table 1 AS A
INNER JOIN  Table2 AS B
ON A.Key = B.Key
```

Inner Join –

problems reported by patients on 1/4/2016 at our station

```
1  USE CDWWork
2  GO
3  SELECT A.ActiveFlag,
4         B.ClinicalTerm,
5         COUNT (DISTINCT B.ClinicalTerm) AS Freq
6  FROM Outpat.ProblemList as A
7  INNER JOIN Dim.ClinicalTerm AS B
8  ON A.ClinicalTermSID = B.ClinicalTermSID
9  WHERE A.RecordedDate = '1/4/2016' AND A.Sta3n = 578
10 GROUP BY A.ActiveFlag , B.ClinicalTerm;
```

Check your execution plan first

SQLQuery1.sql - V...ahingonsom (1043))*

```
USE CDWWork
GO
SELECT A.ActiveFlag,
       B.ClinicalTerm,
       COUNT (DISTINCT B.ClinicalTerm) AS Freq
FROM Outpat.ProblemList AS A
INNER JOIN Dim.ClinicalTerm AS B ON A.ClinicalTermSID = B.ClinicalTermSID
WHERE A.RecordedDate = '1/4/2016' AND A.Sta3n = 578
GROUP BY A.ActiveFlag, B.ClinicalTerm ;
```

100 %

Messages | Execution plan

```
Query 1: Query cost (relative to the batch): 100%
SELECT A.ActiveFlag, B.ClinicalTerm, COUNT (DISTINCT B.ClinicalTerm) AS Freq FROM Outpat.ProblemList AS A INNER JOIN Dim.Clinical
Missing Index (Impact 91.4238): CREATE NONCLUSTERED INDEX [<Name of Missing Index, sysname,>] ON [Outpat].[ProblemList_v043] ([St
```

SELECT ← Compute Scalar ← Stream Aggregate (Aggregate) Cost: 0 % ← Sort (Distinct Sort) Cost: 0 % ← Nested Loops (Inner Join) Cost: 8 %

Nested Loops ← Clustered Index Seek (Clustered) [ProblemList_v043].[cdx_ProblemList… Cost: 84 %

Nested Loops ← Clustered Index Seek (Clustered) [ClinicalTerm_v023].[pk_ClinicalTer… Cost: 8 %

SELECT

Cached plan size	48 KB
Estimated Operator Cost	0 (0%)
Estimated Subtree Cost	19.3886
Estimated Number of Rows	663.429

Statement

```
SELECT A.ActiveFlag,
       B.ClinicalTerm,
  COUNT (DISTINCT B.ClinicalTerm) AS
Freq
FROM Outpat.ProblemList AS A
```

VIReC
RESEARCHERS' GUIDE TO VA DATA

What problems were reported yesterday at my station?

SQLQuery3.sql - V...hahingonsom (969))*

```sql
USE CDWWork
GO
SELECT A.ActiveFlag,
       B.ClinicalTerm,
       COUNT (DISTINCT B.ClinicalTerm) AS Freq
FROM Outpat.ProblemList as A
INNER JOIN Dim.ClinicalTerm AS B
ON A.ClinicalTermSID = B.ClinicalTermSID
WHERE A.RecordedDate = '1/4/2016' AND A.Sta3n = 578
GROUP BY A.ActiveFlag , B.ClinicalTerm;
```

100 %

Results | Messages

	ActiveFlag	ClinicalTerm	Freq
1	A	*Unknown at this time*	1
2	A	Abdominal aortic aneurysm	1
3	A	Abdominal pain	1
4	A	Abnormal weight loss	1
5	A	Abscess of toe	1
6	A	Absolute anemia	1
7	A	Acneiform eruption	1
8	A	Acquired equinus deformity of foot	1
9	A	Acquired hallux rigidus	1

Query executed successfully. | VHACDWa01.vha.med.va.gov (1... | VHA12\vhahingonsom (969) | CDWWork | 00:00:00 | 301 rows

Practice Problem 2…

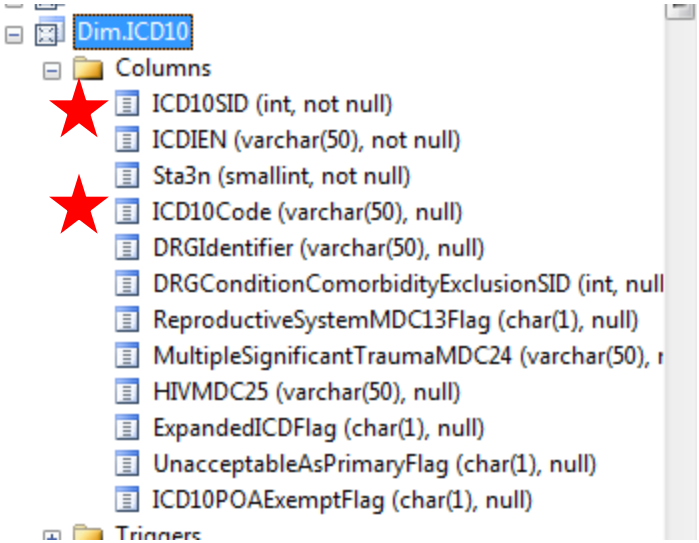
- Let's examine the diagnoses made by physicians by using:
  - Outpat.Vdiagnosis
  - Dim.ICD10

What's in Outpat.VDiagnosis

- Outpat.VDiagnosis
  - Columns
    - VDiagnosisSID (bigint, not null)
    - VDiagnosisIEN (varchar(50), not null)
    - Sta3n (smallint, not null)
    - ICD9SID (int, null)
    - ★ ICD10SID (int, null)
    - PatientSID (int, null)
    - VisitSID (bigint, null)
    - EventDateTime (datetime2(0), null)
    - EventVistaErrorDate (varchar(50), null)
    - EventDateTimeTransformSID (bigint, null)
    - ★ VisitDateTime (datetime2(0), null)
    - VisitVistaErrorDate (varchar(50), null)
    - VisitDateTimeTransformSID (bigint, null)
    - VDiagnosisDateTime (datetime2(0), null)
    - VDiagnosisDateSID (int, null)
    - ProviderNarrativeSID (int, null)
    - Modifier (varchar(50), null)
    - ★ PrimarySecondary (varchar(50), null)
    - InjuryDateTime (datetime2(0), null)
    - InjuryVistaErrorDate (varchar(50), null)
    - InjuryDateTimeTransformSID (bigint, null)
    - ClinicalTermSID (int, null)
    - ProblemListSID (bigint, null)
    - OrderingResulting (varchar(50), null)
    - OrderingProviderSID (int, null)
    - EncounterProviderSID (int, null)
    - ServiceConnectedFlag (char(1), null)
    - AgentOrangeFlag (char(1), null)
    - IonizingRadiationFlag (char(1), null)
    - SWAsiaConditionsFlag (char(1), null)
    - MilitarySexualTraumaFlag (char(1), null)
    - HeadNeckCancerFlag (char(1), null)

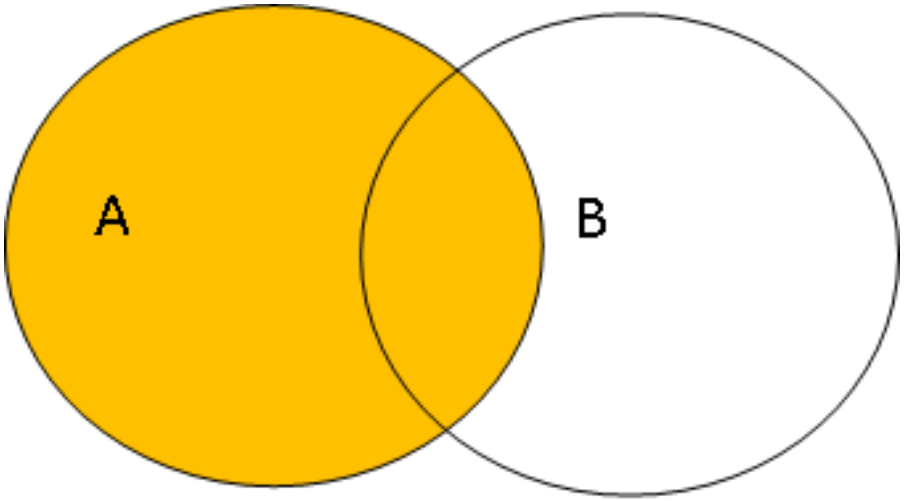
- It contains the provider's definition of what diagnosis to use to represent the patient care given at the visit.
  - ✓ Date and time of the visit in the column VisitDateTime
  - ✓ Whether this diagnosis is considered the primary problem treated in the visit in the column PrimarySecondary
  - ✓ A link to the ICD codes in the column ICD10SID

What's in Dim.ICD10?



- It contains a list of ICD10 codes and a series of information about the nature of those codes:
  - We are here to collect the code stored in the column ICD10Code
  - We will also use the primary key called ICD10SID to link back to Outpat.VDiagnosis

Left Join, we keep all diagnosis records



LEFT JOIN

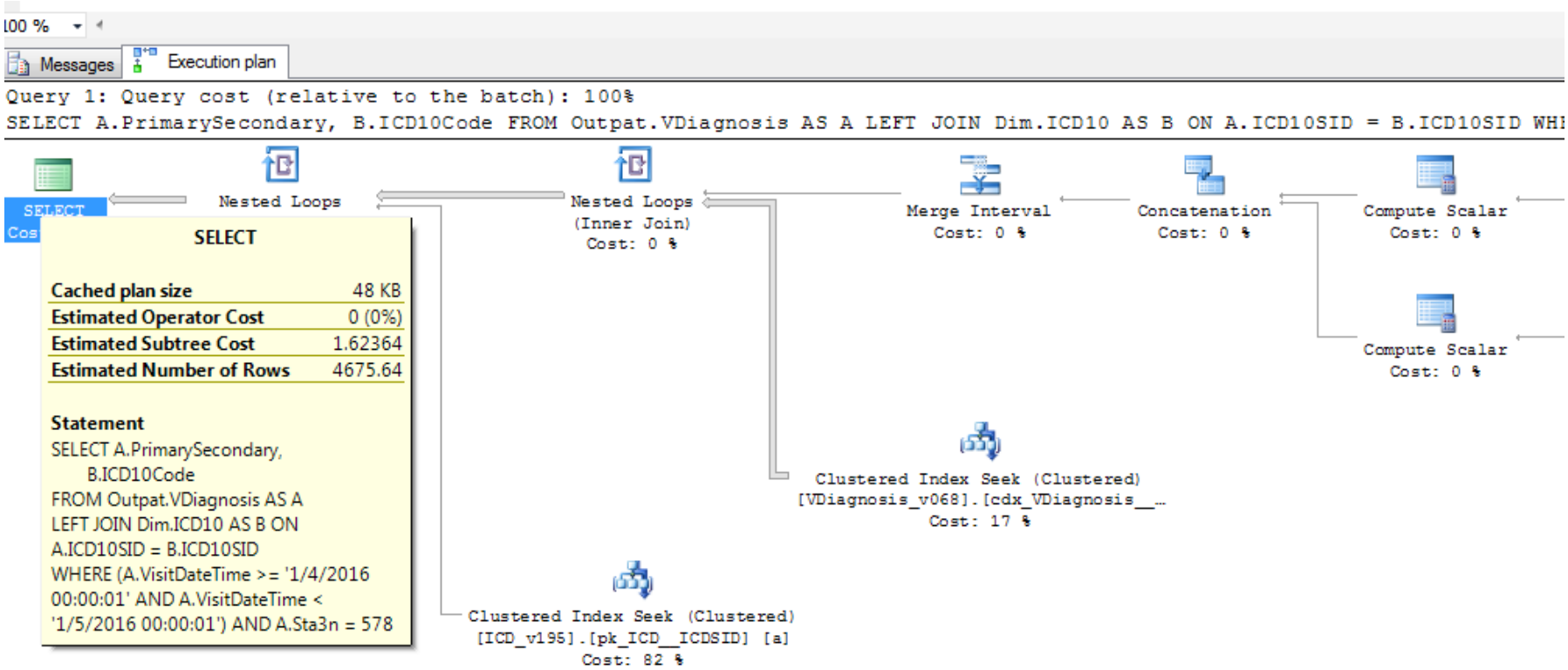
```
SELECT column1, column2
FROM Table 1 AS A
LEFT JOIN  Table2 AS B
ON A.Key = B.Key
```

VIReC
RESEARCHERS' GUIDE TO VA DATA

The top 10 dx so far this month at my station

```
SELECT TOP 10
       a.PrimarySecondary,
        b.ICD10Code,
        count (*) as Freq
FROM Outpat.VDiagnosis as A
LEFT JOIN Dim.ICD10 as B  ON a.ICD10SID = b.ICD10SID
WHERE a.VisitDateTime >= '20161001' AND a.Sta3n = '578'
GROUP BY a.PrimarySecondary , b.ICD10Code
ORDER BY Freq DESC;
```

Check your execution plan first



VIReC
RESEARCHERS' GUIDE TO VA DATA

SQLQuery1.sql - V...AHINGonsoM (919))*

```sql
SELECT TOP 10
       a.PrimarySecondary,
        b.ICD10Code,
        count (*) as Freq
FROM Outpat.VDiagnosis as A
LEFT JOIN Dim.ICD10 as B  ON a.ICD10SID = b.ICD10SID
WHERE a.VisitDateTime >= '20161001' AND a.Sta3n = '578'
GROUP BY a.PrimarySecondary , b.ICD10Code
ORDER BY Freq DESC;
```

100 %

Results | Messages

	PrimarySecondary	ICD10Code	Freq
1	P	Z23.	3097
2	P	Z13.9	3084
3	P	Z71.89	2015
4	S	Z23.	1917
5	P	H54.8	1845
6	S	I10.	1835
7	P	Z51.81	1808
8	P	F43.12	1764
9	S	Z79.01	1677
10	P	N18.6	1674

Example 3

- We will take a quick look at a real life example of a right join

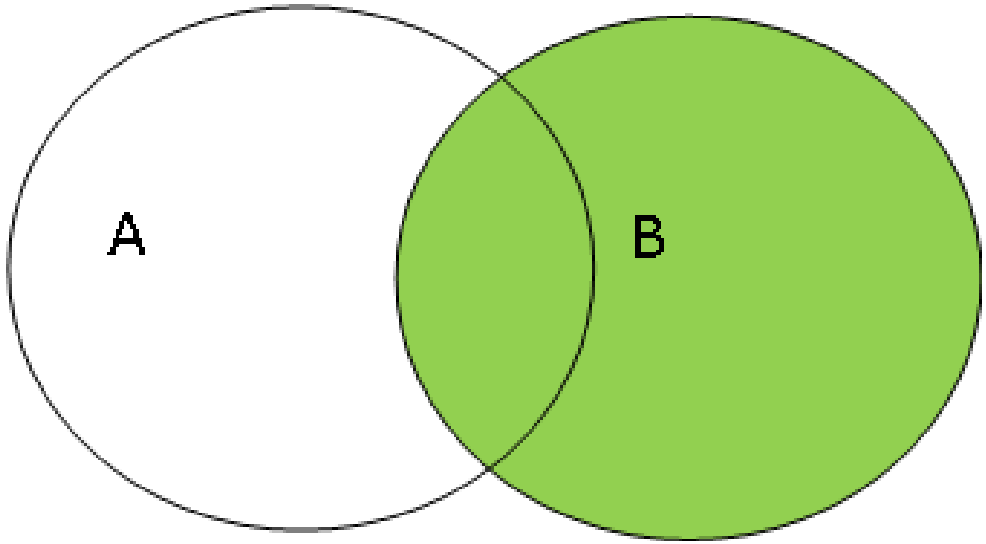
The Researcher's Notebook

Identifying the most recent marital status in CDW

VIReC RESEARCHERS' GUIDE TO VA DATA

http://vaww.virec.research.va.gov/Notebook/Overview.htm

Right Join



RIGHT JOIN

```
SELECT column1, column2
FROM Table 1 AS A
RIGHT JOIN  Table2 AS B
ON A.Key = B.Key
```

Right Join Example

Step 5 | Getting a marital status value for those with no episodes of care

```
/*-----------------------------------------------------------------------------------------------
SELECTING THE MARITAL STATUS FOR PATIENTS WITHOUT A VISIT OR STAY

Individuals with no episode of care will have a NULL value for Sta3n in the first row of each
partition found in the #OrderedCare table.

In the first query we join the table from Step 4 (#OrderedCare) to the Patient Table using a RIGHT
JOIN to retrieve the station number and the registration date for individuals who have not ever had
an episode of care. In the same query, we partition the data by PatientICN and order by descending
registration date (EnteredIntoFileDate). We also flag these records as "Most Recent Registration".

In the second query, we are only choosing the most recent registration for each individual and
storing these records into the table #MostRecentRegistration.
-----------------------------------------------------------------------------------------------*/
SELECT b.PatientICN ,  a.Sta3n as RegistrationStation,
       b.MaritalStatusRecode , a.EnteredIntoFileDate, Flag = 'Most Recent Registration',
       Row_Number() OVER (partition by b.PatientICN order by EnteredIntoFileDate DESC) as RowNumber
INTO #NoCareStation
FROM #Pat as a
RIGHT JOIN #OrderedCare as b on a.PatientICN = b.PatientICN
WHERE b.RowNumber = 1 and b.Sta3n is NULL ;
```

VIReC
RESEARCHERS' GUIDE TO VA DATA

Summary/Conclusions

- There are several methods for identifying linking keys (ER Diagrams, Metadata Report and Metadata views)
- There are a variety of ways to join depending on which parts of the various tables you want to keep (inner, left, right, outer)
- Best practices such as joining dimension tables to fact tables, using aliases and reducing the size of query with WHERE will lead to greater success in working with CDW.

Contact Information

Margaret Gonsoulin, PhD

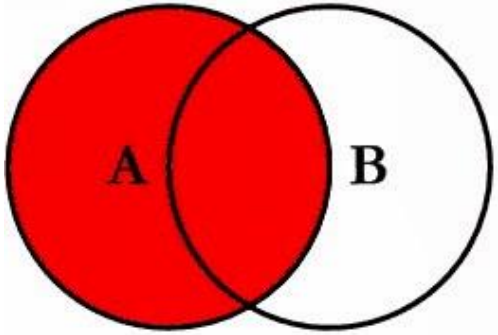
VIReC@va.gov

708-202-2413

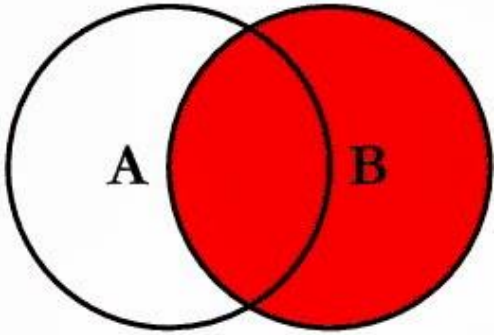
VIReC
RESEARCHERS' GUIDE TO VA DATA

Questions?

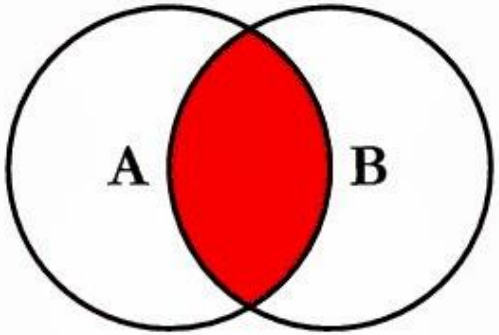
SQL JOINS



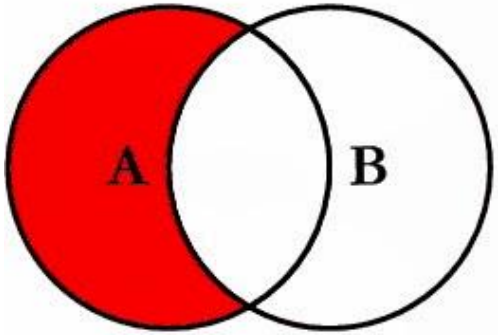

```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
```



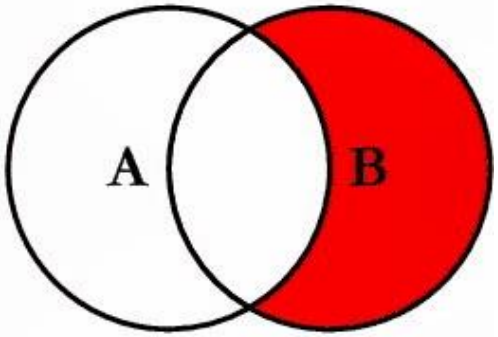

```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
```



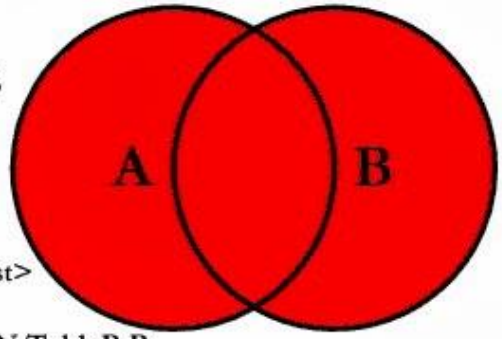

```
SELECT <select_list>
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```



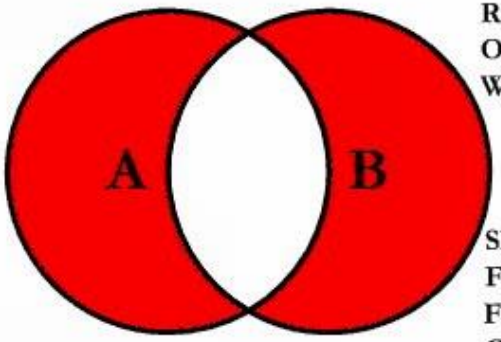

```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```




```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
```




```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```




```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
OR B.Key IS NULL
```