Acmeware

Data Repository Experts Since 1998

The Data Repository Overnight DBA

2018 MUSE International

Educational Session #1104

Thursday, May 31 2:45pm

Presenter: Ian Proffer

Copyright 2020 – Acmeware, Inc.

Today's agenda

What is a DBA?

SQL Server & Management Studio

MEDITECH's expectations for you

Administration: database maintenance, managing growth, user access and security

Analysis: getting data efficiently, monitoring and optimizing performance

Acmeware

Copyright 2020 – Acmeware, Inc.

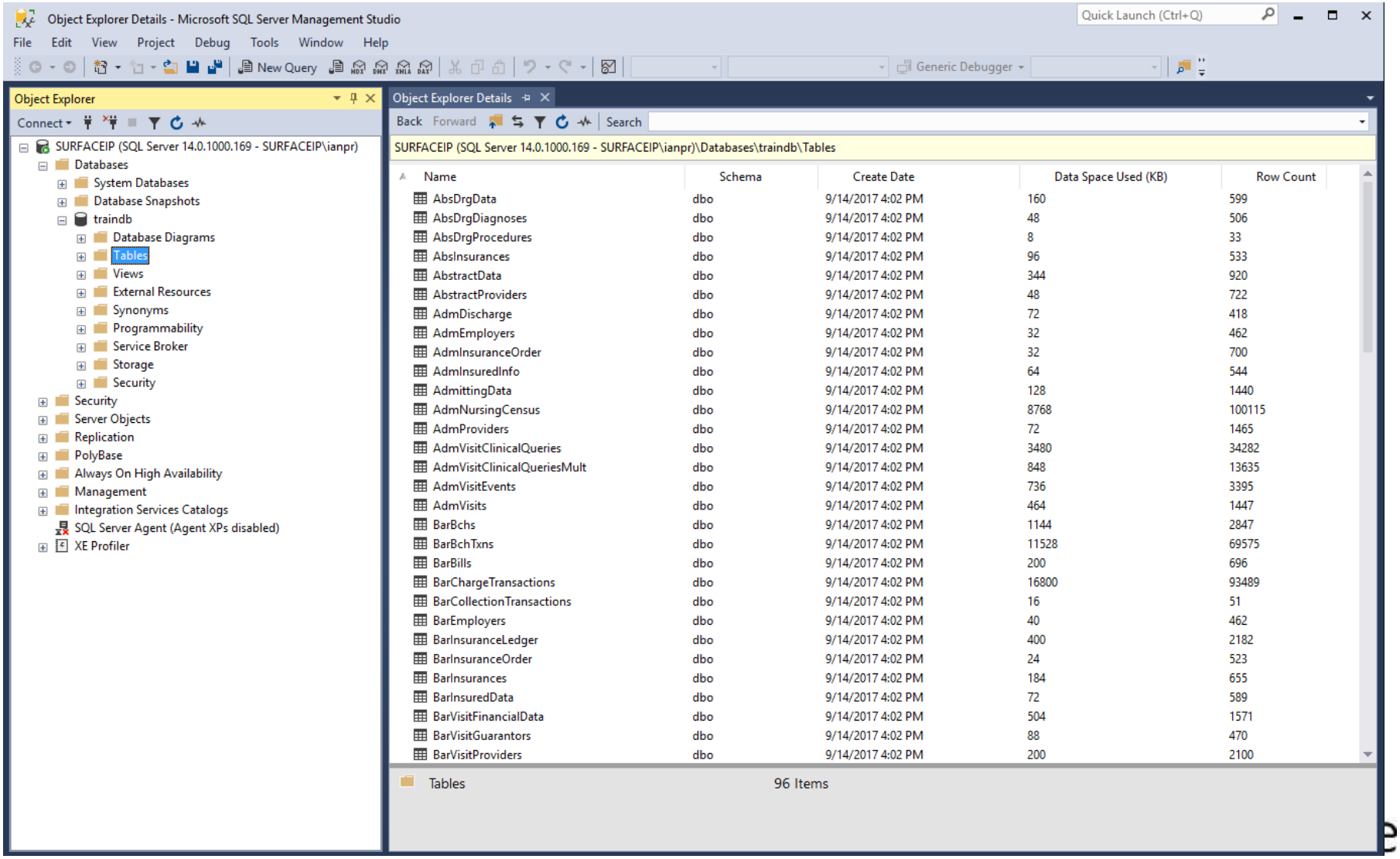
What is a “DBA?”

A. A database administrator
B. A database analyst
C. A database architect
D. An FTE that hasn’t been filled yet
E. A and B (and maybe D)

Acmeware

Copyright 2020 – Acmeware, Inc.

Working with tables in DR



Copyright 2020 – Acmeware, Inc.

Using Management Studio

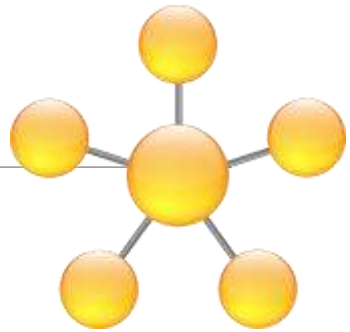
Manage your Microsoft SQL Server instances, including:

- System-wide configuration and security, scheduled job execution
- Create and maintain databases, including security administration (i.e. DBA activities)
- Analyze tables and data
- Write queries to get report results
- Create database programing objects – stored procedures

Acmeware

Copyright 2020 – Acmeware, Inc.

SQL Server & Data Repository



- DR is not a typical database (it's a relational database without relationships)
- C/S and MAGIC 5.67 have 8,000+ tables and 80,000+ data fields
- Expanse/6.1 has 19,000+ tables and 120,000+ data fields in 2 databases
- Security is managed through SQL Server or Windows Active Directory (no connection to MEDITECH security mechanisms)
- DR databases include tables with primary key clustered indexes and stored procedures, but no defined foreign key relationships

Acmeware

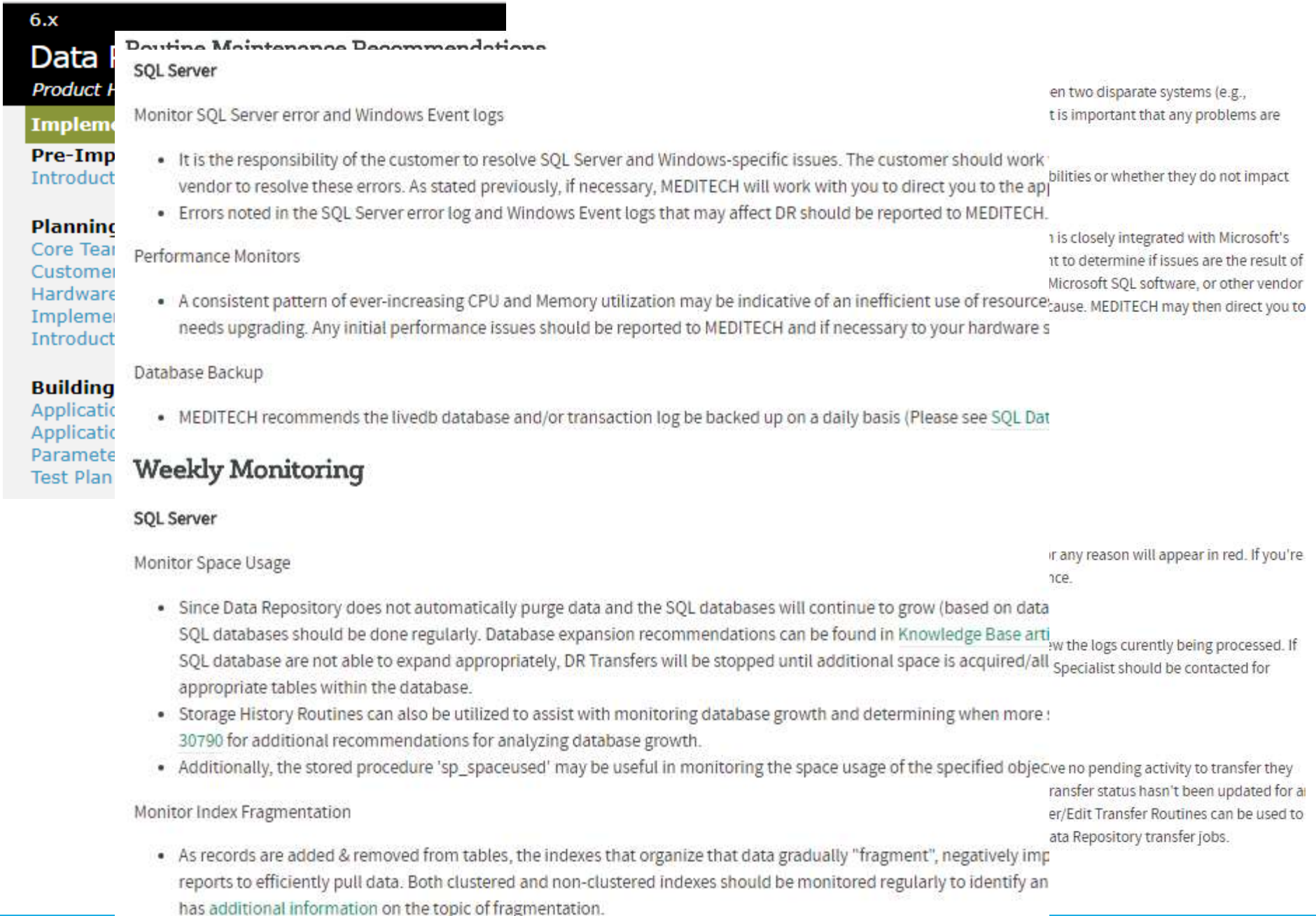
Copyright 2020 – Acmeware, Inc.

MEDITECH Wants You!

To take care of your Data Repository

Acmeware

Copyright 2020 – Acmeware, Inc.



6.x

Data

Product

Implem

Pre-Imp
Introduct

Planning
Core Tea
Custome
Hardware
Implemer
Introduct

Building
Applicatic
Applicatic
Paramete
Test Plan

Routine Maintenance Recommendations

SQL Server

Monitor SQL Server error and Windows Event logs

- It is the responsibility of the customer to resolve SQL Server and Windows-specific issues. The customer should work
vendor to resolve these errors. As stated previously, if necessary, MEDITECH will work with you to direct you to the ap
- Errors noted in the SQL Server error log and Windows Event logs that may affect DR should be reported to MEDITECH.

Performance Monitors

- A consistent pattern of ever-increasing CPU and Memory utilization may be indicative of an inefficient use of resource
needs upgrading. Any initial performance issues should be reported to MEDITECH and if necessary to your hardware s

Database Backup

- MEDITECH recommends the livedb database and/or transaction log be backed up on a daily basis (Please see SQL Dat

Weekly Monitoring

SQL Server

Monitor Space Usage

- Since Data Repository does not automatically purge data and the SQL databases will continue to grow (based on data
SQL databases should be done regularly. Database expansion recommendations can be found in Knowledge Base arti
SQL database are not able to expand appropriately, DR Transfers will be stopped until additional space is acquired/all
appropriate tables within the database.
- Storage History Routines can also be utilized to assist with monitoring database growth and determining when more
30790 for additional recommendations for analyzing database growth.
- Additionally, the stored procedure 'sp_spaceused' may be useful in monitoring the space usage of the specified objec

Monitor Index Fragmentation

- As records are added & removed from tables, the indexes that organize that data gradually "fragment", negatively imp
reports to efficiently pull data. Both clustered and non-clustered indexes should be monitored regularly to identify an
has additional information on the topic of fragmentation.

https://www.meditech.com/prdr/homepageSIX.asp

Server and database administration

Server monitoring & maintenance

- Check the Windows Server and SQL Server logs regularly or when you see symptoms of possible problems
- Perform regular database maintenance, including backups and data integrity checking
- Monitor overall disk space and usage

Acmeware

Copyright 2020 – Acmeware, Inc.

Backups and data integrity

Database backups

- Ideally every database has a full backup done daily during non-business hours
  - Older hardware with very large databases may use differential backups
- Do I need to back up the transaction log?
  - The “Simple” recovery model
  - In the event of a system failure, how quickly can the database be brought back to a given point in time?

Data integrity checking (DBCC)

- Check data integrity regularly (weekly, monthly)
  - DBCC, SQL Maintenance Wizard
- Analyze and address index fragmentation

Acmeware

Copyright 2020 – Acmeware, Inc.

Managing disk and database space

glass = database
water = your data
coaster = disk drive

Acmeware

Copyright 2020 – Acmeware, Inc.

Checking database & file space

1. Management Studio Report

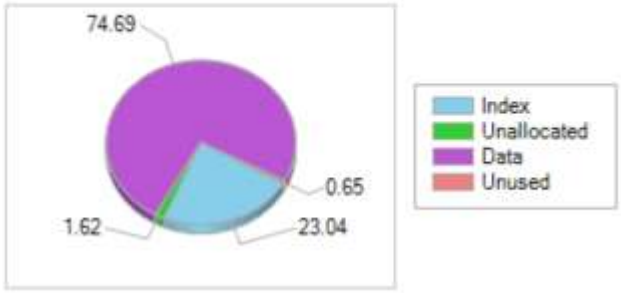
Disk Usage
[Livefdb]
on NMN-DR01 at 5/9/2018 12:32:43 PM

Microsoft SQL Server

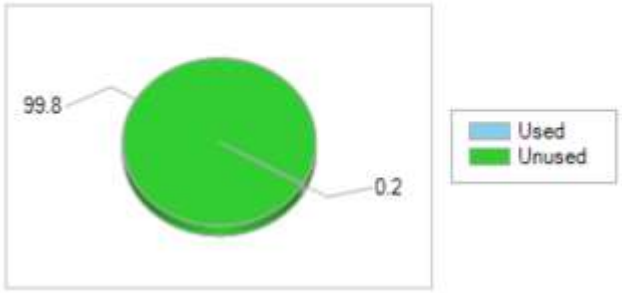
This report provides overview of the utilization of disk space within the Database.

Total Space Usage:	1,072,693.69 MB
Data Files Space Usage:	1,056,627.25 MB
Transaction Log Space Usage:	16,066.44 MB

Data Files Space Usage (%)



Transaction Log Space Usage (%)



Data/Log Files Autogrow/Autoshrink Events

Event	Logical File Name	Start Time	Duration (ms.)	Change In Size (MB)
Data File Auto Growth	Livefdb	5/7/2018 2:33:41 PM	48526	20,718.19

Disk Space Used by Data Files

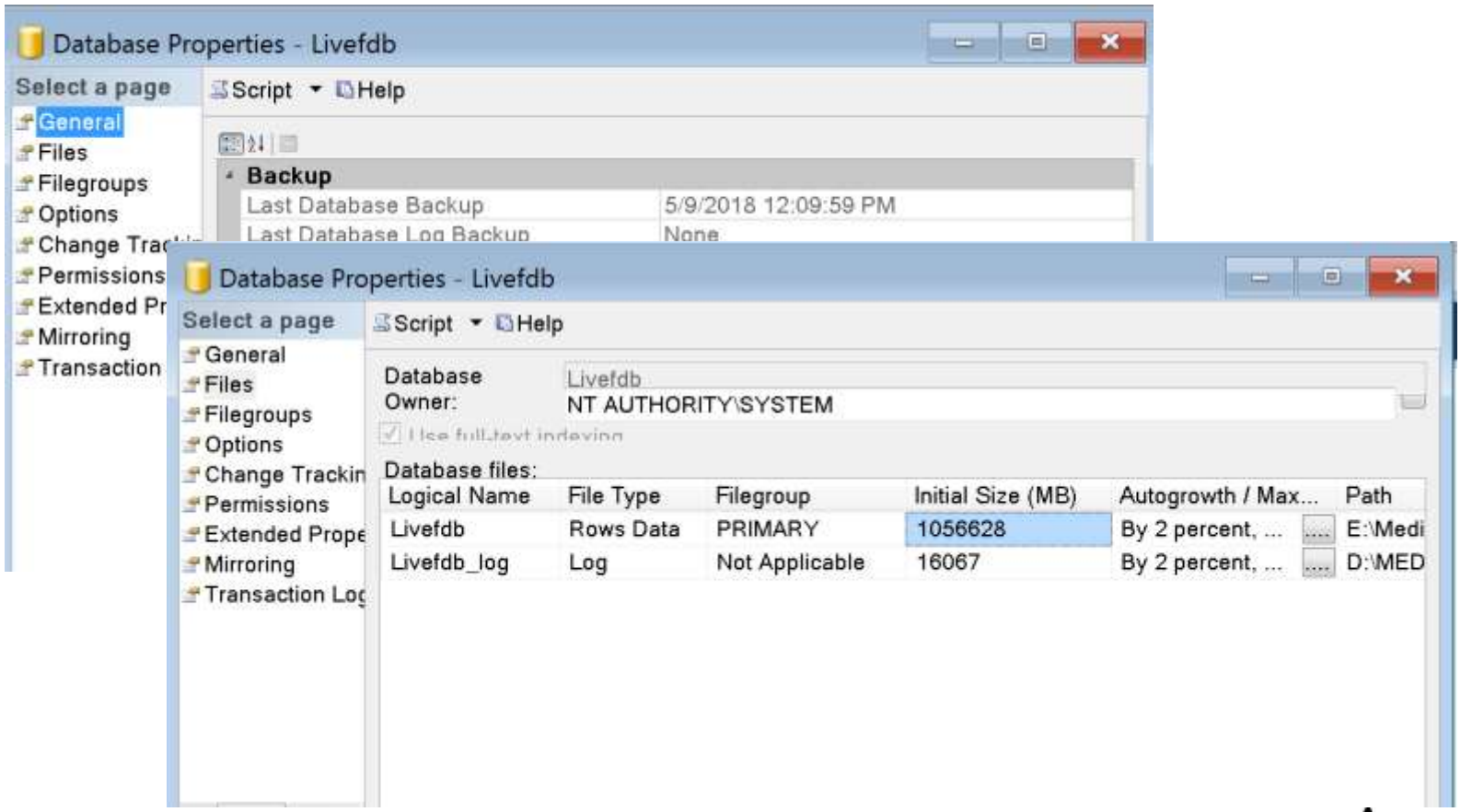
Filegroup Name	Logical File Name	Physical File Name	Space Reserved	Space Used
PRIMARY	Livefdb	E:\MEDIFILES\Livefdb.mdf	1031.86 GB	1015.40 GB

Acmeware

Copyright 2020 – Acmeware, Inc.

Checking database & file space

2. Database Properties



Copyright 2020 – Acmeware, Inc.

Checking database & file space

3. T-SQL sp_spaceused

SQLQuery1.sql - NM...BS\iproffer (157))*

```
USE Livefdb
GO
EXEC sp_spaceused
GO
```

100 %

Results | Messages

	database_name	database_size	unallo
1	Livefdb	1072693.69 MB	1706

	reserved	data	index_
1	1064508752 KB	771125792 KB	28631

SQLQuery1.sql - NM...BS\iproffer (157))*

```
USE Livefdb
GO
EXEC sp_spaceused 'RegAcct_Main'
GO
```

100 %

Results | Messages

	name	rows	reserved	data	index_size	unused
1	RegAcct_Main	2880018	2002352 KB	857896 KB	1142600 KB	1856 KB

Copyright 2020 – Acmeware, Inc.

Checking database & file space

4. Query the system tables directly

SQLQuery1.sql - NM...BS\iproffer (157))*

```
SELECT
    file_id, name, type_desc,
    physical_name, size, max_size
FROM sys.database_files
```

100 %

Results | Messages

	file_id	name	type_de...	physical_name	size	max_size
1	1	Livefdb	ROWS	E:\Medifiles\Livefdb.mdf	135248288	-1
2	2	Livefdb_log	LOG	D:\MEDIFILES\Livefdb_log.ldf	2056504	268435456

Acmeware

Copyright 2020 – Acmeware, Inc.

Managing disk and database space

Disk space vs. database space

- Make sure primary data drive has plenty of capacity for live database growth
- MT recommends you keep 25% free space for the live databases (typically the E: drive)
- Database auto-growth options
  - Percentage vs. amount of space
  - Unrestricted vs. restricted space

What happens when the drive is full?

- Transfers stop (crash) until the db can grow
- Avoid this at all costs!

Acmeware

Copyright 2020 – Acmeware, Inc.

Monitoring growth over time

```
SELECT
    YEAR(DateTimeID) AS Year,
    MONTH(DateTimeID) AS Month,
    SUM(TableDataPages) AS DataPages,
    SUM(TableIndexPages) AS IndexPages
FROM
    Livefdb.dbo.DR_SysStorageHistory
WHERE
    TableNameID = 'RegAcct_Main'
GROUP BY
    YEAR(DateTimeID),
    MONTH(DateTimeID)
ORDER BY
    Year DESC, Month DESC
```

70 %

Results | Messages

	Year	Month	DataPages	IndexPages
2	2018	4	25510032	33160720
3	2018	3	26295000	33428304
4	2018	2	23544280	30473976
5	2018	1	26676568	35840600
6	2017	12	26636200	36137528
7	2017	11	25558920	34506280

DR_SysStorageHistory can be queried to summarize data space usage by date and table. (*SysStorage* in MAGIC and C/S.)

This is helpful if you want to predict future database growth based on the last year of transfer activity.

Acmeware

Copyright 2020 – Acmeware, Inc.

Managing security & access

Create your own database for report objects (stored procedures, functions, views, etc.)

- This avoids co-mingling your code and objects with Meditech's
- This allows you to limit direct access to live databases

Use Active Directory groups for easier administration

- e.g. **ReportAdmins** group, **ReportViewer** group

For those that need access to live data consider:

- Using views in your database to selectively expose data
- Using SQL database-level roles for direct table access
  - **ReportAdmins**: db_owner, db_denydatawriter (read all data from livedb, ability to create indexes but no INSERT, UPDATE or DELETE)
  - **ReportViewers**: db_datareader (SELECT from tables)
  - **ReportViewers_NoPP**: db_datareader with further DENY SELECT permissions*

Acmeware

Copyright 2020 – Acmeware, Inc.

An example of a DENY script

```
SELECT 'DENY SELECT ON ' + name + ' TO ReportViewers_NoPP'
FROM livedb.dbo.sysobjects
WHERE xtype = 'U' AND name LIKE 'Pp%'
ORDER BY name
```

Results | Messages

	(No column name)
1	DENY SELECT ON PpApplicantComments TO ReportViewers_NoPP
2	DENY SELECT ON PpApplicantEducationDetails TO ReportViewers_NoPP
3	DENY SELECT ON PpApplicantEmployeeAddDetails TO ReportViewers_NoPP
4	DENY SELECT ON PpApplicantEmployeeAddresses TO ReportViewers_NoPP
5	DENY SELECT ON PpApplicantEmployeeDetails TO ReportViewers_NoPP

Acmeware

Copyright 2020 – Acmeware, Inc.

Database analysis

Optimizing queries
- Query execution plans

Creating your own indexes

Handling index fragmentation

Other performance tools

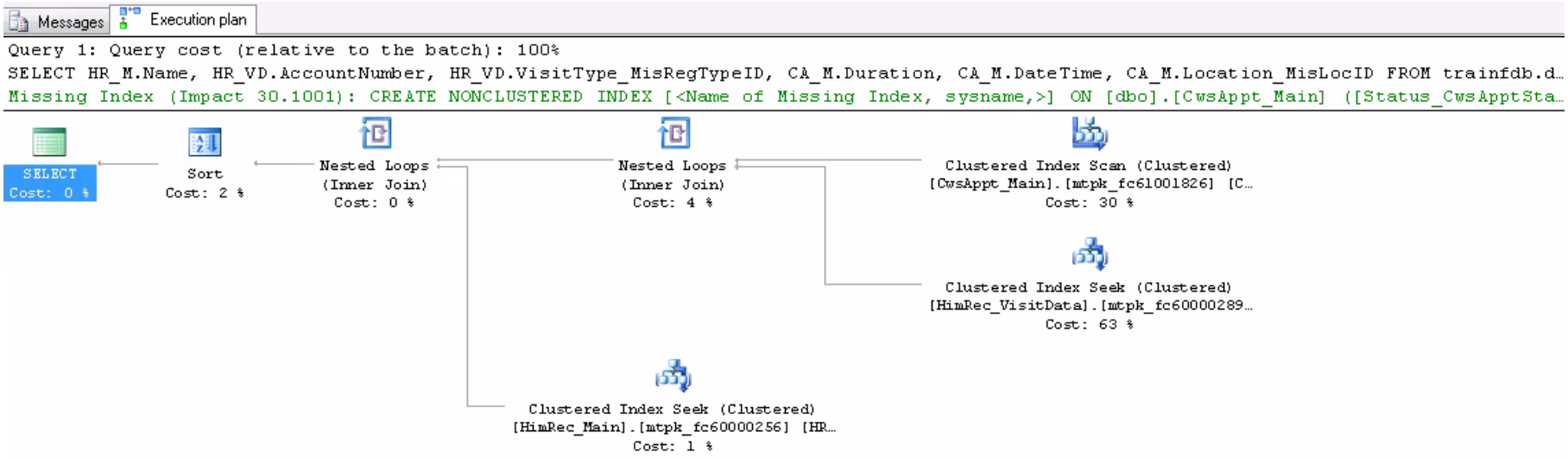
Acmeware

Copyright 2020 – Acmeware, Inc.

Optimizing queries

Viewing the query execution plan

```
SELECT
    HR_M.Name,
    HR_VD.AccountNumber,
    HR_VD.VisitType_MisRegTypeID,
    CA_M.Duration,
    CA_M.DateTime,
    CA_M.Location_MisLocID
FROM trainfdb.dbo.HimRec_Main HR_M
INNER JOIN trainfdb.dbo.HimRec_VisitData HR_VD
    ON HR_M.SourceID = HR_VD.SourceID
```



Copyright 2020 – Acmeware, Inc.

Creating non-clustered indexes

Clustered vs. non-clustered

- All DR tables already have a primary key clustered index. Don't change them!
- Non-clustered indexes can be added wherever you like.

Document and save scripts for your indexes

General guidelines for new non-clustered indexes:

- SourceID, VisitID and SourceID, PatientID in any top-level table where it's not already a primary key
  - AbstractData, BarVisits, LabSpecimens, PhaRx, OeOrders, OmOrd_Main
- Fields with highly discrete data values that are used as report parameters (e.g. DateTime)

Fewer general purpose indexes are better than many purpose-built indexes (indexes consume space and require maintenance)

Acmeware

Copyright 2020 – Acmeware, Inc.

Index fragmentation

The SQL Maintenance Wizard can address this, but…

- Every table in the database is analyzed and acted upon
- Indexes may be dropped and recreated

Doing your own, targeted index maintenance is better

- Assemble a list of tables you use regularly for reports
- If maintenance hasn't been done, address tables selectively on a series of first passes, then on a regular schedule (weekly, monthly, after initial loads)
- After important tables are done, address the others

Limit amount of work done based on a fragmentation threshold or set number of tables (based on size) per session

Acmeware

Copyright 2020 – Acmeware, Inc.

Analyzing & correcting index fragmentation

First evaluate it with a system dynamic management view

```
use Livefdb
go
select
    index_id, index_type_desc, avg_fragmentation_in_percent,
    fragment_count, avg_fragment_size_in_pages, page_count
from sys.dm_db_index_physical_stats
    (DB_ID(N'Livefdb'), OBJECT_ID(N'RegAcct_Main'), NULL, NULL , NULL);
```

70 %

Results | Messages

	index_id	index_type_desc	avg_fragmentation_in_percent	fragment_count	avg_fragment_size_in_pages	page_count
1	1	CLUSTERED INDEX	99.9011673441987	107163	1.00083051053069	107252
2	20	NONCLUSTERED INDEX	99.906611878969	16050	1.0007476635514	16062
3	21	NONCLUSTERED INDEX	99.6913770754892	16155	1.00284741566079	16201
4	22	NONCLUSTERED INDEX	99.4708994708995	15991	1.0046276030267	16065
5	23	NONCLUSTERED INDEX	99.9003611906838	16048	1.00062313060818	16058
6	24	NONCLUSTERED INDEX	99.9439182452642	16043	1.00031166240728	16048
7	26	NONCLUSTERED INDEX	99.9866372686577	14967	1	14967
8	29	NONCLUSTERED INDEX	99.8637992831541	13934	1.00114827041768	13950
9	30	NONCLUSTERED INDEX	99.5087941689114	31555	1	31555

Acmeware

Copyright 2020 – Acmeware, Inc.

Analyzing & correcting index fragmentation

Then correct it. Indexes can be either reorganized or rebuilt:

- ALTER INDEX with REORGANIZE for indexes with 5-30% fragmentation
- ALTER INDEX with REBUILD for indexes with >30% fragmentation

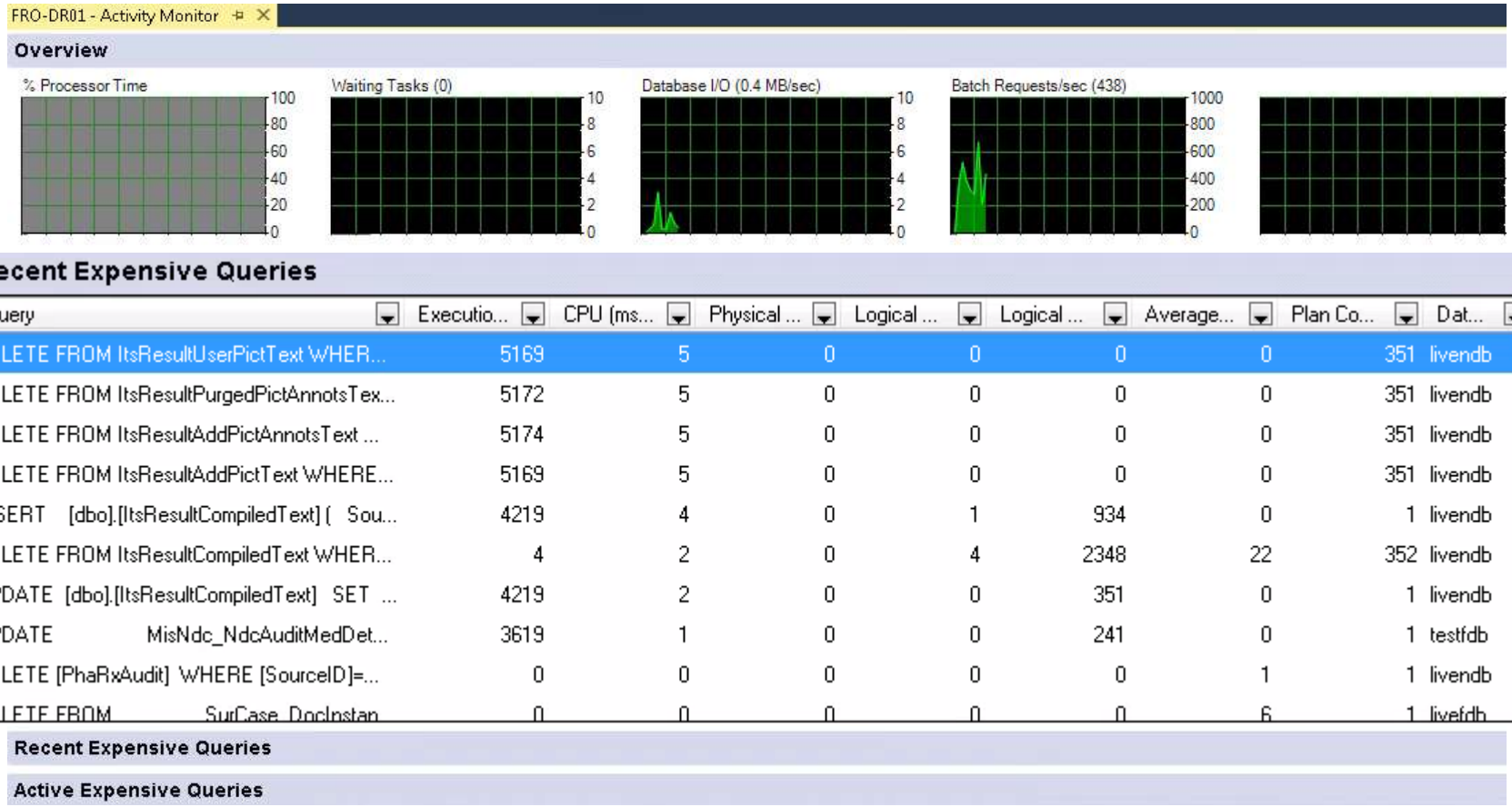
https://docs.microsoft.com/en-us/sql/relational-databases/indexes/reorganize-and-rebuild-indexes?view=sql-server-2016

Acmeware

Copyright 2020 – Acmeware, Inc.

Other performance & monitoring tools

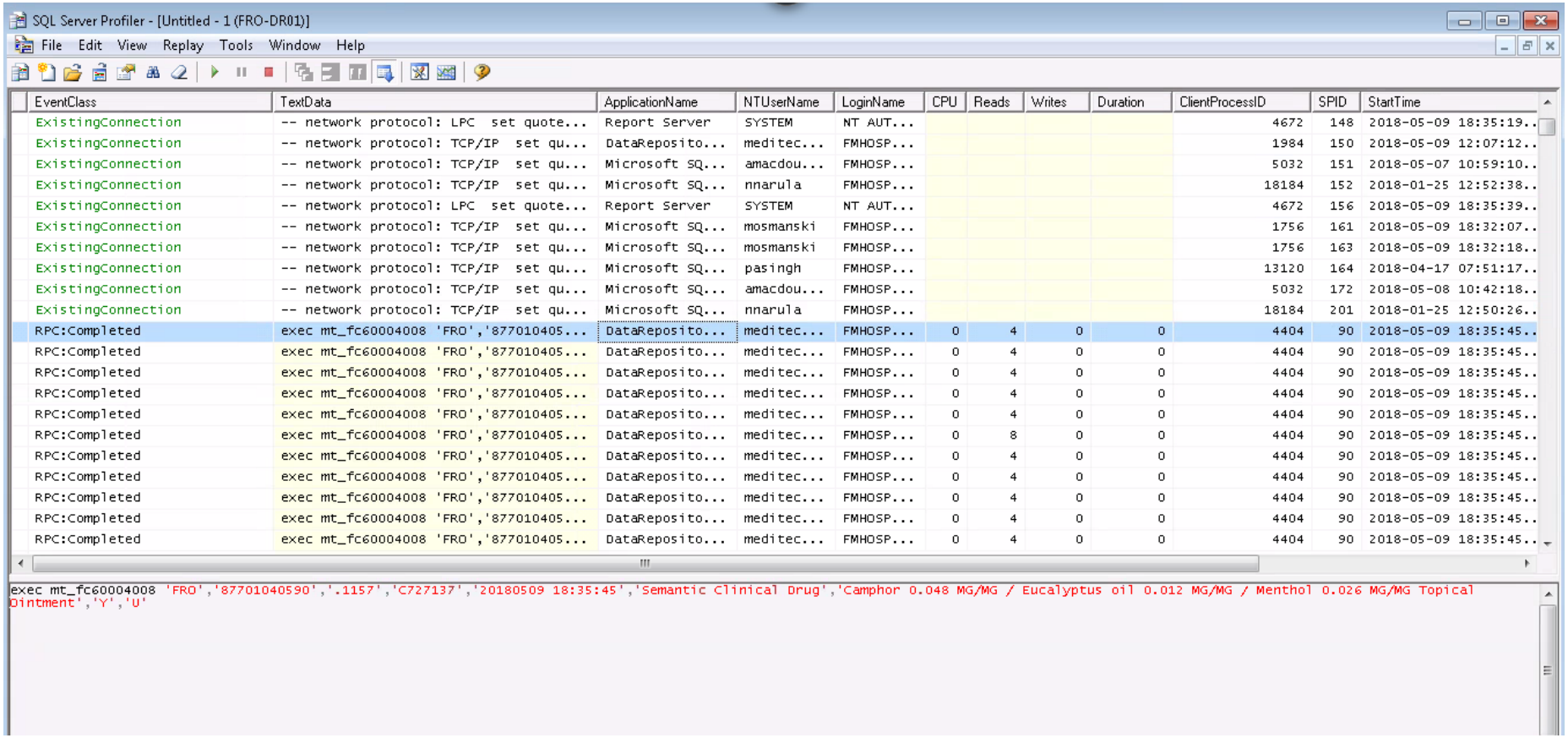
Activity Monitor



Copyright 2020 – Acmeware, Inc.

Other performance & monitoring tools

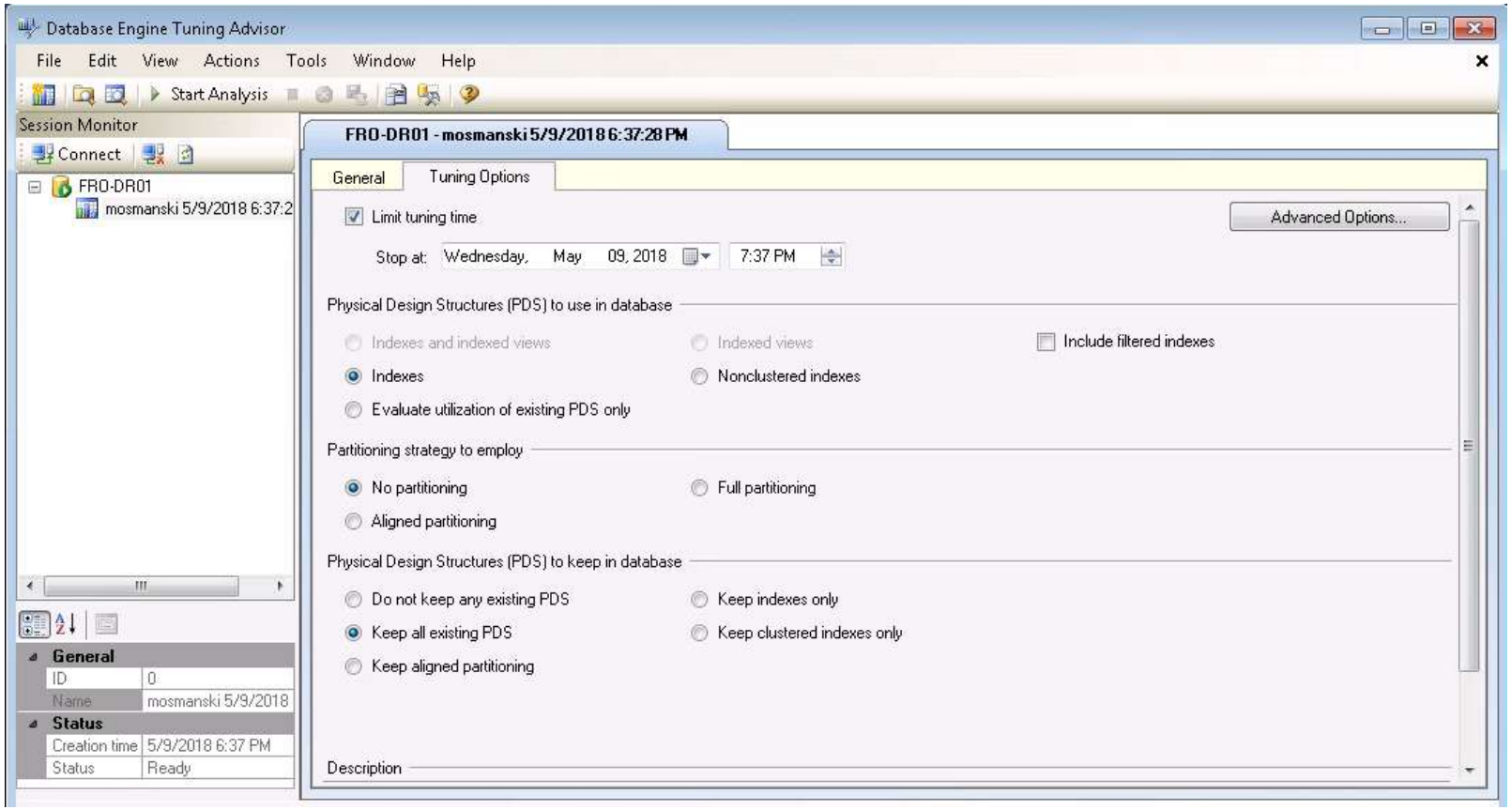
SQL Profiler



Copyright 2020 – Acmeware, Inc.

Other performance & monitoring tools

Database Engine Tuning Advisor



Copyright 2020 – Acmeware, Inc.

Discussion, Q&A

Thank you!

Acmeware

Copyright 2020 – Acmeware, Inc.