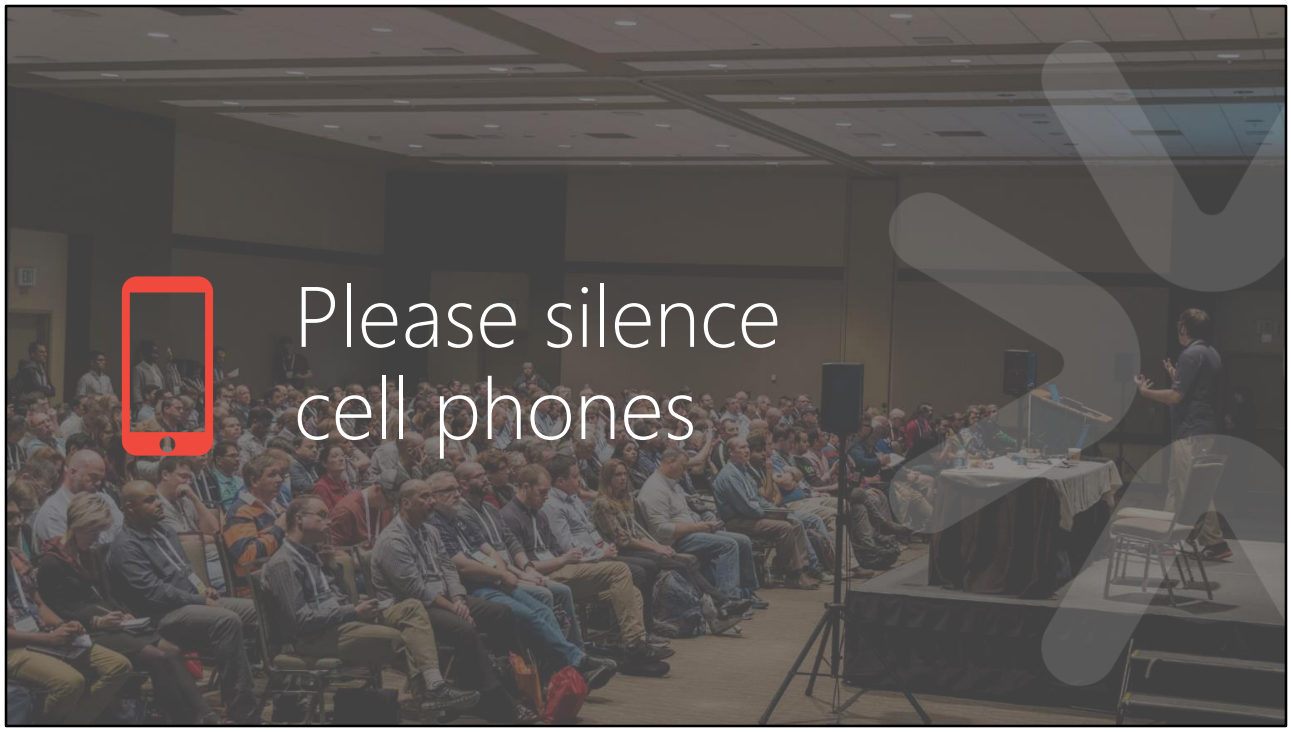
SQL Server Migrations Done the Right Way

Argenis Fernandez
Pam Lahoud
Pedro Lopes


Please silence
cell phones

Explore everything PASS has to offer

Free Online Resources

Newsletters

PASS.org

Free online webinar events

Local user groups around the world

Free 1-day local training events

Online special interest user groups

Business analytics training

Get involved

Session evaluations

Your feedback is important and valuable.

Submit by 5pm Friday, November 16th to win prizes.

3 Ways to Access:

Go to passSummit.com

Download the GuideBook App
and search: PASS Summit 2018

Follow the QR code link displayed on session signage throughout the conference venue and in the program guide

PASS

Pedro Lopes

Sr. Program Manager, Microsoft

/pedroazevedolopes

@SQLPedro

Role

Program Manager on SQL Server engineering team.

Focus Areas

Relational Engine: Query Processor, Programmability, Performance

Background

Working with SQL Server for 17y+.

Pam Lahoud

Sr. Program Manager, Microsoft

/pam_lahoud

@SQLGoddess

Over 20 years SQL Server experience

SQL Server 6.5 to now

Developer/DBA/Support

With Microsoft for 12 years

- Premier Field Engineer – 8 years
- Premier Developer Consultant – 4 years
- Program Manager (Storage Engine) – 1 year

Microsoft Certified Master

#MCM4LIFE

Argenis Fernandez

Principal Program Manager, Microsoft

/argenis

@DBArgenis

Focus on Hosting, OEM/Hardware vendor partnerships, and Persistent Memory

VMware vExpert 2016-2018

Microsoft Data Platform (SQL Server) MVP 2013-2016

Former Director, SQLSaturday, PASS

Microsoft Certified Master: SQL Server #MCM4LIFE

Founded the Security Virtual Chapter for PASS

Twitter enthusiast and occasional blogger

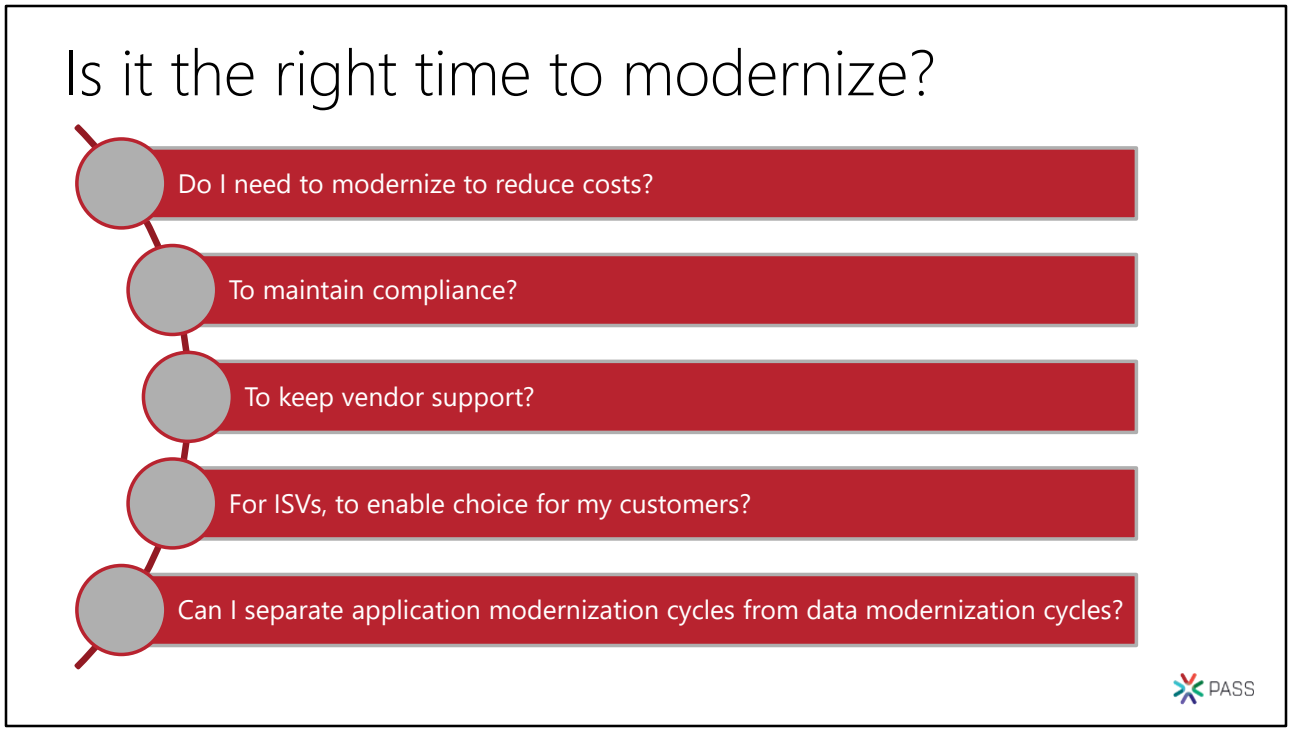
Agenda

Everything thing you need to know to successfully modernize your SQL Server environment on-prem with minimal risk.

- Why Upgrade?
- Database Compatibility Certification
- Upgrade Process
- 15 Minute Break (around 2:45PM)
- Post-migration
- Hardware Modernization

PASS

Pam does this one



As you may have observed, we released SQL 2016 about 24m after 2014, and 2017 about 16m after 2016. So there is a faster engineering cycle that's observable to meet market demands and pace of technology improvements. It really doesn't make sense that our customers would have to wait 2+ years to see innovations coming to their non-azure Data solutions, like it had happened historically.

Users and ISVs all face a fundamental and recurring question with their application and database estates: Is it the right time to modernize?
With that question come a number of considerations:

- Do I need to modernize to reduce costs? To maintain required industry certification? To keep vendor support? For ISVs, to enable choice for my customers?
- And how to minimize risk? Can I separate application modernization cycles from data modernization cycles?
- And what's my required certification process? What's the cost of re-certifying for Azure SQL, or newer SQL Server version?

Winter is coming…

PASS

July 9, 2019 – End of Support for SQL Server 2008 and 2008 R2
January 14, 2020 – End of Support for Windows Server 2008 and 2008 R2

https://www.microsoft.com/en-US/sql-server/sql-server-2008

Pedro starts here

Reference: https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-database-transact-sql-compatibility-level

Database Compatibility Level based certification

Stop certifying for any given platform (Cloud, on-prem)!

Stop certifying for a named SQL Server version!

Any certification process should be thought in terms of "*which target database compatibility level am I certifying to?*"

Updated public documentation: http://aka.ms/dbcompat

PASS

We believe this paradigm shift unlocks agile modernization cycles and reduces upgrade risks.
And we are backing it up with actual engine level protections, as well as updated public documentation that moves in this direction: see http://aka.ms/dbcompat.

Microsoft stands by DB Compat based certification

Microsoft Database Compatibility Level Protection

- Full Functional protection once assessment tools runs clean with no errors
- Query Plan shape protection on comparable hardware
- Maintaining backward compatibility is very important to SQL Server team

Key Benefits

Simplified application certification on-premise and Azure (e.g. Azure SQL DB MI)

Ability to provide customer a choice of latest SQL Server platform based on certified DB compat level

Improved risk management by decoupling application upgrade cycles from Database upgrade cycles

PASS

Reference: https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-database-transact-sql-compatibility-level

Database Compatibility Level behavior

Database Compatibility Level sets certain database behaviors to be compatible with the specified version of SQL Server.

- Compatibility level affects behaviors only for the specified database, not for the entire server.

Product	Compatibility Level Designation	Supported Compatibility Level Values
SQL Server 2019	150	150, 140, 130, 120, 110, 100
SQL Server 2017	140	140, 130, 120, 110, 100
Azure SQL Database	130	140, 130, 120, 110, 100
SQL Server 2016	130	130, 120, 110, 100
SQL Server 2014	120	120, 110, 100
SQL Server 2012	110	110, 100, 90
SQL Server 2008 R2	100	100, 90, 80
SQL Server 2008	100	100, 90, 80
SQL Server 2005	90	90, 80
SQL Server 2000	80	80

PASS

New Databases are set to compatibility level mapping to the version of the Database Engine, unless the **model** database has a lower compatibility level.

When a database is upgraded from any earlier version of SQL Server, the database **retains its existing compatibility level** if it is at least minimum allowed for that instance of SQL Server.

Upgrading a database with a compatibility level lower than the allowed level, sets the database to the lowest compatibility level allowed.

Azure SQL Database V12 was released in December 2014. One aspect of that release was that newly created databases had their compatibility level set to 120. In 2015 SQL Database began support for level 130, although the default remained 120.

Starting in **mid-June 2016**, in SQL Database, the default compatibility level are 130 instead of 120 for **newly created** databases. Existing databases created before mid-June 2016 are not affected, and maintain their current compatibility level (100, 110, or 120).

If you want level 130 for your database generally, but you have reason to prefer the level 110 **cardinality estimation** algorithm, see ALTER DATABASE SCOPED CONFIGURATION (Transact-SQL), and in particular its keyword

LEGACY_CARDINALITY_ESTIMATION = ON.
For details about how to assess the performance differences of your most important queries, between two compatibility levels on SQL Database, see Improved Query Performance with Compatibility Level 130 in Azure SQL Database.

Functional change protection

Breaking Changes = behavior changes resulting in different outcome.

- Not all protected under Database Compatibility.

Breaking Changes protected by Database Compatibility:

- The query below works until Database Compatibility 90, but errors out starting with Database Compatibility 100 (error 241, conversion fail):

```
SELECT DATEPART (year, '2007/05-30')
```

- Instead use:

```
SELECT DATEPART (year, '2007/05/30')
SELECT DATEPART (year, '2007-05-30')
```

PASS

Until DB Compat 90, datetime intrinsics such as DATEPART do not require string input values to be valid datetime literals. For example, SELECT DATEPART (year, '2007/05-30') compiles successfully.

Starting with DB Compat 100, datetime intrinsics such as DATEPART require string input values to be valid datetime literals. Error 241 is returned when an invalid datetime literal is used.

Functional change protection

Breaking Changes protected by Database Compatibility:

```
DECLARE @value datetime = '1900-01-01 00:00:00.003'
SELECT CAST(@value AS datetime2)
```

- In Database Compatibility 120 or lower, result is: `1900-01-01 00:00:00.0030000`.
- Under Database Compatibility 130, these show improved accuracy by accounting for the fractional milliseconds, resulting in: `1900-01-01 00:00:00.0033333`.

PASS

Implicit conversion from **datetime** to **datetime2** data types, and many others. Refer to https://support.microsoft.com/en-us/help/4010261/sql-server-and-azure-sql-database-improvements-in-handling-some-data-t

Always use explicit casting to datetime2 datatype whenever a mixed comparison scenario between datetime and datetime2 datatypes exists.

Functional change protection

Breaking Changes protected by Database Compatibility:

```
SET DATEFORMAT dmy;
DECLARE @t2 date = '12/5/2011' ;
SET LANGUAGE dutch;
SELECT CONVERT(varchar(11), @t2, 106);
```

- Up to Database Compatibility Level 110:

  `12 May 2011`

- Starting with Database Compatibility 120:

  `12 mei 2011`

PASS

In compatibility levels lower than 120, the language setting is ignored when converting a **date** value to a string value. Note that this behavior is specific only to the **date** type.

Functional change protection - caveats

Breaking Changes not protected by Database Compatibility:

- **Changed column names in system objects**. In SQL Server 2012 (11.x) the column *single_pages_kb* in *sys.dm_os_sys_info* was renamed to *pages_kb*.
  - Regardless of the compatibility level, the query below will produce error 207 (Invalid column name):

```
SELECT single_pages_kb FROM sys.dm_os_sys_info;
```

Functional change protection - caveats

Discontinued = removed from product.

- Discontinued functionality introduced in a given SQL Server version is not protected by compatibility level.

Removed T-SQL syntax.

- In SQL Server 2012 the *fastfirstrow* hint was removed.
- Regardless of the compatibility level, the query below will produce error 321 (not a recognized table hints option):

```
SELECT * FROM HumanResources.Employee WITH (FASTFIRSTROW);
```

- Instead use:

```
SELECT * FROM HumanResources.Employee OPTION (FAST = <n>);
```

PASS

Functional change protection - caveats

Discontinued = removed from product.

- Discontinued functionality introduced in a given SQL Server version is not protected by compatibility level.

Removed system objects.

- In SQL Server 2012 the *sp_dboption* was removed.
- Regardless of the compatibility level, the statement below will produce error 2812 (Could not find stored procedure 'sp_dboption'):

```
EXEC sp_dboption 'AdventureWorks2016', 'autoshrink', 'FALSE';
```

Plan affecting changes

Most noticeable changes between Database Compatibility that can affect performance are:

- Query Optimizer fixes under TF 4199.
- Changes to the Cardinality Estimator.

Plan affecting changes – TF 4199

QO changes that are made to previous releases of SQL Server are enabled by default under the latest Database Compatibility on a given product release, without trace flag 4199 enabled.

Database compatibility level	TF 4199	QO changes from previous database compatibility levels	QO changes for current version post-RTM
100 to 120	**Off**	Disabled	Disabled
	On	Enabled	Enabled

PASS

To accomplish this at the database level, see the QUERY_OPTIMIZER_HOTFIXES option in ALTER DATABASE SCOPED CONFIGURATION (Transact-SQL).

Starting with SQL Server 2016 (13.x) SP1, to accomplish this at the query level, add the USE HINT 'ENABLE_QUERY_OPTIMIZER_HOTFIXES' query hint instead of using this trace flag.

Plan affecting changes – TF 4199

QO changes that are made to previous releases of SQL Server are enabled by default under the latest Database Compatibility on a given product release, without trace flag 4199 enabled.

Database compatibility level	TF 4199	QO changes from previous database compatibility levels	QO changes for current version post-RTM
100 to 120	**Off**	Disabled	Disabled
	On	Enabled	Enabled
130	**Off**	Enabled	Disabled
	On	Enabled	Enabled

PASS

To accomplish this at the database level, see the QUERY_OPTIMIZER_HOTFIXES option in ALTER DATABASE SCOPED CONFIGURATION (Transact-SQL).

Starting with SQL Server 2016 (13.x) SP1, to accomplish this at the query level, add the USE HINT 'ENABLE_QUERY_OPTIMIZER_HOTFIXES' query hint instead of using this trace flag.

Plan affecting changes – TF 4199

QO changes that are made to previous releases of SQL Server are enabled by default under the latest Database Compatibility on a given product release, without trace flag 4199 enabled.

Database compatibility level	TF 4199	QO changes from previous database compatibility levels	QO changes for current version post-RTM
100 to 120	**Off**	Disabled	Disabled
	On	Enabled	Enabled
130	**Off**	Enabled	Disabled
	On	Enabled	Enabled
140	**Off**	Enabled	Disabled
	On	Enabled	Enabled

PASS

To accomplish this at the database level, see the QUERY_OPTIMIZER_HOTFIXES option in ALTER DATABASE SCOPED CONFIGURATION (Transact-SQL).

Starting with SQL Server 2016 (13.x) SP1, to accomplish this at the query level, add the USE HINT 'ENABLE_QUERY_OPTIMIZER_HOTFIXES' query hint instead of using this trace flag.

Plan affecting changes – CE

Timeout! CE?

- CE predicts how many rows your query will likely return and is used by the Query Optimizer to generate the optimal query plan.

Most systems benefit from the latest CE because it is the most accurate.

PASS

Why is the latest CE more accurate?

The CE predicts how many rows your query will likely return and is used by the Query Optimizer to generate the optimal query plan. With more accurate estimations, the Query Optimizer can usually do a better job of producing a more optimal query plan.

When migrating from an older versions of SQL Server to SQL Server 2014 or newer, **and** upgrading the Database Compatibility level 120 or above, a workload may be exposed to the risk of performance regression.

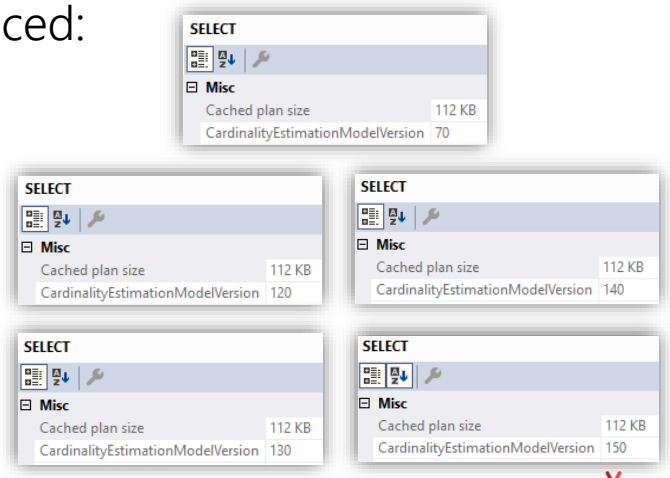
This is because starting with SQL Server 2014, all Query Optimizer changes are tied to the latest database compatibility level, so plans are not changed right at point of upgrade but rather when a user changes the COMPATIBILITY_LEVEL database option to the latest one. This capability, in combination with Query Store gives you a great level of control over the query performance in the upgrade process.

References:
https://docs.microsoft.com/en-us/sql/relational-databases/performance/cardinality-estimation-sql-server
https://docs.microsoft.com/en-us/sql/relational-databases/query-processing-architecture-guide

Plan affecting changes – CE

CE versions are tied to the Compatibility Level of the version it was first introduced:

Database compatibility level	CE Version
100 to 110	70
120	120
120	130
140	140
150	150



PASS

References:
https://docs.microsoft.com/sql/relational-databases/performance/cardinality-estimation-sql-server
https://docs.microsoft.com/sql/relational-databases/query-processing-architecture-guide

Feature Roadmaps

For new development work, or when an existing application requires use of new features, as well as performance improvements done in the query optimizer space, plan to certify on the latest database compatibility level…

In case you missed it, check out the following session for more information on the SQL Server roadmap:

- The Roadmap for SQL Server (November 7 10:45AM – 12:00 PM)

PASS

At the end of the slide, call for the break (UNLESS TIMING DOESN'T WORK)

Argenis starts here

Upgrade Strategies

Side-by-side	In-place
• Allows for upgrade of OS • Easier testing • Easier rollback strategy • Less downtime	• Doesn't require additional hardware • No data migration required

PASS

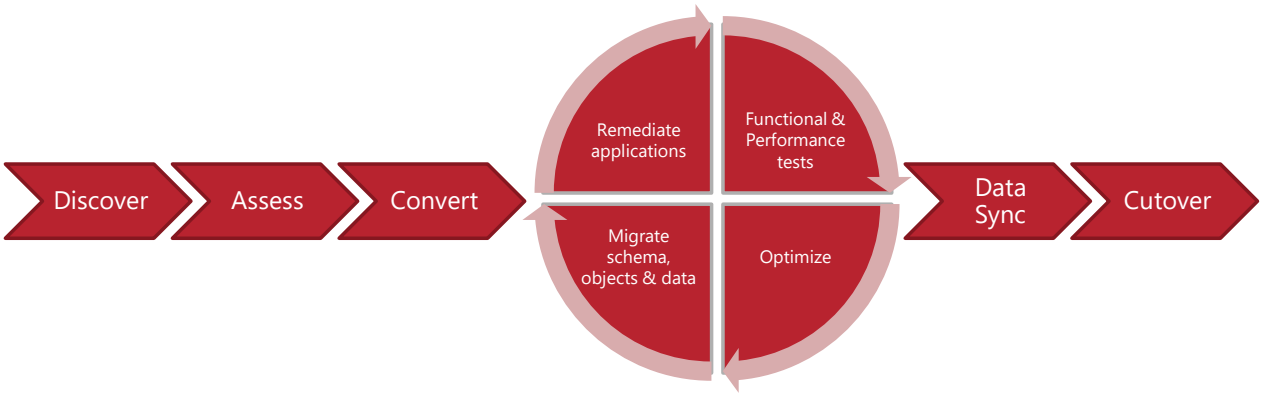
Encourage customers to perform side-by-side upgrades in order to encourage thorough testing before cutover. Also generally allows for less downtime, easier rollback and overall less risk.

Traditional Upgrade Strategies

Feature	Notes
Log Shipping	Cutover measured in (typically) minutes
Replication	Cutover measured in (potentially) seconds
Backup and Restore	This is going to take a while!
Filesystem/SAN Copy	Ditto - the latter being significantly faster
Availability Groups (NOT available in < 2012)	Cutover measured in (typically) seconds

PASS

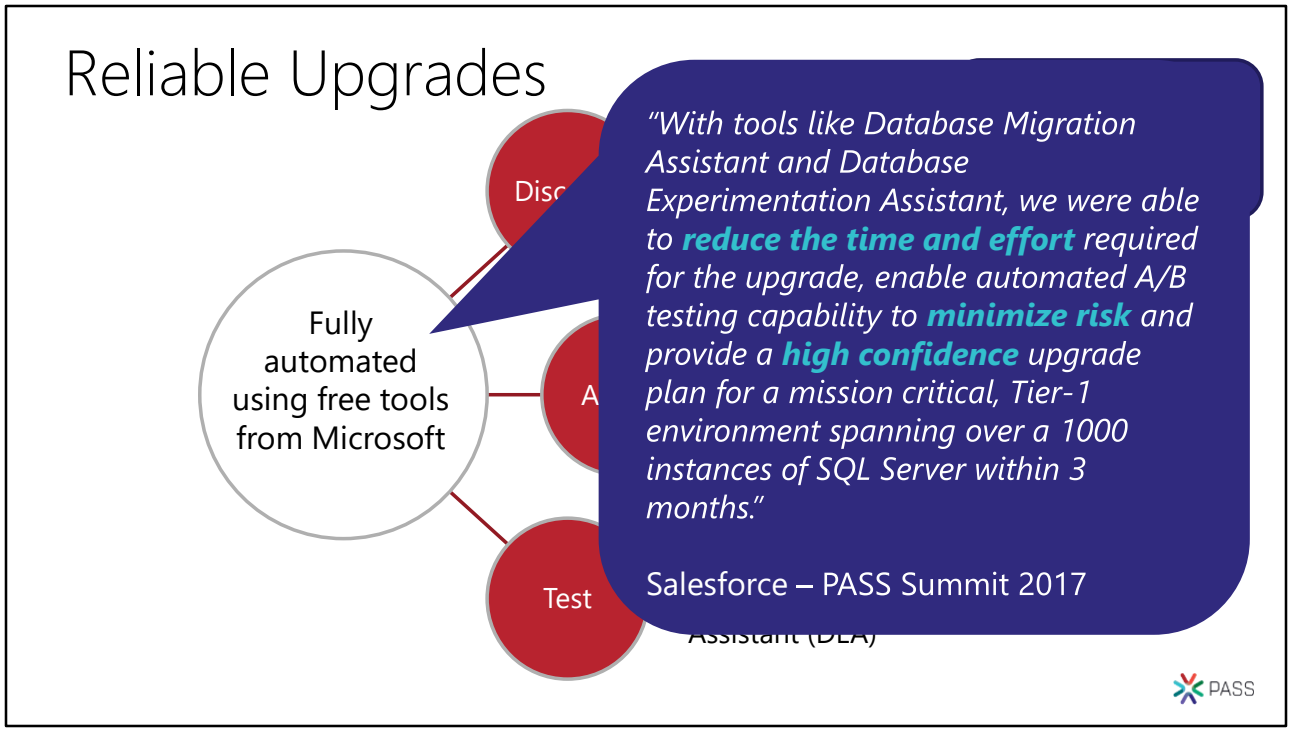
Minimize Risk with the Database Migration Guide



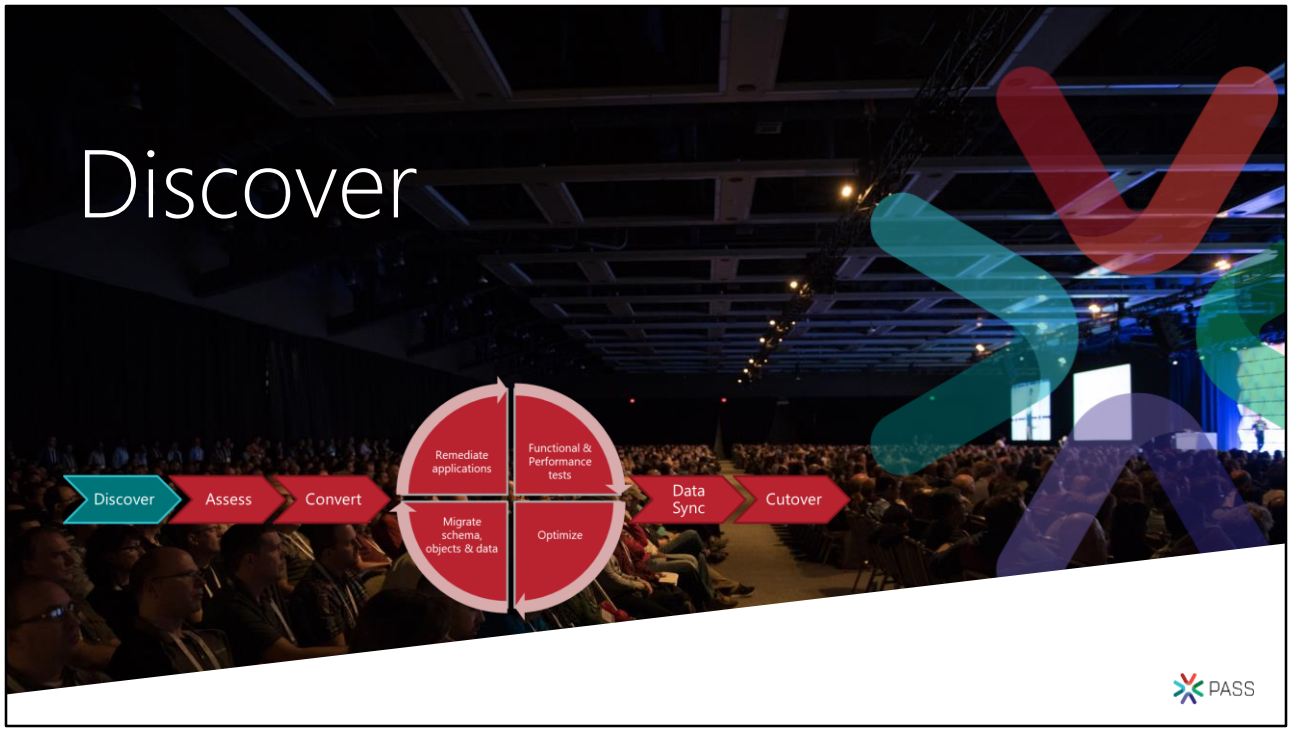
Pam does this one

Migrations should be approached with the same rigor and processes as a full software or hardware project – a solid methodology is required for success. Microsoft provides you with all the tools you need to achieve a seamless, reliable upgrade experience.

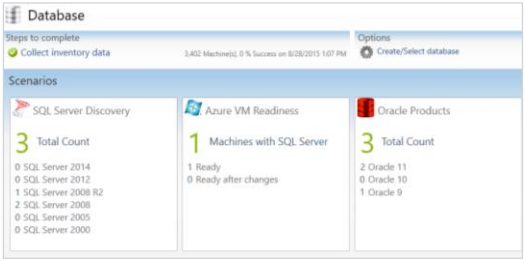
The Database Migration Guide provides step-by-step guidance on reliable upgrade methodology.
https://datamigration.microsoft.com/scenario/sql-to-sqlserver



Link in slide: https://datamigration.microsoft.com/scenario/sql-to-sqlserver



Discover with MAP Toolkit



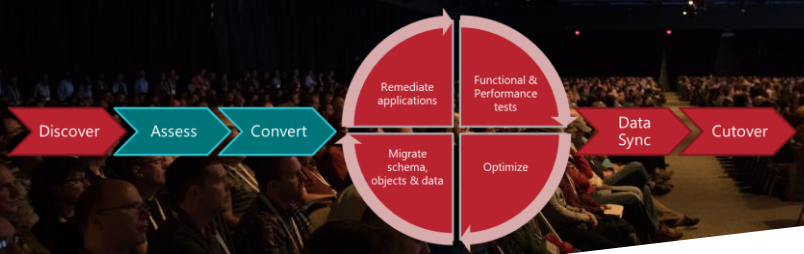
- Which SQL Server versions do I have?
- Which Editions am I running?
- Which SQL Server components are installed?
- How many cores are on each server?
- How many databases are in each instance?
- What are the sizes of all my databases?
- What are the settings for each instance and database?

The Microsoft Assessment and Planning Toolkit (MAP) is an agentless, automated, multi-product planning and assessment tool for quicker and easier desktop, server and cloud migrations. MAP can be used to provide an inventory of all your database servers, including installed components, database details, settings and capacity planning information. This Solution Accelerator provides a powerful inventory, assessment, and reporting tool to simplify the migration planning process.

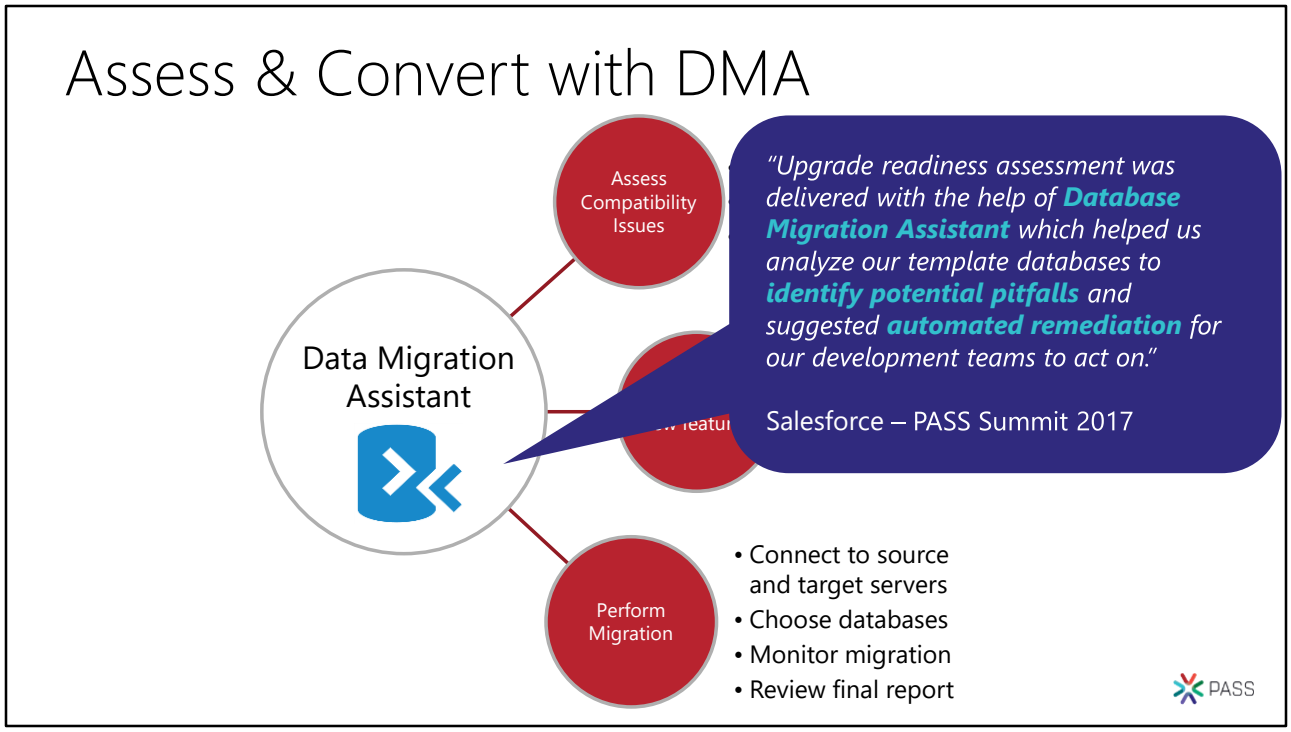
There are 8 areas that the Microsoft Assessment and Planning Toolkit (MAP) toolkit assesses:

- Cloud
- Desktop
- Server
- Desktop Virtualization
- Server Virtualization
- Database
- Usage Tracking
- Environment

Assess & Convert

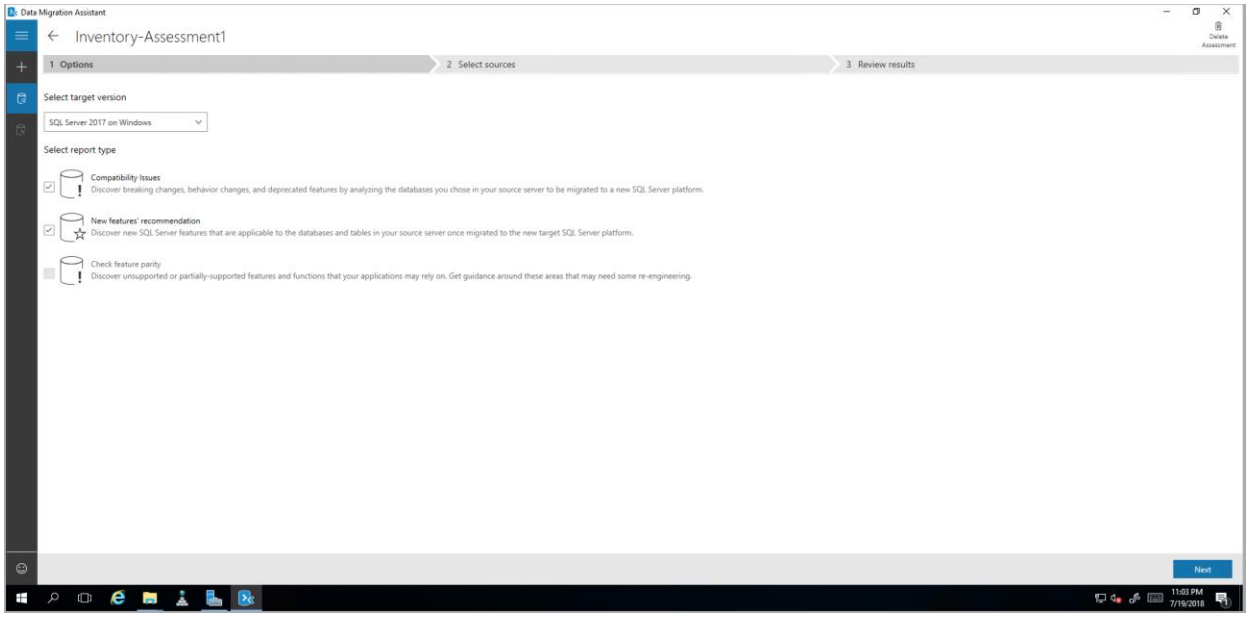


PASS

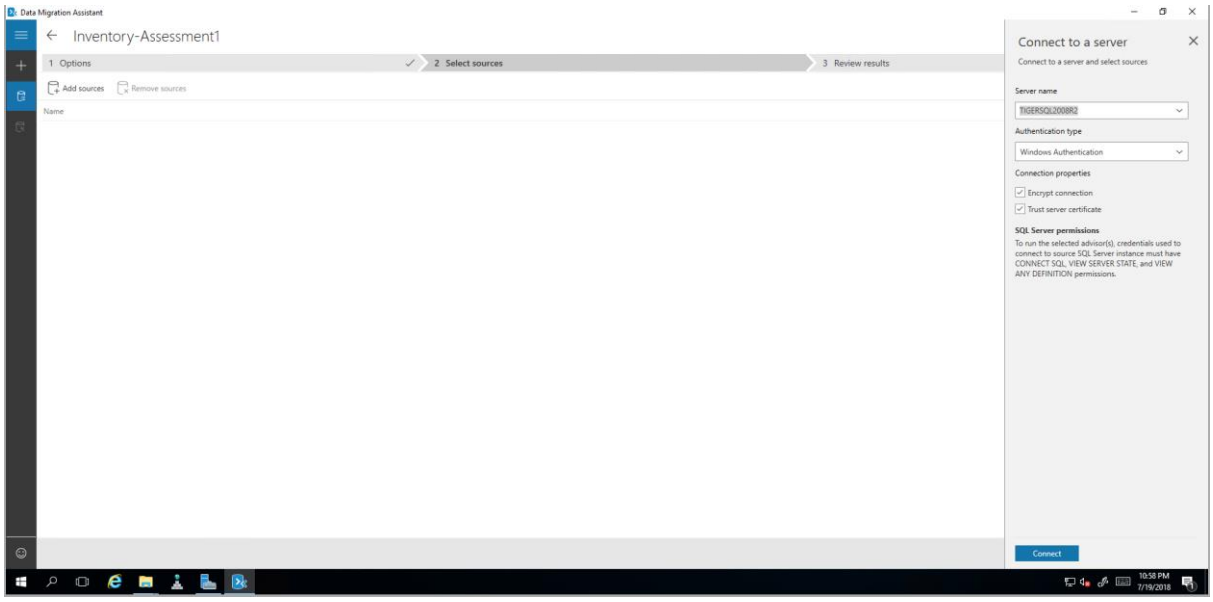


Data Migration Assistant (DMA) detects compatibility issues that can impact database functionality on your new version of SQL Server. It recommends performance and reliability improvements for your target environment. It allows you to not only move your schema and data, but also uncontained objects from your source server to your target server. Once the assessment is complete and issues have been mitigated, the DMA will automate the process of migrating your schema and data to the target instance.


Database
Migration
Assistant
Demo
PASS

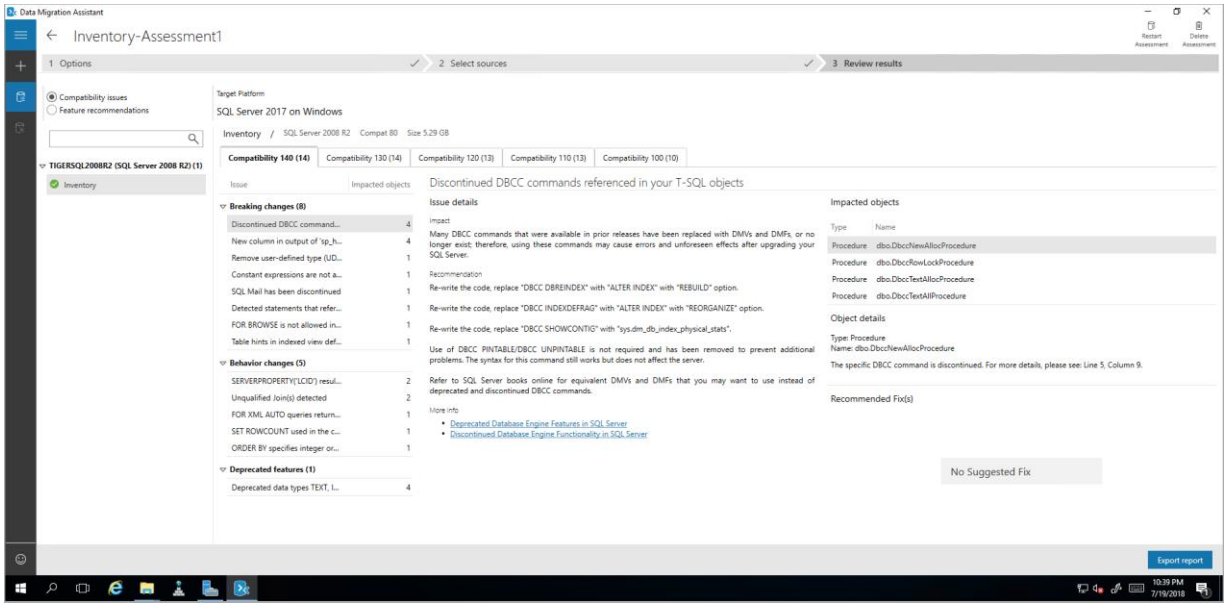


DMA Assessment: Options



PASS

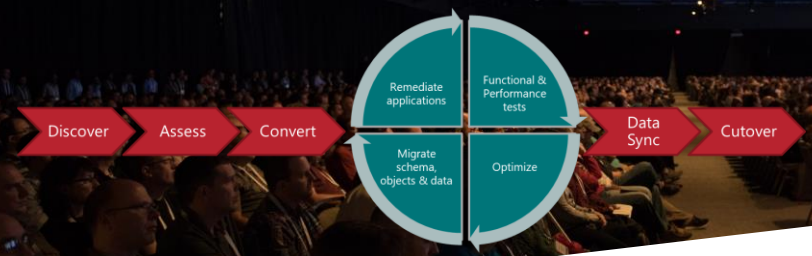
DMA Assessment: Select Sources



PASS

DMA Assessment: Completed

Test & Optimize



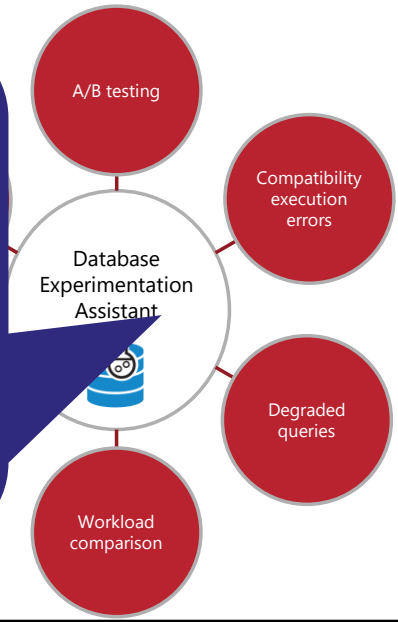
PASS

Test & Optimize with DEA

*"**Database Experimentation Assistant** helped us perform comparison tests between our current environment and a test SQL Server 2016 environment to identify **regressions**, **breaking changes** and **performance characteristics**.*

*Because of the **automation**, we were able to repeat such an exercise with multiple iterations using **different configuration settings**."*

Salesforce – PASS Summit 2017

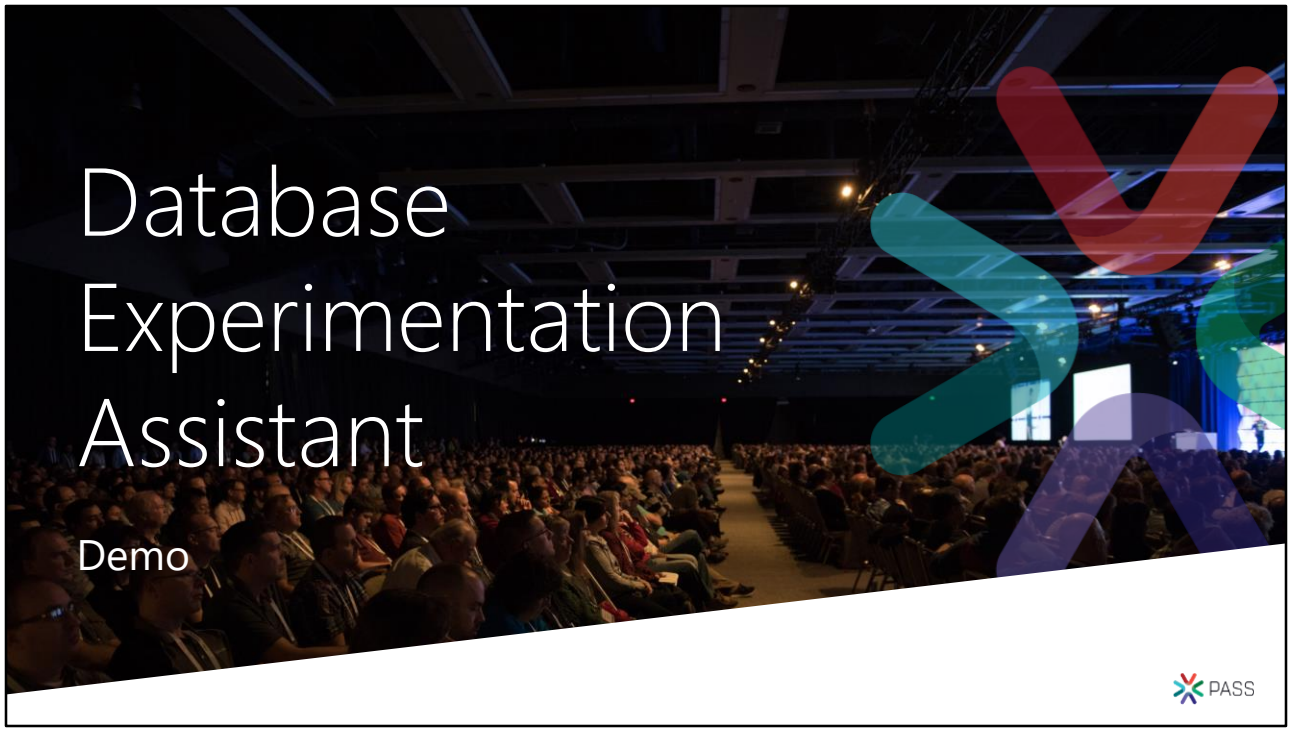


PASS

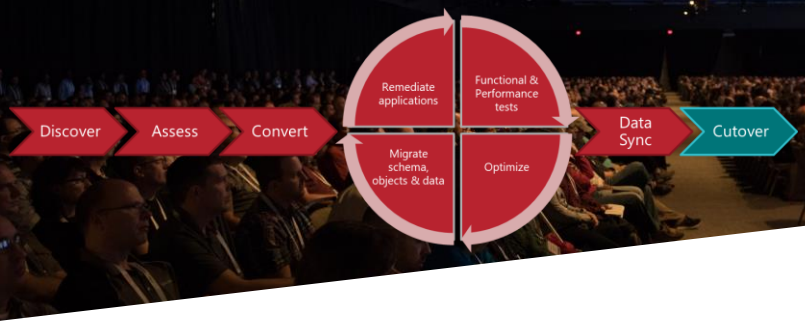
[Database Experimentation Assistant (DEA)] is an A/B testing solution for SQL Server upgrades. It will assist in evaluating a targeted version of SQL for a given workload. Customers who are upgrading from previous SQL Server versions (SQL Server 2005 and above) to any new version of the SQL Server will be able to use these analysis metrics.

DEA allows:

- Automated script to set up **workload capture and replay** of production database (using existing SQL server functionality Distributed Replay & SQL tracing).
- Perform **statistical analysis** on traces collected using both old and new instances.
- Visualize data through detailed **reports.**


Database
Experimentation
Assistant
Demo
PASS

Post-Migration



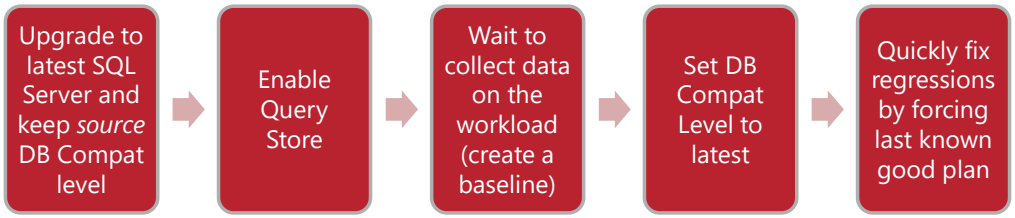
PASS

I moved the data, am I done?

SQL Server post migration step is crucial for reconciling any data accuracy and completeness.

But also to uncover performance issues with the workload.

Recommended DB Compatibility Level upgrade process:

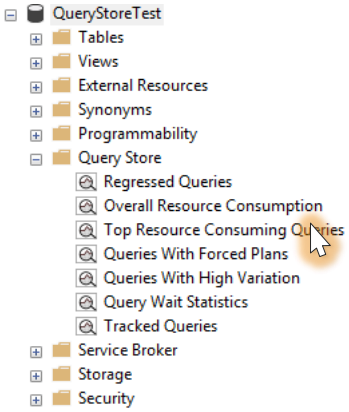


PASS

Reference:
https://docs.microsoft.com/sql/relational-databases/performance/query-store-usage-scenarios#CEUpgrade

Query Store

Comprehensive query-performance information when you need it most!

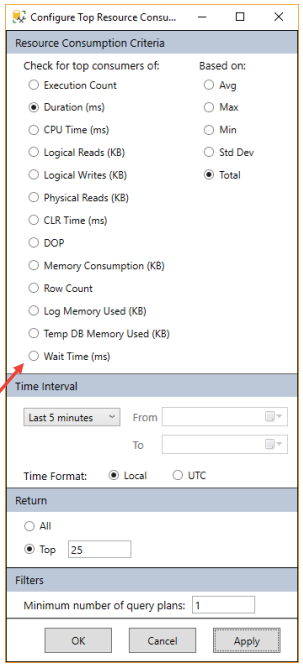


PASS

Find issues with 2 clicks.

Query Store

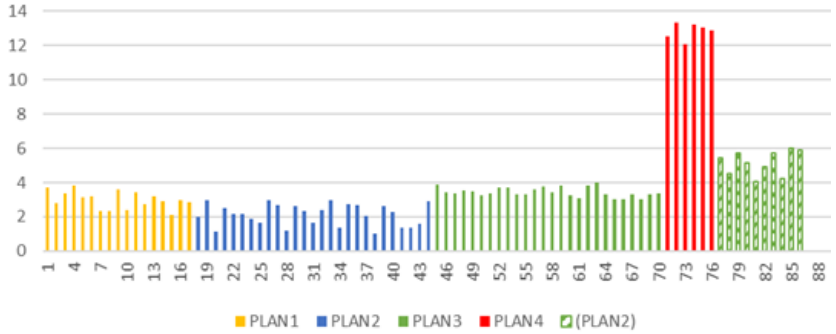

Top 25 resource consumers for database QueryStoreTest. Time period: Last 5 minutes ending at 10/29/2018 8:36 AM
Metric Duration (ms)
Statistic Total
execution count
query id
Plan summary for query 1
Total
Plan Id
8:33 AM 8:34 AM 8:35 AM 8:36 AM 8:37 AM 8:38 AM 8:39 AM
Plan 1 [not forced]
Force Plan
Unforce Plan
Query 1: Query cost (relative to the batch): 100%
select top 10000 * from t1 where c1=@p1 and c2=@p2
Missing Index (Impact 99.711): CREATE NONCLUSTERED INDEX [<Name of Missing Index, sysname,>] ON [dbo].…
SELECT
Cost: 0 %
Top
Cost: 1 %
Table Scan
[t1]
Cost: 99 %
Includes waits in SQL Server 2017


Configure Top Resource Consu...
Resource Consumption Criteria
Check for top consumers of:
Based on:
Execution Count
Duration (ms)
CPU Time (ms)
Logical Reads (KB)
Logical Writes (KB)
Physical Reads (KB)
CLR Time (ms)
DOP
Memory Consumption (KB)
Row Count
Log Memory Used (KB)
Temp DB Memory Used (KB)
Wait Time (ms)
Avg
Max
Min
Std Dev
Total
Time Interval
Last 5 minutes
From
To
Time Format:
Local
UTC
Return
All
Top
25
Filters
Minimum number of query plans:
1
OK
Cancel
Apply

Query Store and Automatic Plan Correction

Identifies the problematic query plan and “fixes” it to be optimal

In the scope of a DB Compatibility upgrade, only works if the recommended process was followed!



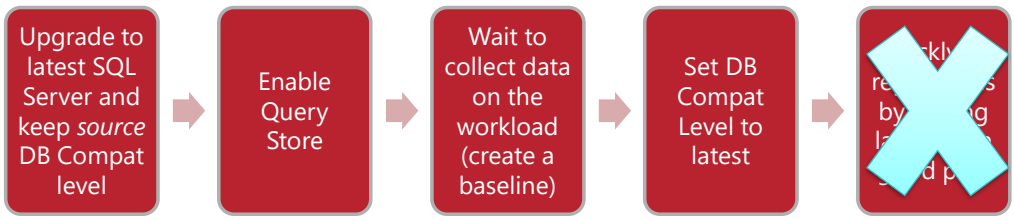
PASS


Automatic Tuning
Demo
PASS

Database Upgrade with Query Tuning Assistant

It's crucial to uncover query performance issues with the workload, now that it's running on a newer version.

Priority 1: follow documented DB Compatibility upgrade procedure, and now you can be guided through that.



PASS

Post-migration/upgrade

Crucial to uncover query performance issues with the workload, as it runs on the newer version of SQL Server Database Engine.

User needs to follow documented DB upgrade procedure (https://docs.microsoft.com/en-us/sql/relational-databases/performance/query-store-usage-scenarios#CEUpgrade).

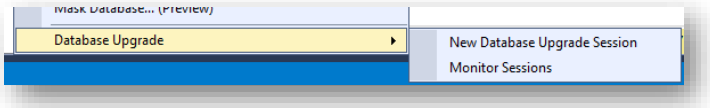
User database is still in before-upgrade database compatibility level, and QTA will assist in collecting baseline workload data (if none available), bump database compatibility level, collect 2nd pass of workload data, and work on any regressions found based on "Regressed Queries" QS report.

Database Upgrade with Query Tuning Assistant

Targets known possible patterns of query regressions due to change in CE version:

- Independence vs Correlation
- Simple Containment vs Base Containment
- TVF fixed estimation of 100 rows vs 1 row

Workflow requires user interaction at well-defined stages, via GUI or PS.

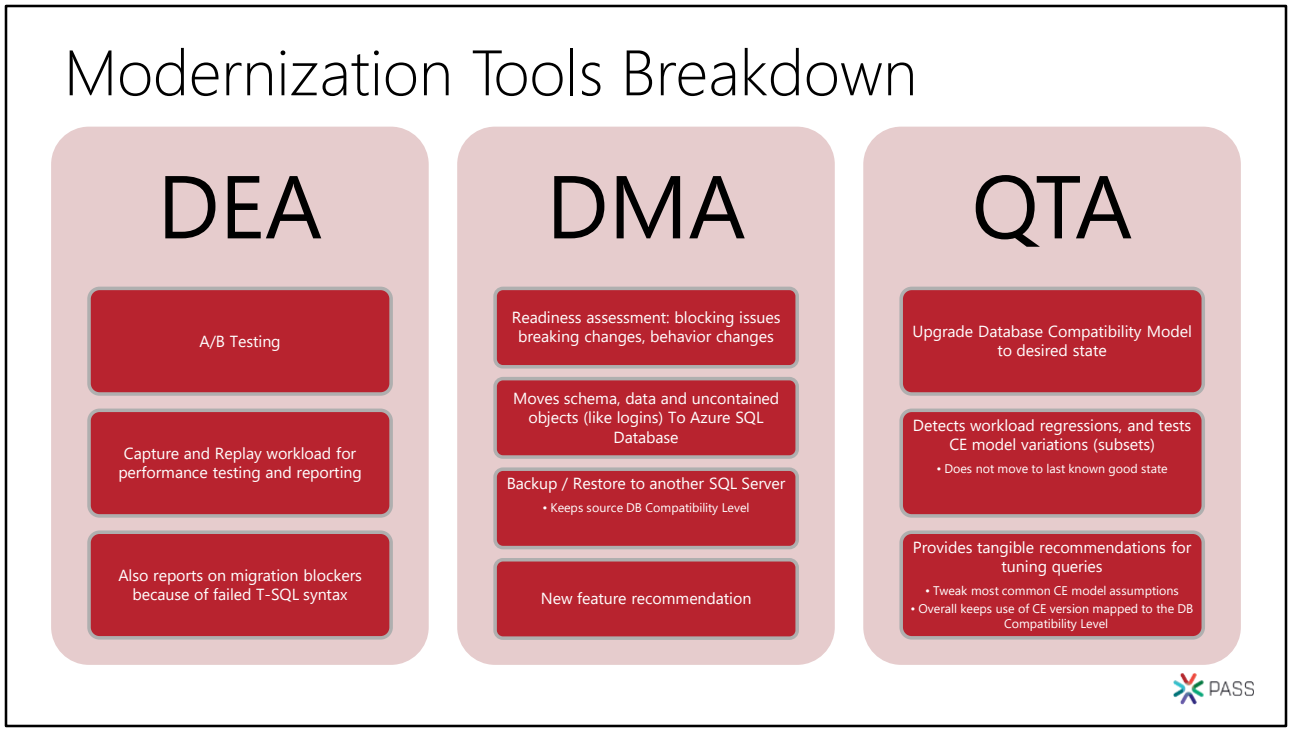


Available as:

- SSMS-based wizard-like experience
- Powershell for use at scale

PASS


Upgrading a database with QTA
Demo
PASS



DMA
https://docs.microsoft.com/en-us/sql/dma/dma-assesssqlonprem
https://docs.microsoft.com/en-us/sql/dma/dma-overview

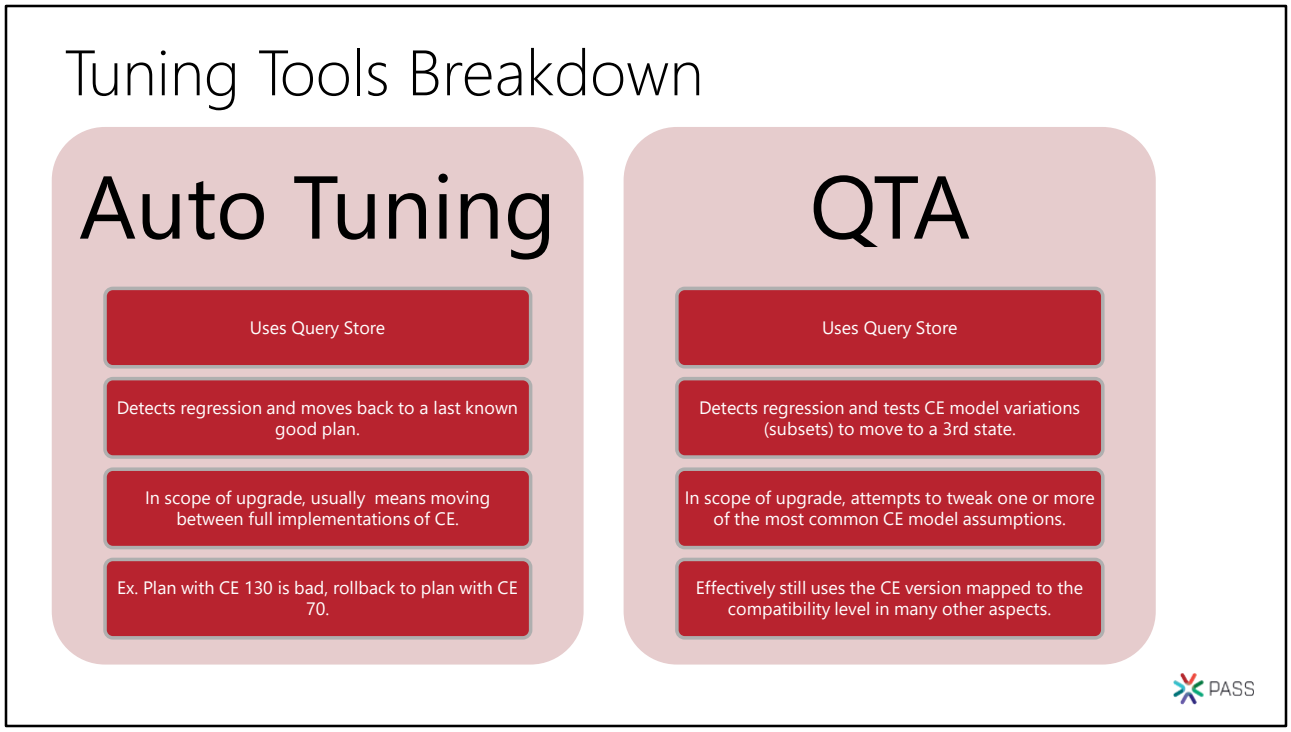
New feature recommendations include In-Memory OLTP and Columnstore, Stretch Database, Always Encrypted, Dynamic Data Masking, and Transparent Data Encryption.

DEA
Database Experimentation Assistant (DEA) is an A/B testing solution for changes in SQL Server environments (e.g. upgrade, new indexes, etc.). It assists in evaluating how the workload on your source server (current environment) will perform against your new environment. It guides you through performing an A/B test through three steps: capture, replay, and analysis.

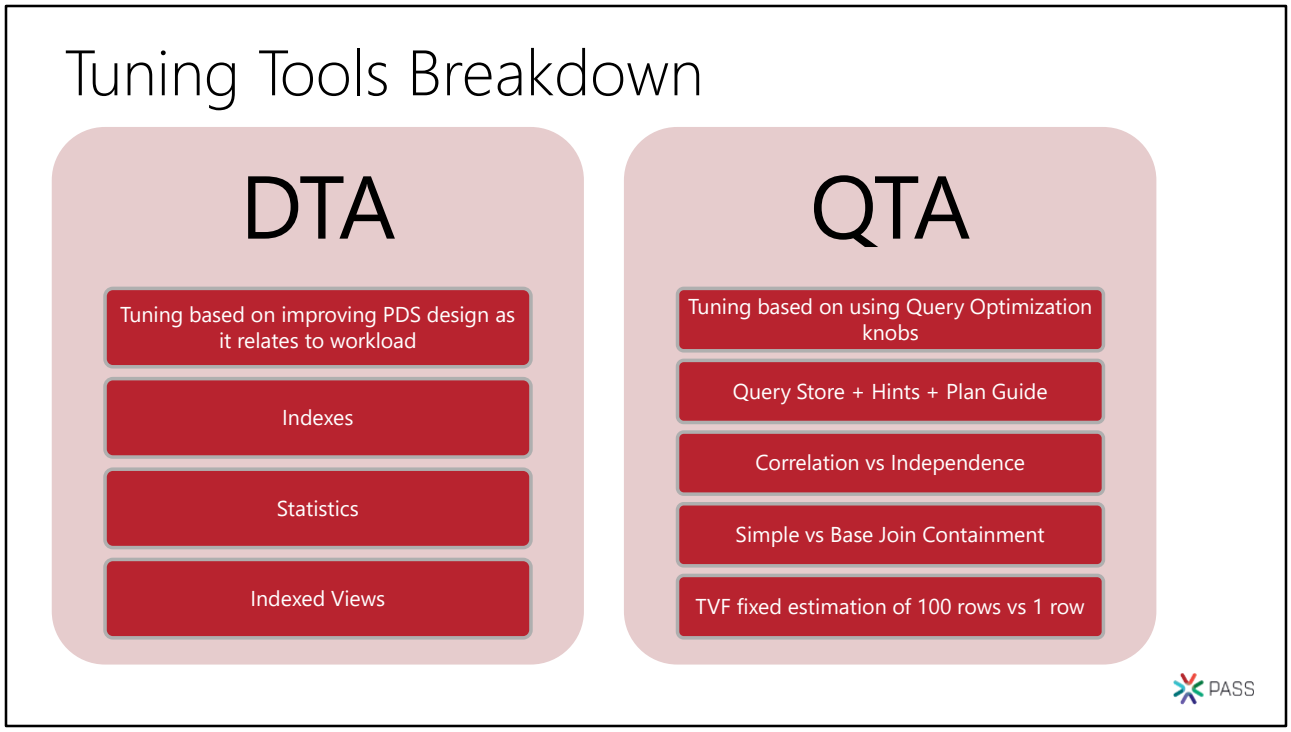
QTA
Query Tuning Assistant (QTA) helps address some of the most common causes of CE-related performance regression, namely the following model assumption changes, starting with SQL Server 2014:

- Independence vs Correlation
- Simple Containment vs Base Containment
- TVF fixed estimation of 100 rows vs 1 row

Note: as a last resort, full-fledge use of CE 70 is also considered, when all else doesn't yield desired results.
This is done by attempting to use targeted USE HINT query hints that change these assumptions, for regressed SELECT-based queries.



How is QTA different from Auto-Tuning's Automatic Plan Correction (APRC)?
APRC works by detecting the regression and freezing the last known good plan, which in the scope of an upgrade with subsequent CE regressions, can mean rolling back to CE 70.
QTA works by moving to a 3rd state, where it attempts to tweak one or more of the most common CE model assumptions (Correlation vs Independence; Simple containment vs Base containment), but effectively still using the CE version mapped to the compatibility level in many other aspects.



Here's some info about QTA:
The 1st release targets known possible patterns of query regressions due to change in CE version.
Workflow requires user interaction at well-defined stages, via GUI or PS.
Relies on Query Store as the source of truth, from where we get the regressed queries – this assumes that Query Store has been populated with a baseline – typically test or production workload running in the source/lower compatibility level. From there, compatibility level is bumped at some point, and as the workload executes, some queries start to regress (plan shape is different due to change in CE model version). QTA takes these queries* and experiments a few knobs that aim to improve its performance (see USE HINT docs). Then allows the user to create plan guides for queries that showed improvements.
* scoped to SELECT statements only, and parameterized queries where parameter is known. Queries that depend on runtime constructs such as temp tables or table variables are scoped out for now.
The UI experience is tailored around a post-upgrade workflow that follows the documented DB upgrade procedure.


Hardware Modernization
PASS

The Benefits of Modernizing

Increased scalability

Enable your developers to leverage new and useful features to make their code and time-to-market more efficient

Avoid an expensive CSA (Custom Support Agreement)

Increase your business' agility

Stay compliant by applying security patches for known vulnerabilities

PASS

Faster CPUs

- You're paying per core, no matter how fast they are
- The faster they get, the further your investment goes
- No one complains until it's too late
- Avoid you into instance sprawl -> Increased licensing cost

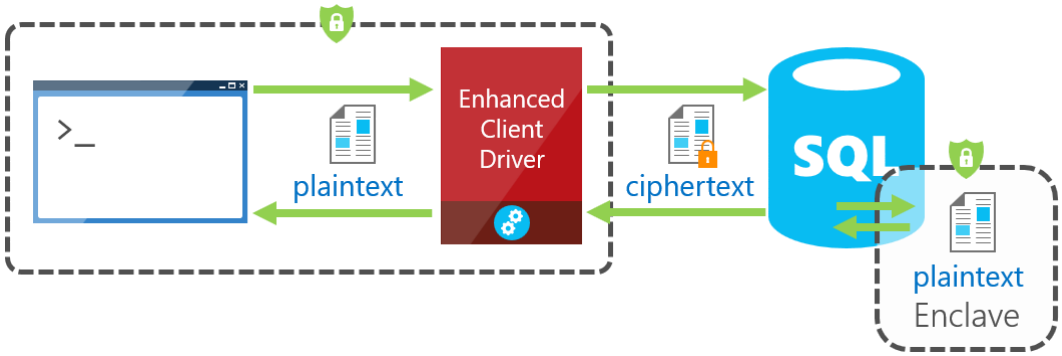
- The (tangible + intangible) cost to your business of not modernizing is **much** higher than the cost of removing blockers for adoption!

Secure Enclave: The Basics

```
0:061> du 0x000001cc0e5a24f0
000001cc`0e5a24f0  "????????????????????????????????"
000001cc`0e5a2530  "????????????????????????????????"
000001cc`0e5a2570  "????????????????????????????????"
000001cc`0e5a25b0  "????????????????????????????????"
000001cc`0e5a25f0  "????????????????????????????????"
000001cc`0e5a2630  "????????????????????????????????"
000001cc`0e5a2670  "????????????????????????????????"
000001cc`0e5a26b0  "????????????????????????????????"
000001cc`0e5a26f0  "????????????????????????????????"
000001cc`0e5a2730  "????????????????????????????????"
000001cc`0e5a2770  "????????????????????????????????"
000001cc`0e5a27b0  "????????????????????????????????"
```

PASS

Secure Enclaves with SQL Server 2019



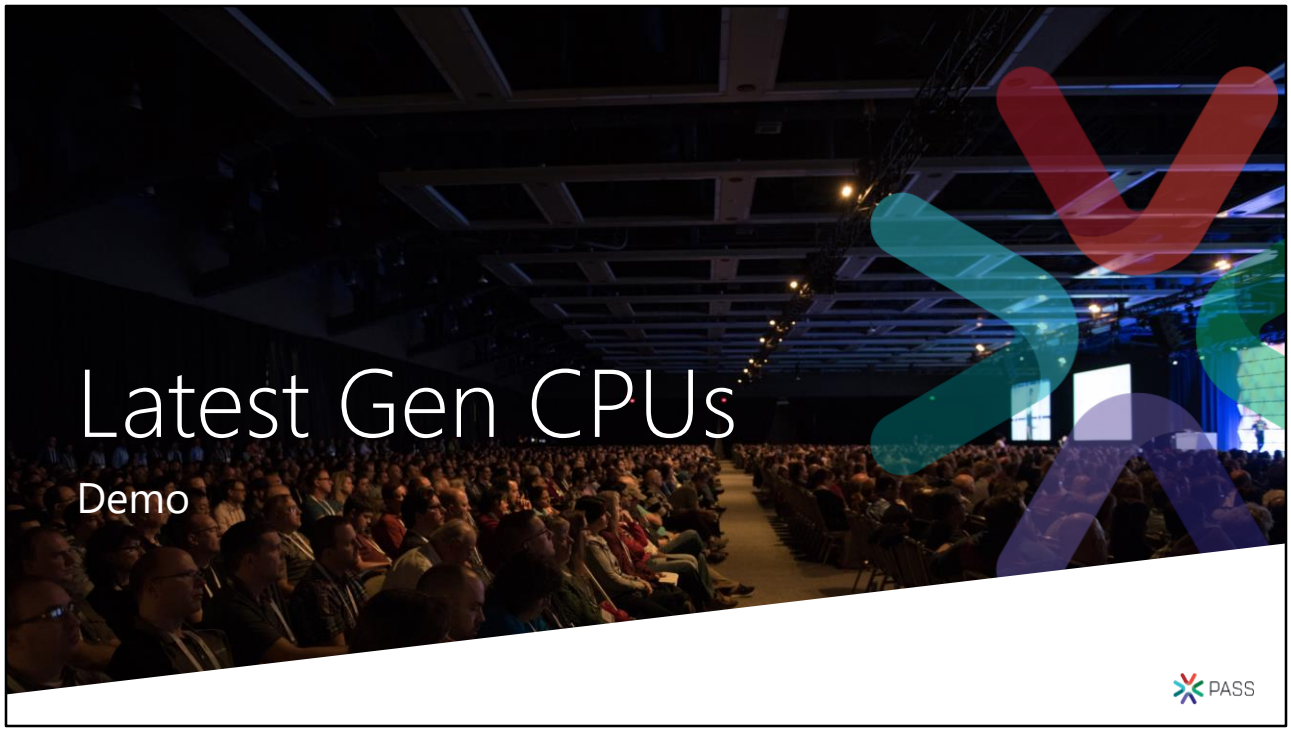
CPUs: Don't Leave Money On The Table!

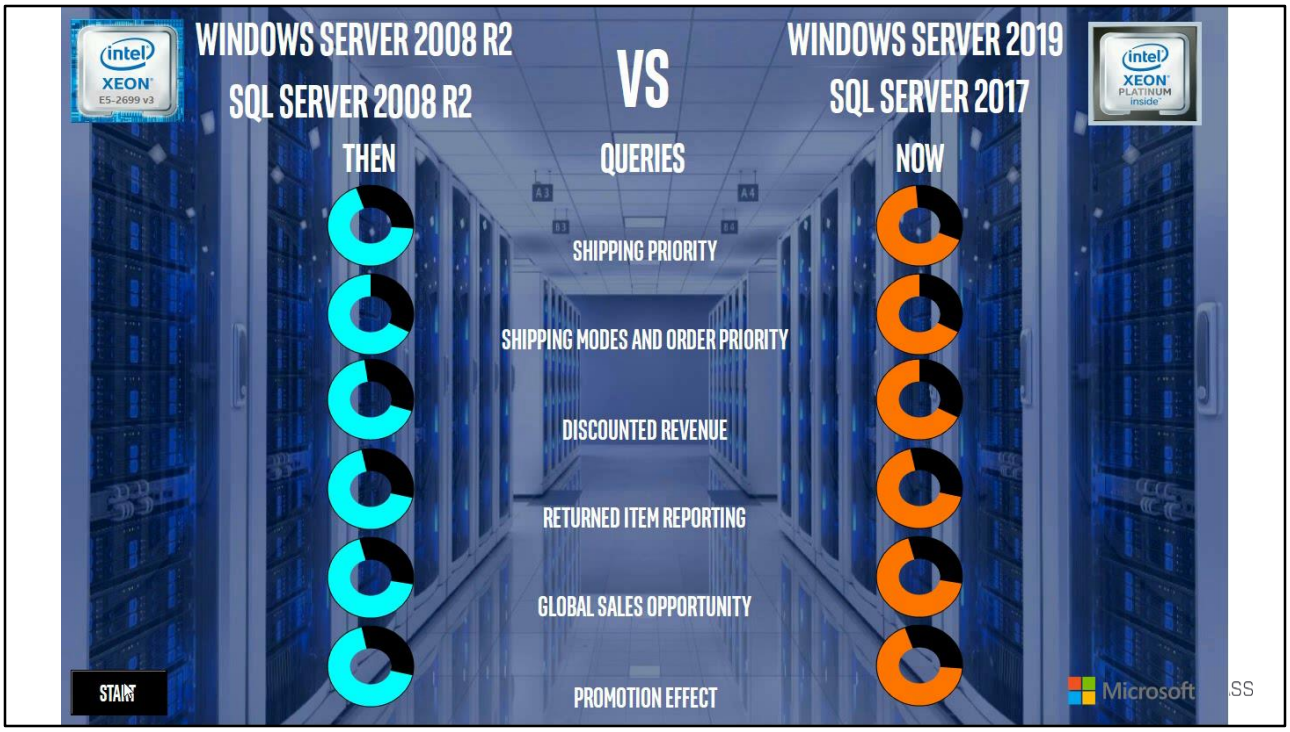
Faster CPUs means bigger bang for your buck on per-core license

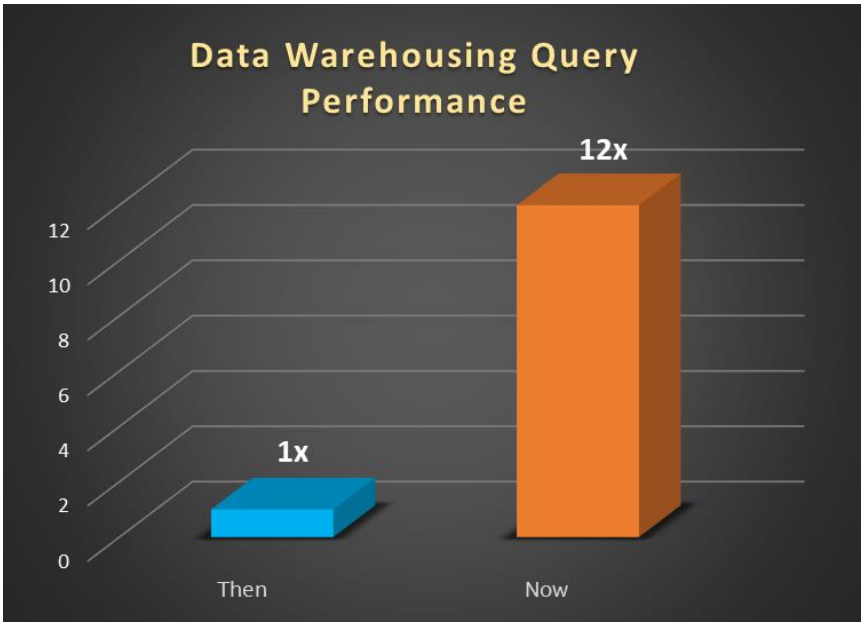
Combine with the latest on Windows Server + SQL Server

Run the same workload at a much lower cost

- This allows you to consolidate, or
- Sets you up for future growth


Latest Gen CPUs
Demo
PASS


WINDOWS SERVER 2008 R2
SQL SERVER 2008 R2
THEN
VS
QUERIES
WINDOWS SERVER 2019
SQL SERVER 2017
NOW
SHIPPING PRIORITY
SHIPPING MODES AND ORDER PRIORITY
DISCOUNTED REVENUE
RETURNED ITEM REPORTING
GLOBAL SALES OPPORTUNITY
PROMOTION EFFECT
START
Microsoft
SS


Data Warehousing Query Performance
12x
1x
12
10
8
6
4
2
0
Then
Now

Pick The Right CPU for your SQL Server workload!

Model	TPC-E Score	Total Cores	Score/Core	License Cost (YMMV)
(2) Xeon E5-2609 v4	693.79	16	43.36	$114,048.00
(2) Xeon E5-2637 v4	1,428.79	8	178.54	$57,024.00

Special thanks to Glenn Berry from SQLSkills

PASS

How Does Storage Help?

Core Storage Concepts

IOPS – I/Os per second (But think currency!)

- Disk Reads/sec
- Disk Writes/sec
- Disk Transfers/sec

Throughput – Overall bandwidth to the storage device

- Disk Read Bytes/sec
- Disk Write Bytes/sec
- Disk Bytes/sec

Latency – Time to complete a single I/O (Averaged)

- Avg. Disk sec/Read
- Avg. Disk sec/Write
- Avg. Disk sec/Transfer

Tying Storage Concepts to Database Performance

Capacity planning means more than just disk size

Know your workload

- OLTP = Focus on latency
- OLAP = Focus on throughput
- Backups = Focus on capacity BUT also time to complete

Don't know your workload? Replay!

PASS

A 1TB disk may mean very different things

If your workload is OLTP/Transactional in nature, then you want to focus on lower latency

If your workload if OLAP/Data Warehouse/Analytics, then you want to focus on throughput/bandwidth

If you workload is mixed, make sure you replay the actual workload and not just use a synthetic test.

Latest Generation DAS (Direct Attached Storage)

- PCIe based solutions
- Extreme Low latency (Microseconds range)
- High Throughput
- Typically no HW RAID support, but you can have multiple devices on a single system
- NAND Flash based or Intel 3D XPoint based
- Good for Availability Group Topologies

Latest Generation Shared Storage

Typically NAND flash based, some leverage Intel's 3D Xpoint

Very good latency (avg < 1 millisecond)

Excellent throughput (Typically 3GB/sec and > 10GB/sec in some cases)

Block services built-in: snapshots, replication

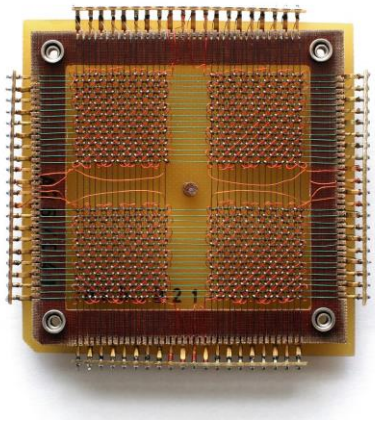
Offerings from DellEMC, Hitachi Vantara, HPE Nimble Storage, IBM, Kaminario, NetApp, Pure Storage and others

Excellent for performing tests and validation of workloads

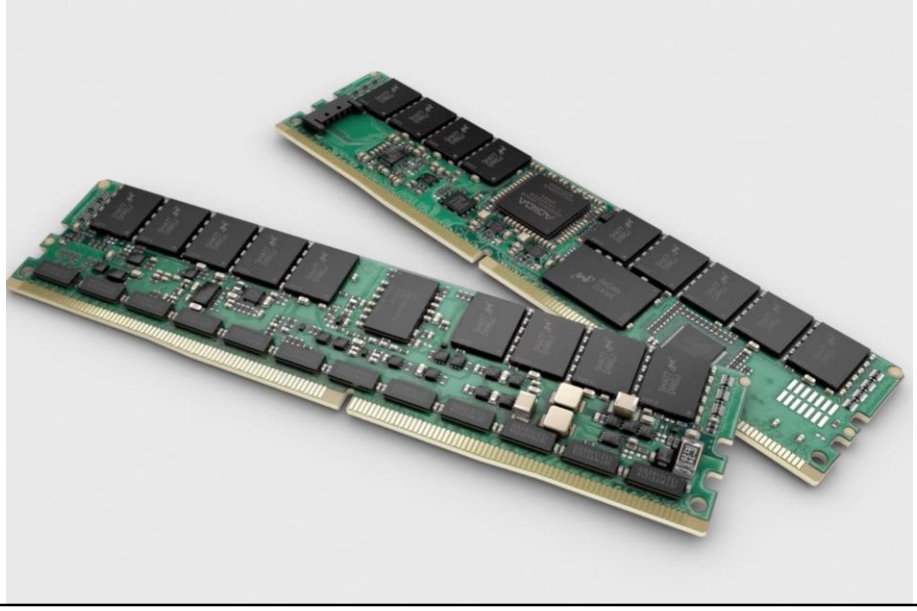
The Future is Now: Persistent Memory Support


Persistent Memory
PASS

What is Persistent Memory (PMEM)

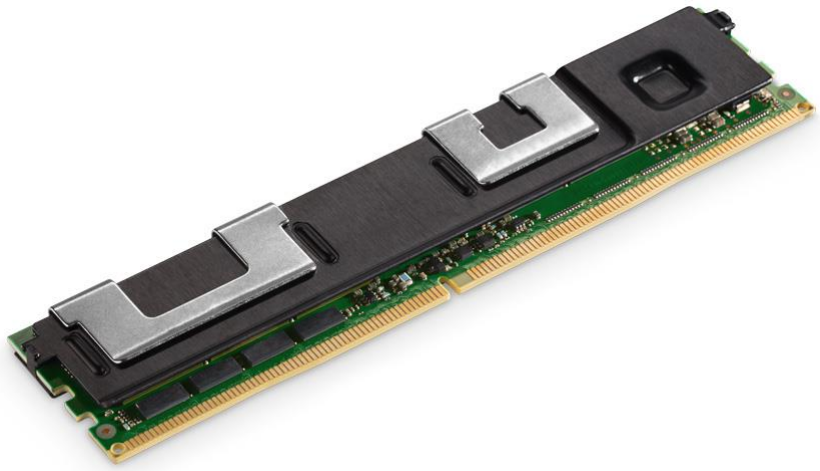


What is Persistent Memory in the year 2018 – NVDIMM-N (16GB ea.)

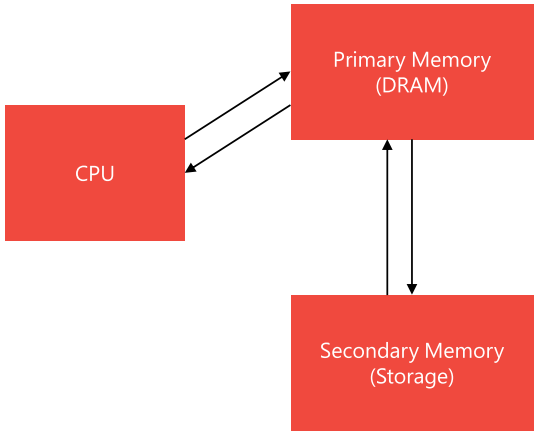


PASS

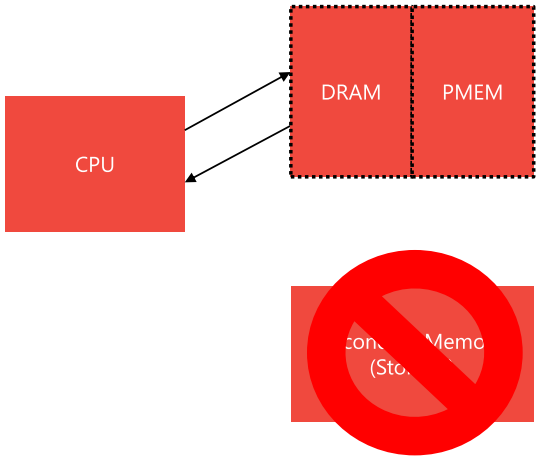
The Game Changer: High Capacity NVDIMMs (Up to 512GB ea.)



Traditional Computer Architecture



How Things Look With Persistent Memory

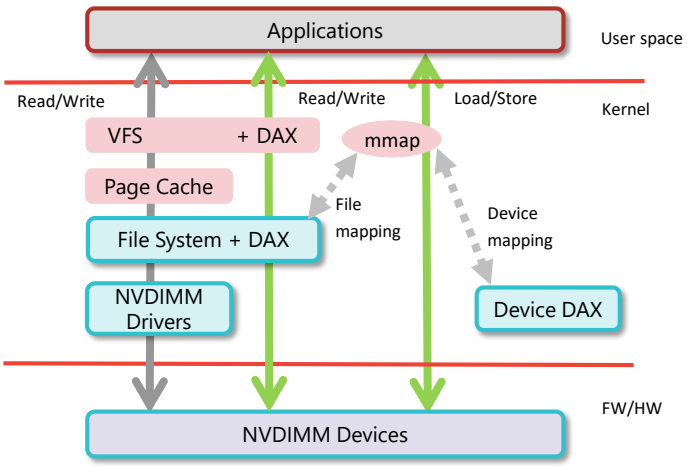


Persistent Memory native features in SQL Server

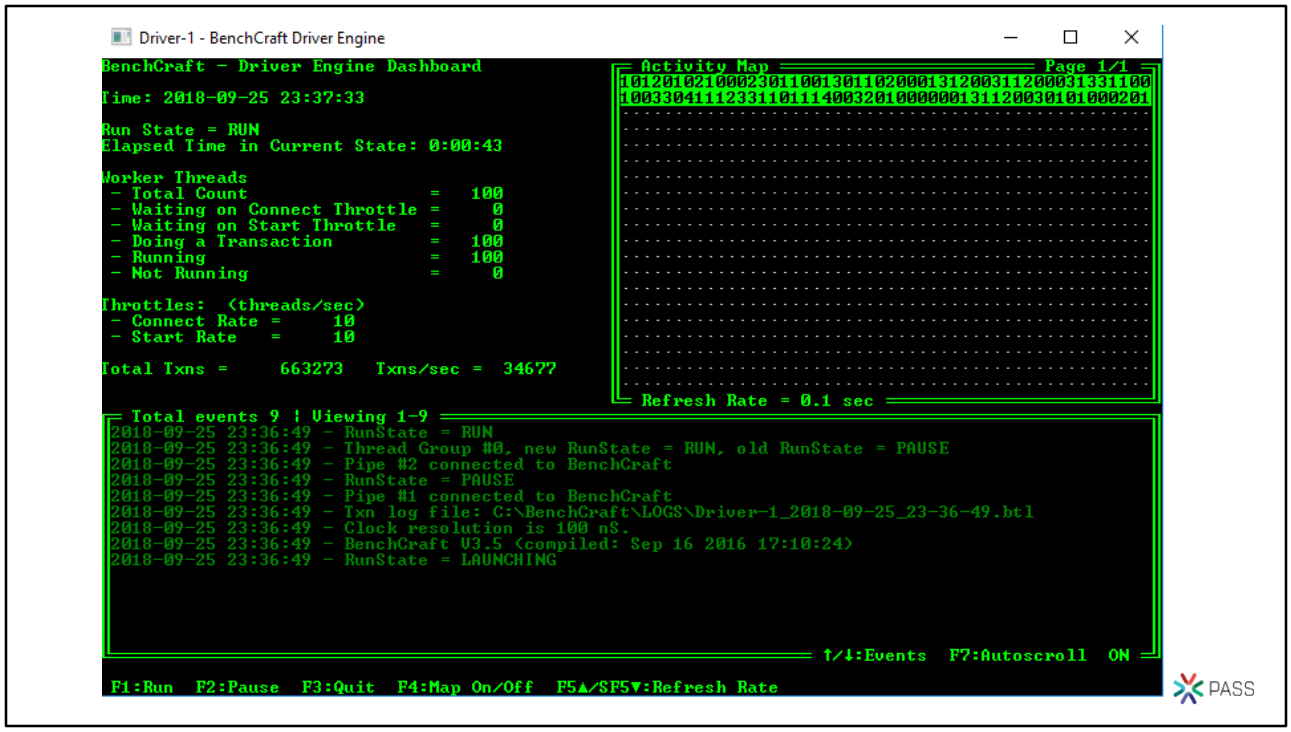
Available with SQL Server 2016+: Tail of the log caching (ToL)

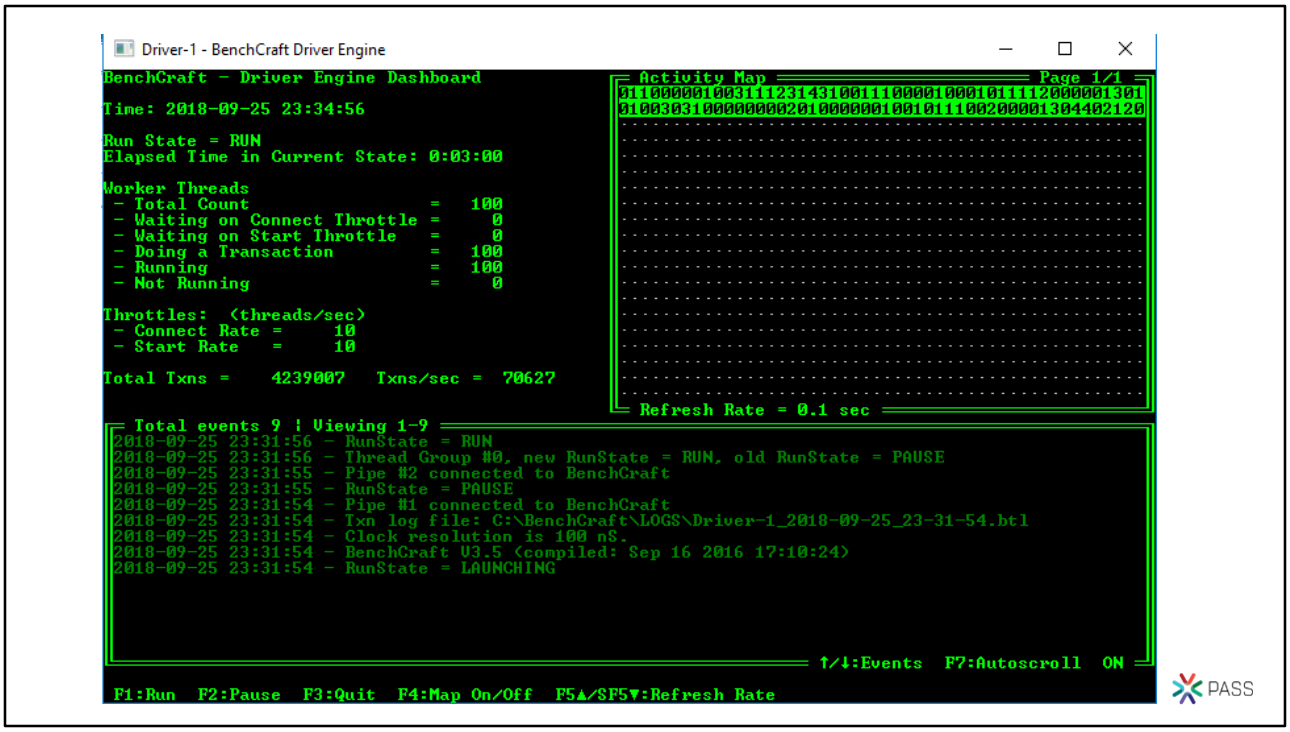
Persistent Memory native features in SQL Server 2019

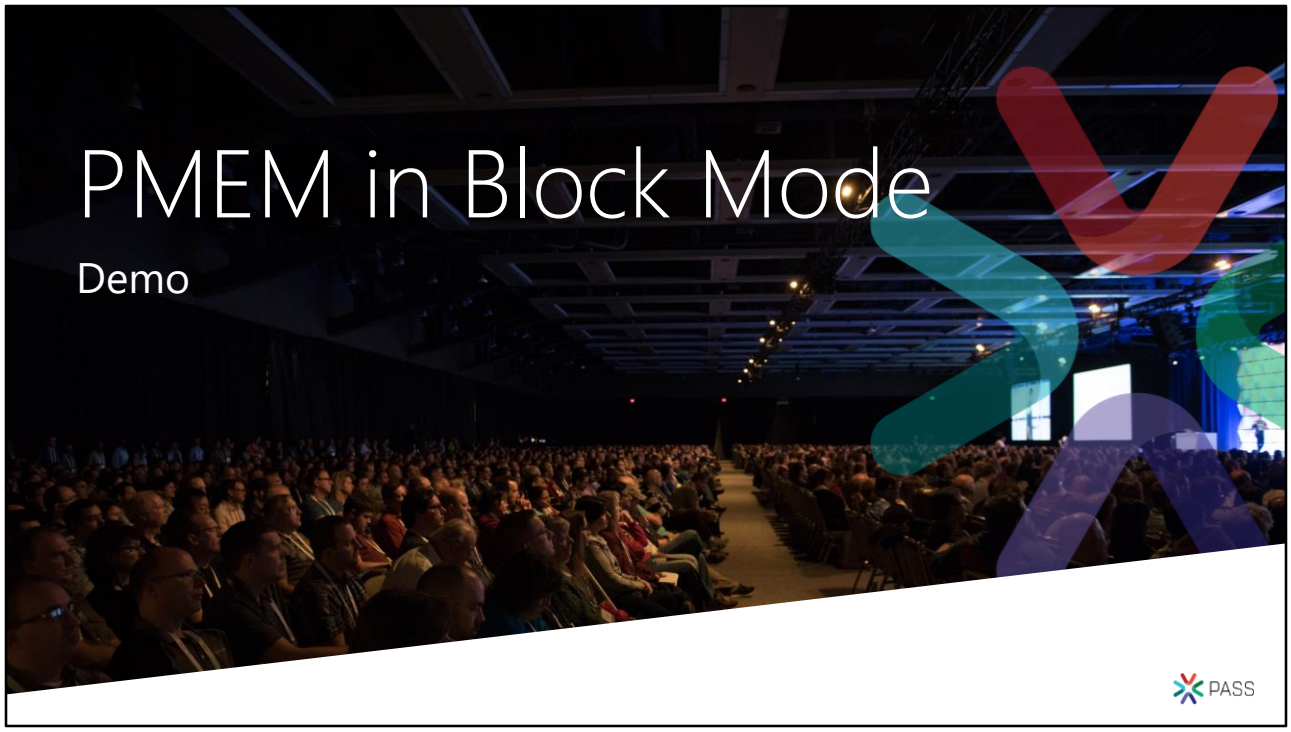
Available with CTP 2.0: *Enlightenment*



PASS


Driver-1 - BenchCraft Driver Engine
BenchCraft - Driver Engine Dashboard
Time: 2018-09-25 23:37:33
Run State = RUN
Elapsed Time in Current State: 0:00:43
Worker Threads
- Total Count = 100
- Waiting on Connect Throttle = 0
- Waiting on Start Throttle = 0
- Doing a Transaction = 100
- Running = 100
- Not Running = 0
Throttles: (threads/sec)
- Connect Rate = 10
- Start Rate = 10
Total Txns = 663273 Txns/sec = 34677
Activity Map
Page 1/1
Refresh Rate = 0.1 sec
Total events 9 ¦ Viewing 1-9
2018-09-25 23:36:49 - RunState = RUN
2018-09-25 23:36:49 - Thread Group #0, new RunState = RUN, old RunState = PAUSE
2018-09-25 23:36:49 - Pipe #2 connected to BenchCraft
2018-09-25 23:36:49 - RunState = PAUSE
2018-09-25 23:36:49 - Pipe #1 connected to BenchCraft
2018-09-25 23:36:49 - Txn log file: C:\BenchCraft\LOGS\Driver-1_2018-09-25_23-36-49.btl
2018-09-25 23:36:49 - Clock resolution is 100 nS.
2018-09-25 23:36:49 - BenchCraft V3.5 (compiled: Sep 16 2016 17:10:24)
2018-09-25 23:36:49 - RunState = LAUNCHING
↑/↓:Events F7:Autoscroll ON
F1:Run F2:Pause F3:Quit F4:Map On/Off F5▲/SF5▼:Refresh Rate
PASS


Driver-1 - BenchCraft Driver Engine
BenchCraft - Driver Engine Dashboard
Time: 2018-09-25 23:34:56
Run State = RUN
Elapsed Time in Current State: 0:03:00
Worker Threads
- Total Count = 100
- Waiting on Connect Throttle = 0
- Waiting on Start Throttle = 0
- Doing a Transaction = 100
- Running = 100
- Not Running = 0
Throttles: (threads/sec)
- Connect Rate = 10
- Start Rate = 10
Total Txns = 4239007 Txns/sec = 70627
Activity Map
Page 1/1
Refresh Rate = 0.1 sec
Total events 9 ¦ Viewing 1-9
2018-09-25 23:31:56 - RunState = RUN
2018-09-25 23:31:56 - Thread Group #0, new RunState = RUN, old RunState = PAUSE
2018-09-25 23:31:55 - Pipe #2 connected to BenchCraft
2018-09-25 23:31:55 - RunState = PAUSE
2018-09-25 23:31:54 - Pipe #1 connected to BenchCraft
2018-09-25 23:31:54 - Txn log file: C:\BenchCraft\LOGS\Driver-1_2018-09-25_23-31-54.btl
2018-09-25 23:31:54 - Clock resolution is 100 nS.
2018-09-25 23:31:54 - BenchCraft V3.5 (compiled: Sep 16 2016 17:10:24)
2018-09-25 23:31:54 - RunState = LAUNCHING
↑/↓:Events F7:Autoscroll ON
F1:Run F2:Pause F3:Quit F4:Map On/Off F5▲/SF5▼:Refresh Rate
PASS


PMEM in Block Mode
Demo
PASS

Persistent Memory native features in SQL Server 2019

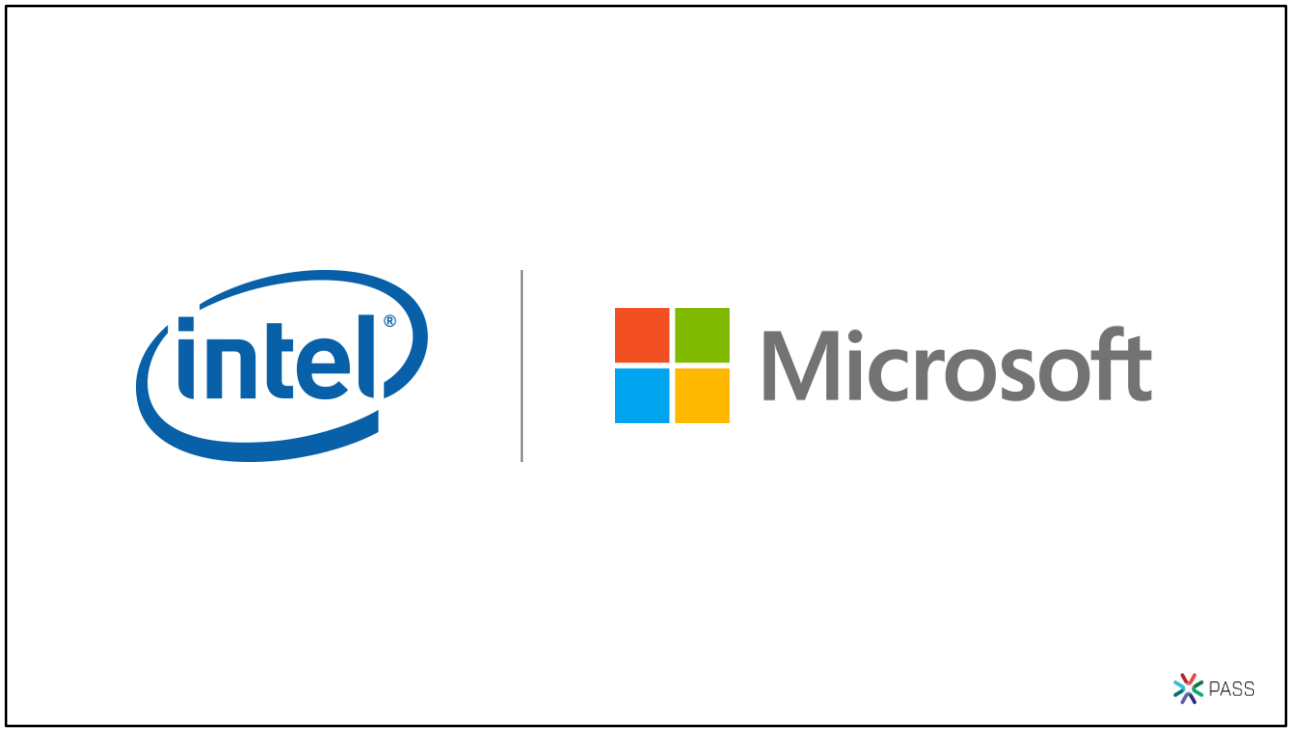
Announcing! Available with CTP 2.1: Hybrid Buffer Pool

Expanded Support for Persistent Memory Devices

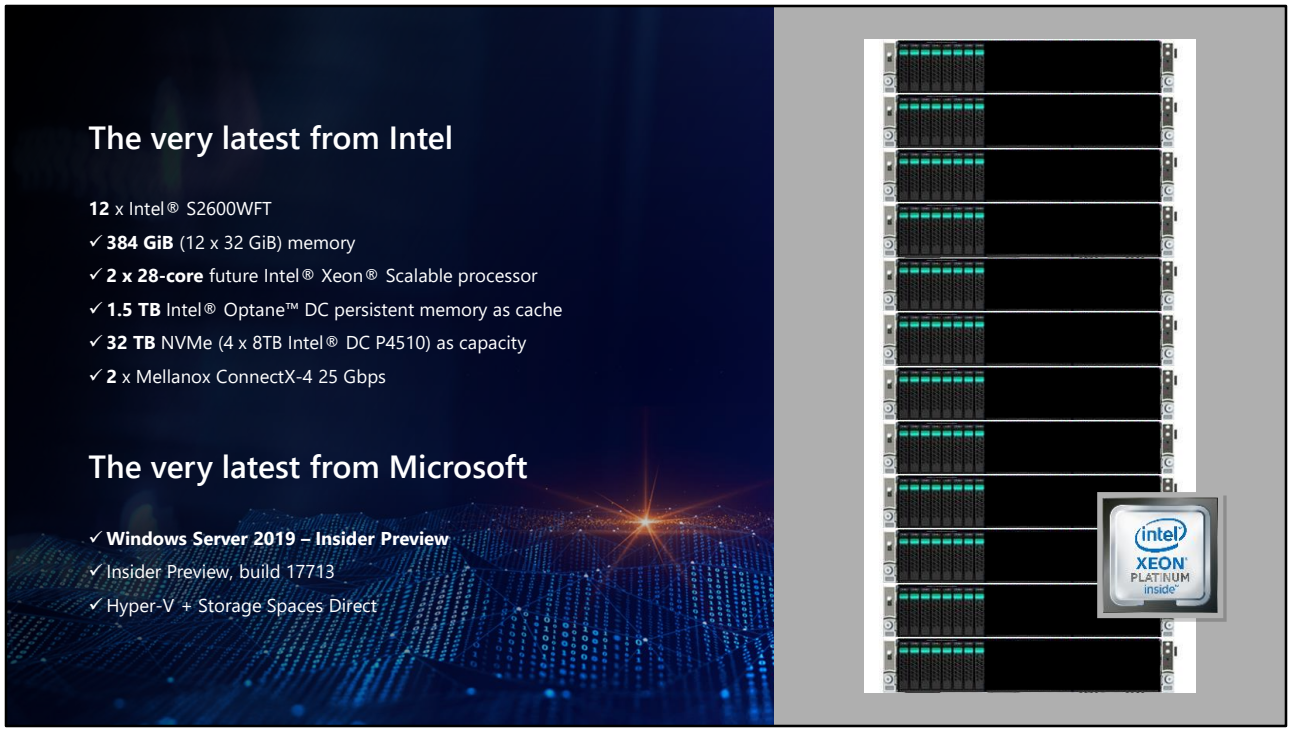
SQL Server Release	NVDIMM-N in Block Mode (Windows Server 2016+)	High Capacity NVDIMMs in Block Mode (Windows Server 2019+)	*Enlightenment* of SQL Server Files	Hybrid Buffer Pool (Windows Server 2019)
2014	Yes	Yes	N/A	N/A
2016	Yes	Yes	N/A	N/A
2017	Yes	Yes	No	N/A
2019	Yes	Yes	Yes	Yes

PASS

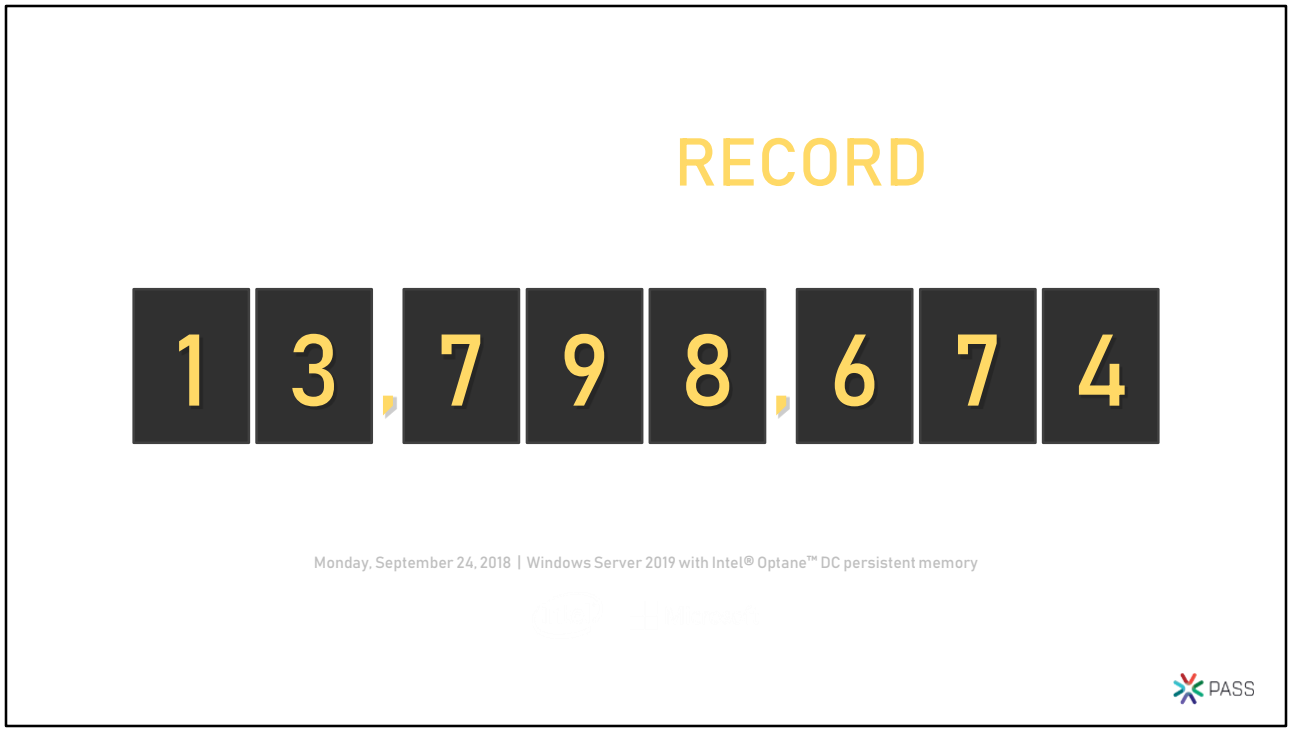

Latest Gen HCI
PASS



The demo you're about to see is the result of a deep engineering collaboration between Microsoft and our friends at Intel, and it will show the very latest in storage innovation:



A reference design that Intel and Microsoft have been working on together: 12 server nodes, running Windows Server 2019, each packed with a future Intel Xeon Scalable processor, persistent memory, and NVMe.



We believe, the industry record for any hyper converged platform.

Session resources

Upgrade SQL Server

Database Migration Guide

Microsoft Assessment and Planning Toolkit

Overview of Data Migration Assistant

DEA 2.1 General Availability: Release Overview – Database Experimentation Assistant

Post-migration Validation and Optimization Guide

http://aka.ms/dbcompat (DB Compatibility Level based upgrades)

PASS

Bookmarks

SQL Server Team (Tiger) Blog	http://aka.ms/sqlserverteam
Tiger Toolbox GitHub	http://aka.ms/tigertoolbox
SQL Server Release Blog	http://aka.ms/sqlreleases
BP Check	http://aka.ms/bpcheck
SQL Server Standards Support	http://aka.ms/sqlstandards
Trace Flags	http://aka.ms/traceflags
SQL Server Support lifecycle	http://aka.ms/sqllifecycle
SQL Server Updates	http://aka.ms/sqlupdates
SQL Server Guides	http://aka.ms/sqlserverguides
SQL Feedback (New "Connect")	http://aka.ms/sqlfeedback
T-SQL Syntax Conventions	http://aka.ms/sqlconventions
SQL Server Errors	http://aka.ms/sqlerrors
Twitter	@mssqltiger

PASS

Thank You

Learn more from the Tiger Team

@mssqltiger

https://aka.ms/sqlserverteam