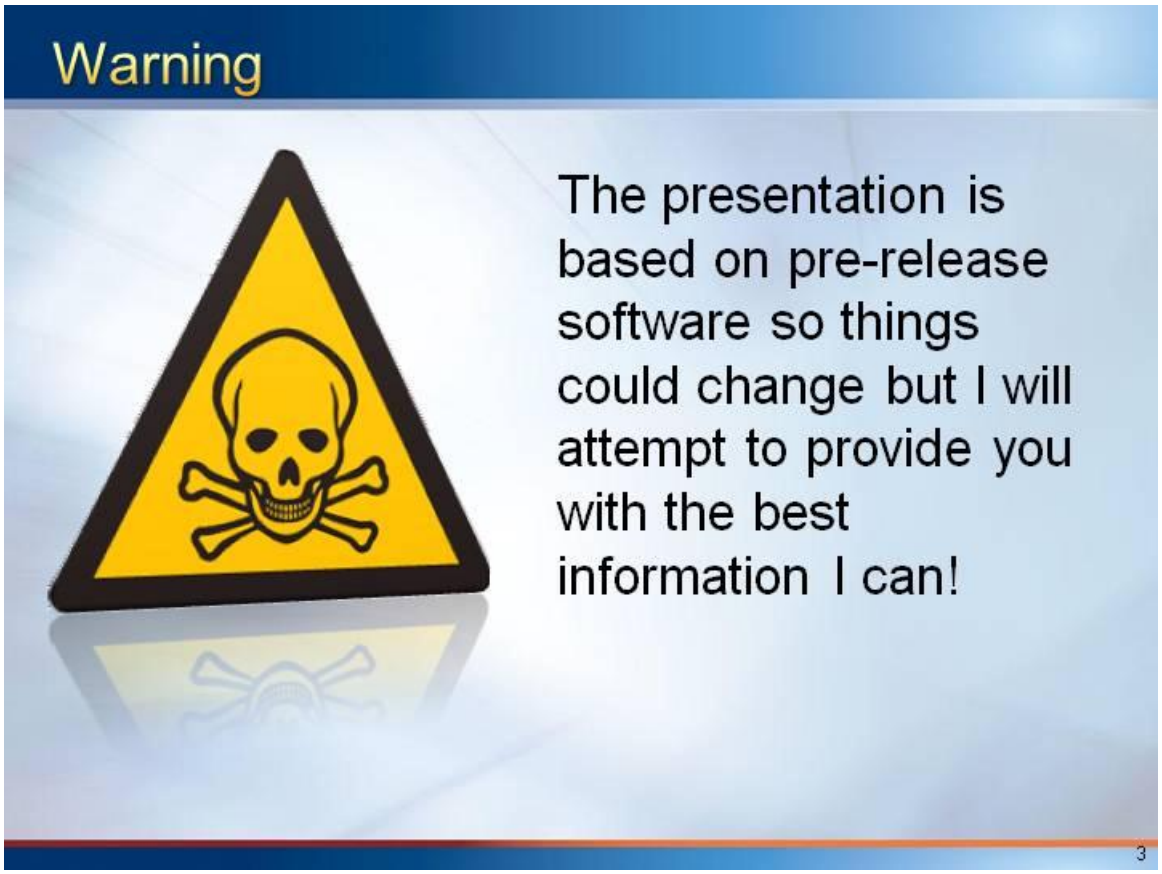
SQL Server Denali

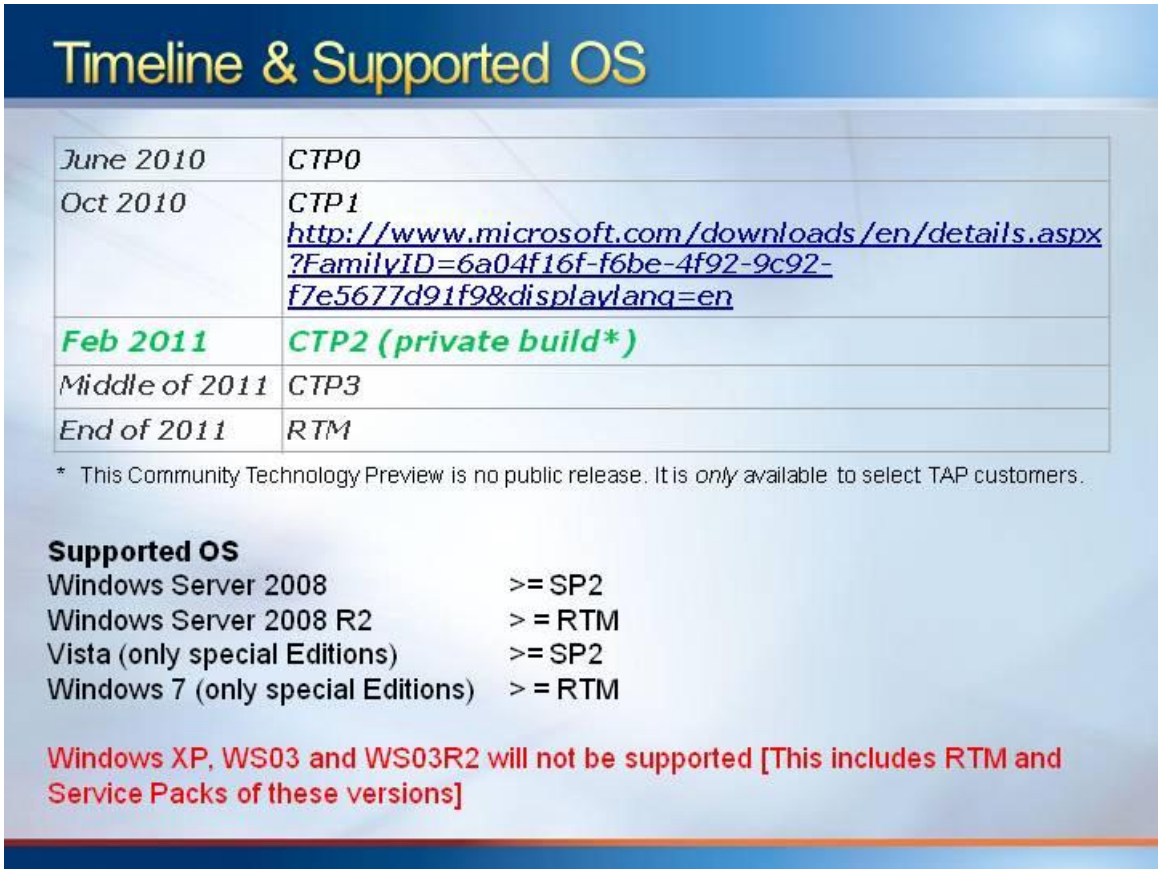
SQL PASS



<SPEAKER>

All information in this presentation are public available. You can find these in BOL, Blogs, or other publications.

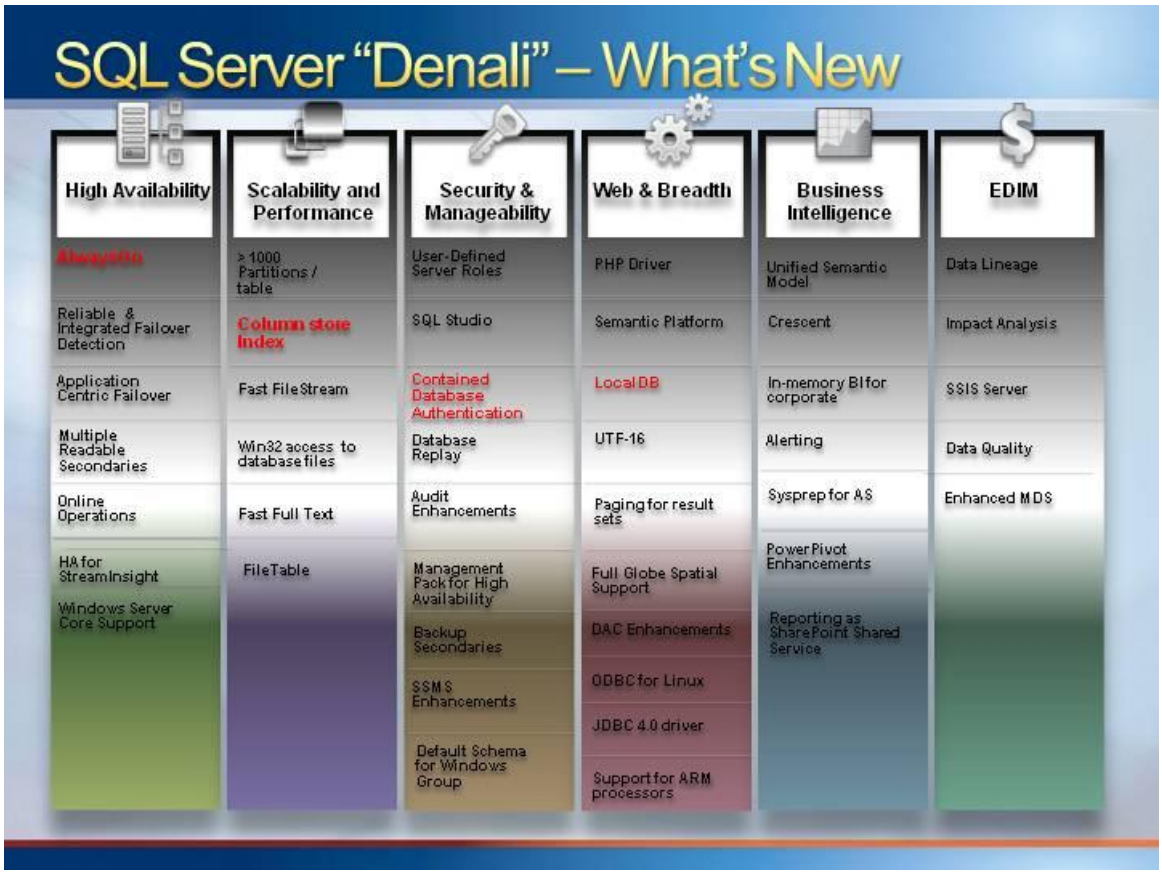
</SPEAKER>



CTP1: Release during SQL Pass 2010 conference in Neuss

<Linklist>

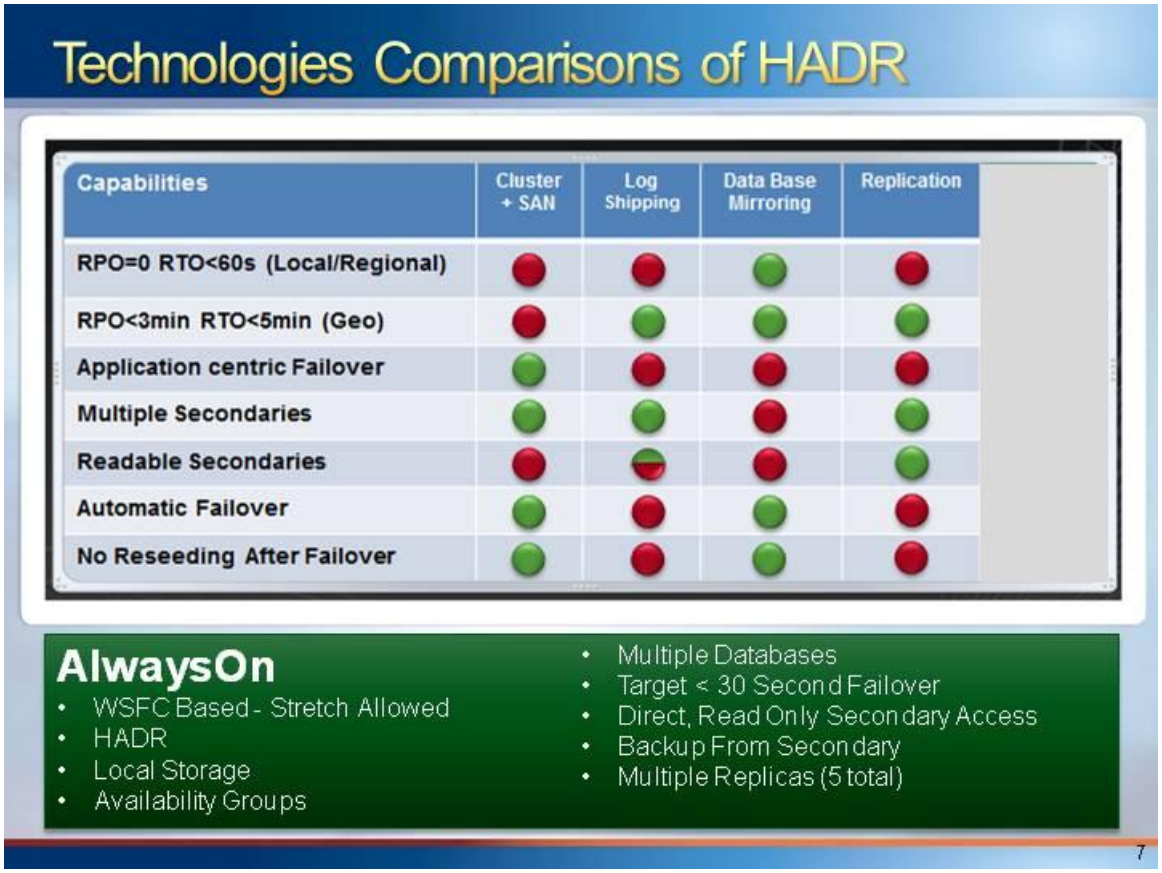
http://www.microsoft.com/downloads/en/details.aspx?FamilyID=6a04f16f-f6be-4f92-9c92-f7e5677d91f9&displaylang=en



EDIM = Enterprise Data Integration and Management

In particle physics, a **hadron** (Greek: αδρός, *hadrós*, "stout, thick") is a composite particle made of quarks held together by the strong force (as atoms and molecules are held together by the electromagnetic force). Hadrons are categorized into two families: baryons (made of three quarks) and mesons (made of one quark and one antiquark).

The best-known hadrons are protons and neutrons (both baryons), which are components of atomic nuclei.



<SPEAKER>

recovery point objective (RPO)

recovery time objective (RTO)

geographically disperse cluster (geo)

</SPEAKER>

AlwaysOn Failover Clustering Instances

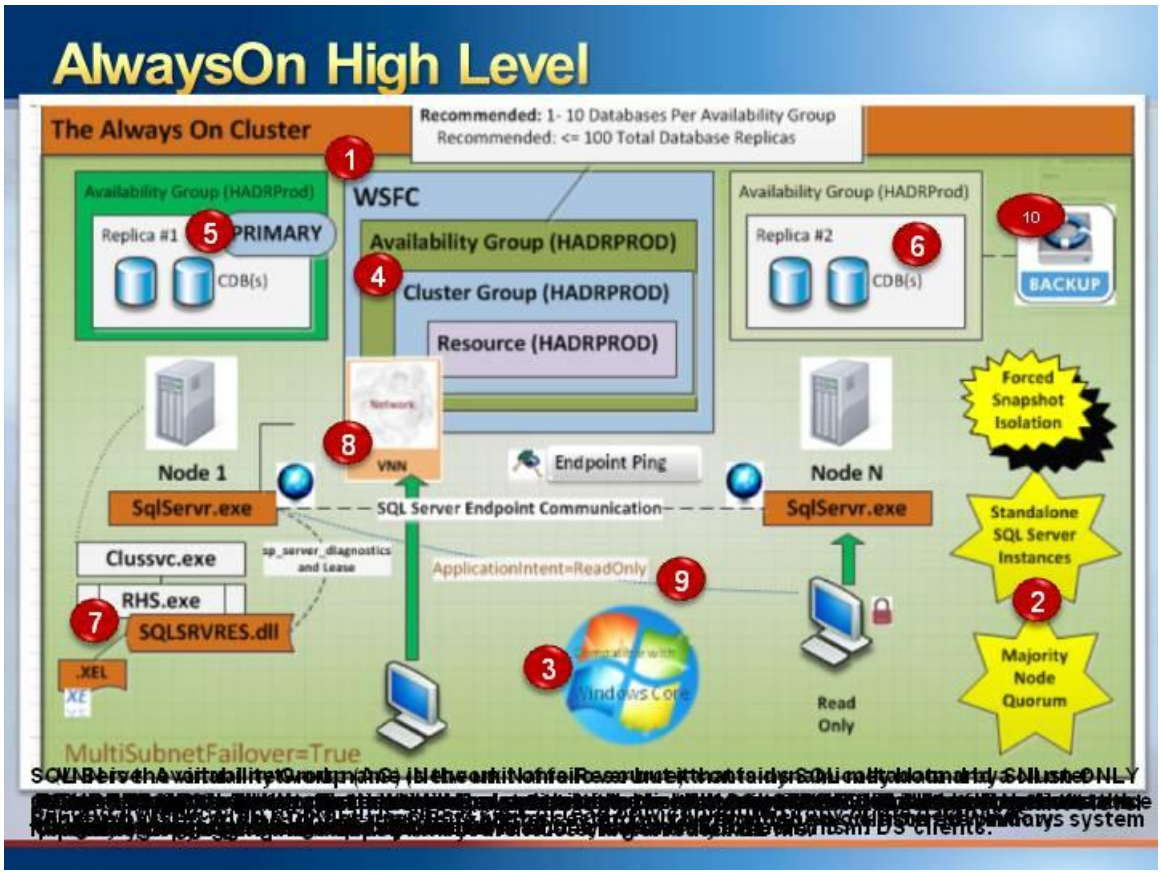
- Multisite Clustering
- Flexible Failover Policy
- Improved Diagnostics
- Built for consolidation scenarios

SQL Server AlwaysOn is the new high availability and disaster recovery solution for SQL Server. Using AlwaysOn businesses can achieve increased application availability for their mission critical applications and get higher returns on their high availability investments through better utlization of hardware resources. AlwaysOn also increases productivity and lowers TCO by greatly simplifying high availability deployment and management.

The new solution will be available in the next release of SQL Server, code named Denali. The initial feature set of AlwaysOn was available in CTP1 that was just released during the SQL PASS 2010 conference.

AlwaysOn Availability Group for Databases

- Multi-Database Failover
- Multiple Secondaries
- Active Secondaries
- Integrated HA Management



1 – Based on WSFC and no longer a separate witness. Can only be enabled on a clustered Windows system

2 – Any cluster quarom can be used but majority node it the most likely with simple election and shared nothing. Instances of SQL can be clustered but do not need to be unless you are really in need of shared resource usage and the associated dependencies.

3 – Does run on and will be targeted at Windows core because of the significant patch reduction that provides better uptime and HA capabilities

4 – SQL Server Availability Group (AG) is the unit of failover but it contains SQL metadata and a cluster group. The cluster group contains the AG resource and can also contain the VNN (Network Name Resource) that is setup as a dependency for the AG cluster resource.

5 – Primary Replica owns the Read/Write copy and this is where the SQLSRVRES.dll is handling the health detection, logging of diagnostics and IsAlive/Online Thread lease mechanism.

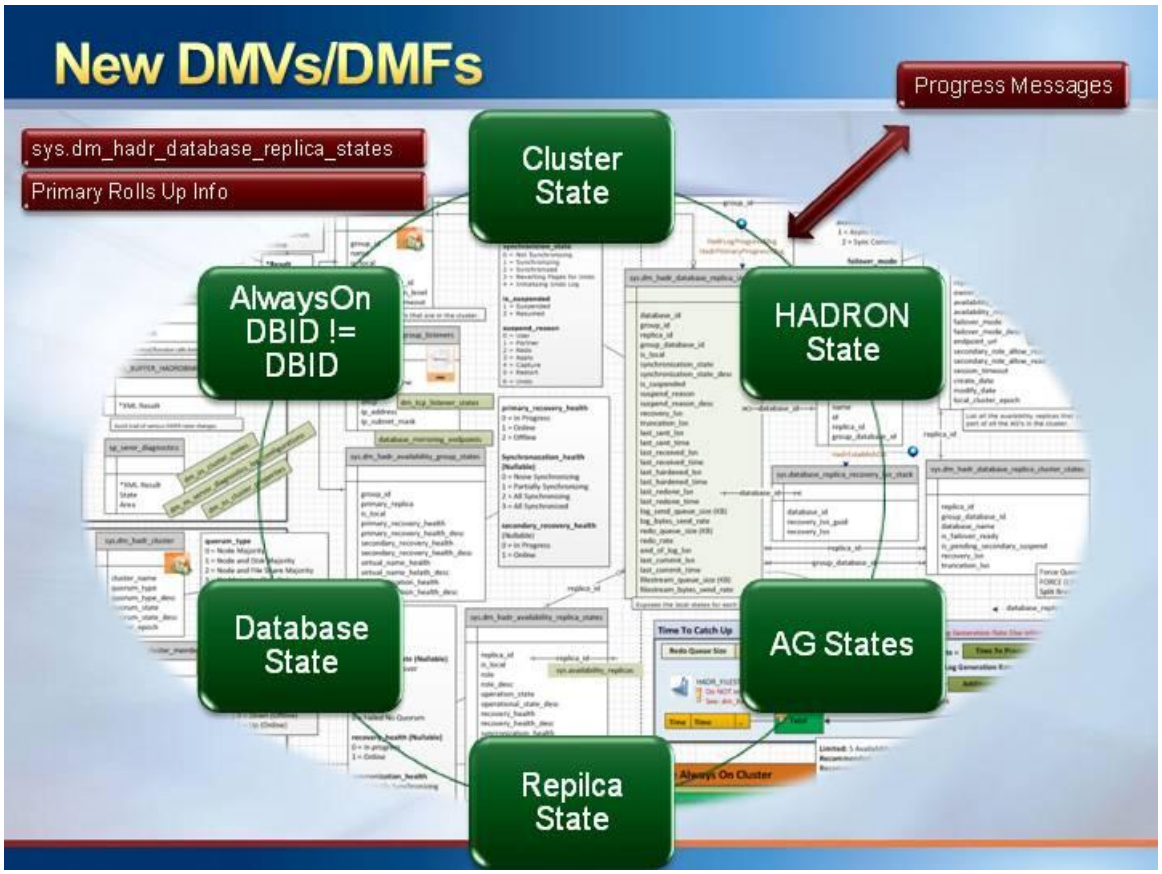
6 – Secondary replica (up to 4 total) is a no access or read only configured copy of the databases.

7 – SQLSRVRES.dll contains the cluster resource DLL logic for the AG as well as the code for instance level failover (both new sp_server_diagnostics and the downlevel select @@SERVERNAME) logic. Logging now takes place in the cluster log and the XEL file.

8 – VNN is the virtual network name (Network Name Resource) that is dynamically bound by SNI on ONLY THE PRIMARY. The application need only connect to the VNN name and it will always connect to the primary copy of the database.

9 – Connecting to the VNN with the additional option of ApplicationIntent=ReadOnly allows the client to activly redirect the client to a valid, online read only secondary for new 7.5 > TDS clients.

10 – You can backup WITH COPY or run DBCC on the secondary to offload the work from the primary.



There are ~12 new DMVs that are HADRON related.

Running the DMVs on the primary will show secondary information.

Running the DMVs on the secondary will show ONLY secondary information.

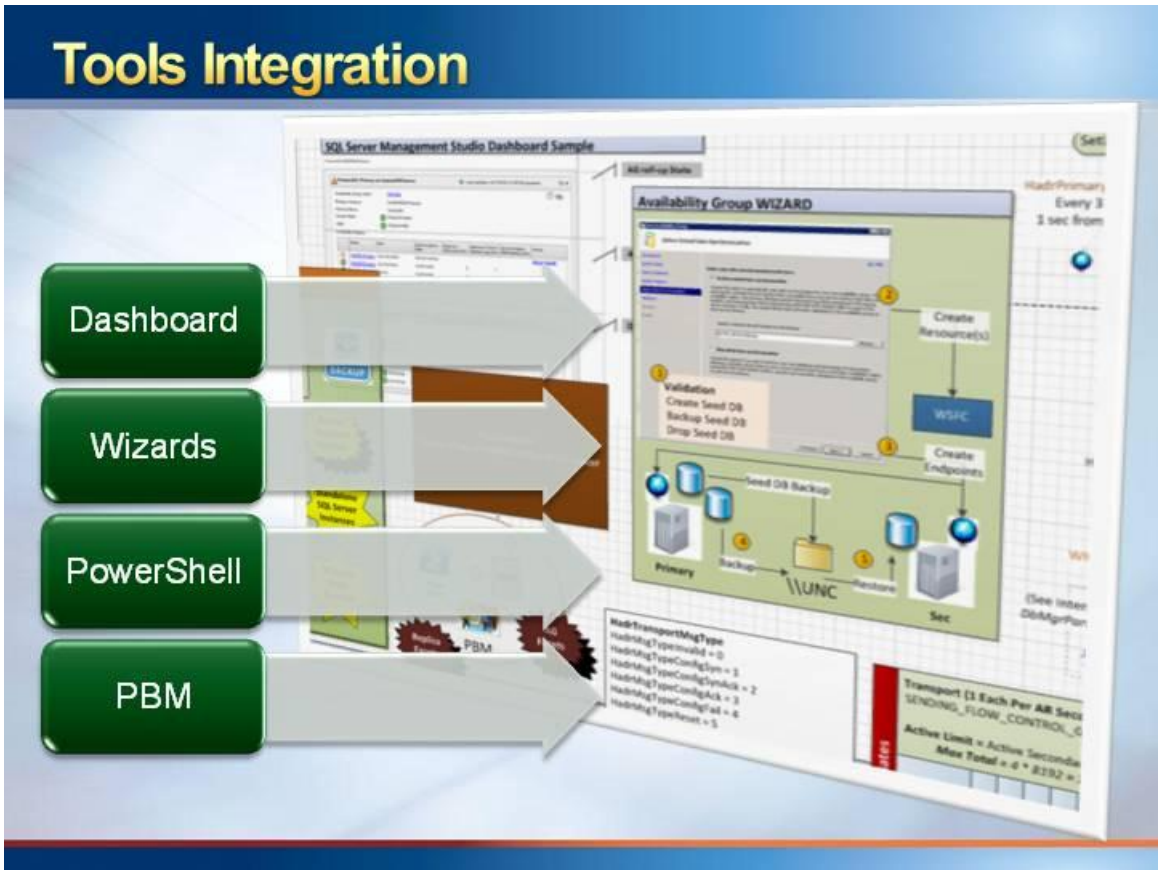
The progress messages that are returned from the secondary's contains state information that is stored in memory for use by the DMVs and other HADRON decisions. The primary does not propagate the information back other secondary's. It is only a collection location.

This is why there are some questions about the UI that are being fixed. For example, running the Dashboard against a secondary shows strange errors right now.

The icons in the SSMS tree will be limited down because there is not enough information on the secondary to show Primary and other secondary states.

CTP 3 will be better with state indicators.

DBID in some of the HADRON DMVs is the DMV within the AG and not that assigned on the server as they may not be the same across all instances the AG is assigned to.

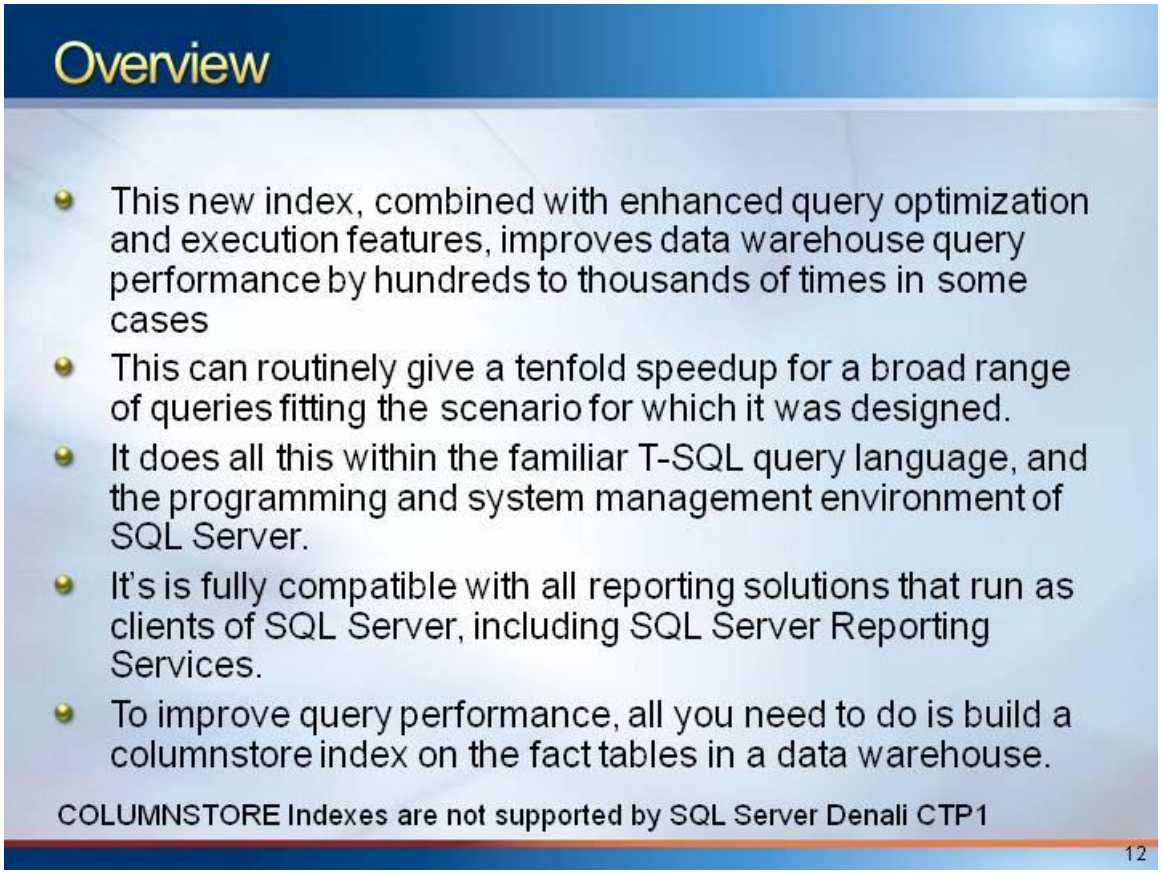


The Managability team has been working to integrate the SQL Server Always On feature set. There are the dashboard, wizard to create AG, PBM policies that run the dashboard that can all be accessed from Powershell and similar utilities.

The SQL Server 11.0 release (code named “Denali”) introduces a new data warehouse query acceleration feature based on a new type of index called the columnstore. This new index, combined with enhanced query optimization and execution features, improves data warehouse query performance by hundreds to thousands of times in some cases, and can routinely give a tenfold speedup for a broad range of queries fitting the scenario for which it was designed. It does all this within the familiar T-SQL query language, and the programming and system management environment of SQL Server. It’s thus fully compatible with all reporting solutions that run as clients of SQL Server, including SQL Server Reporting Services.

Column Store Index as the name implies is used to store data in Column-Oriented format. By using encoding and compression techniques, Apollo is able to achieve 10-1000x performance gains for targeted queries or 5-10x performance gains for Data Warehousing workloads.

Apollo builds on Vertipaq algorithms (used in PowerPivot) to achieve near instant response times through in-memory processing and ability to process multiple rows (batches) at a time instead of one row at a time.



A columnstore index stores each column in a separate set of disk pages, rather than storing multiple rows per page as data traditionally has been stored. We use the term “row store” to describe either a heap or a B-tree that contains multiple rows per page.

Near-instant response time

- Users can explore data, get more business value from it

Predictable, uniform response time for a broad range of queries

- “Just scan all the data”

Easy to set up – doesn’t require deep expertise

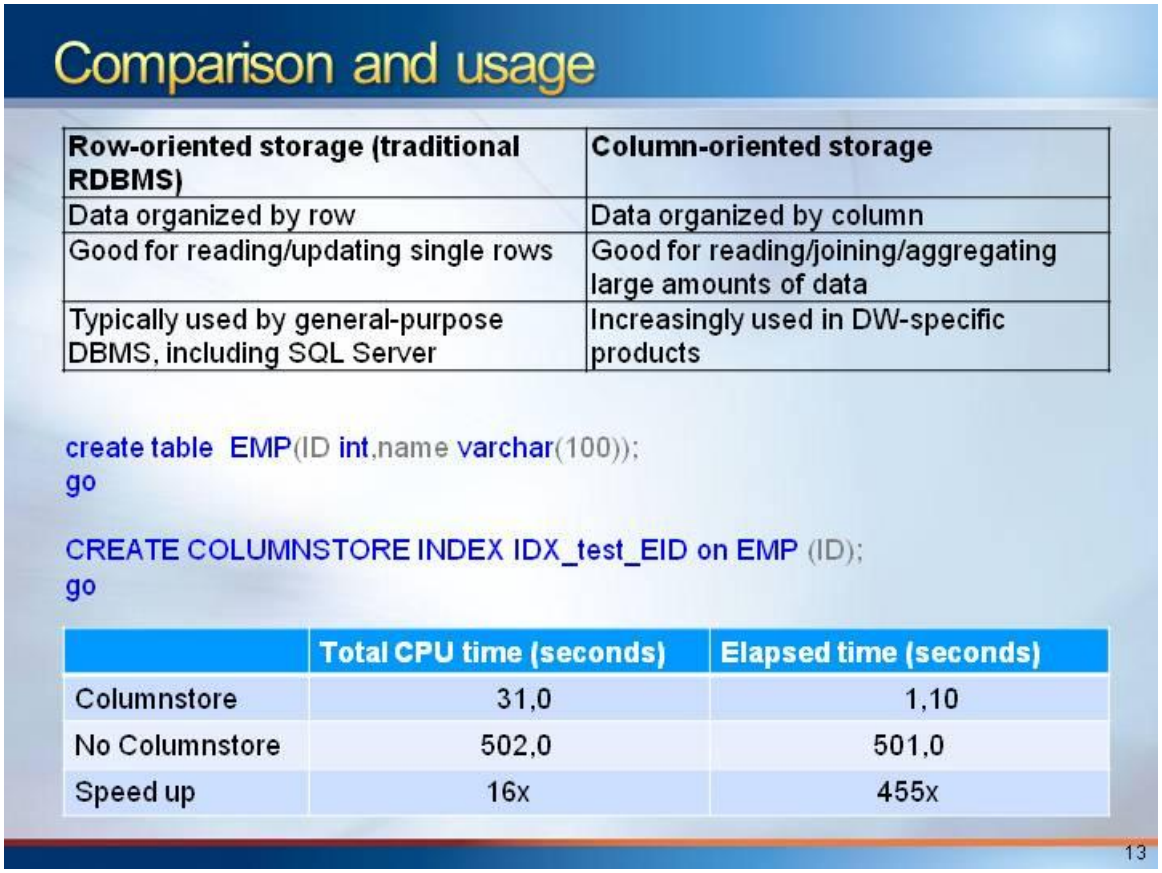
- No need to build summary aggregates (indexed views, summary tables, OLAP cubes)
  - Shortens ETL time
  - Simpler to use than aggregates
  - More predictable performance than aggregates
- Fewer indexes to design, create, and maintain

- Reduces need for developers to manually tune queries

Lower TCO

- Reduced people costs
- Lower hardware costs

Building block of PDW (Madison) for data warehouses up to 100s of TB with sub-second query response time



Traditional RDBMS like SQL Server store data in row-based format and is suitable for OLTP workloads.

Performance test was done on a 32-logical processor machine with 256GB of RAM, we ran this query on a pre-release build of Denali, with and without the columnstore index on the fact table

Serverless is a way to ship just the engine bits of SQL Server. It installs the binaries but not as a service, no agent, etc...

When the client uses the AttachDB=MyDatabase.mdf the (RANU) like behavior will occur.

The provider components for SQL Denali will look for an instance of SQLServr.exe running under the security context of the current user. If a running process exists the database is attached to that process. If not SqlServr.exe is launched (CreateProcess) under the user context and the database attached.

This allows an applicaiton to just ship the code and MDF and the simple Serverless package. When the application needs SQL Server the providers will launch the process on demand, not a service that has to be running.

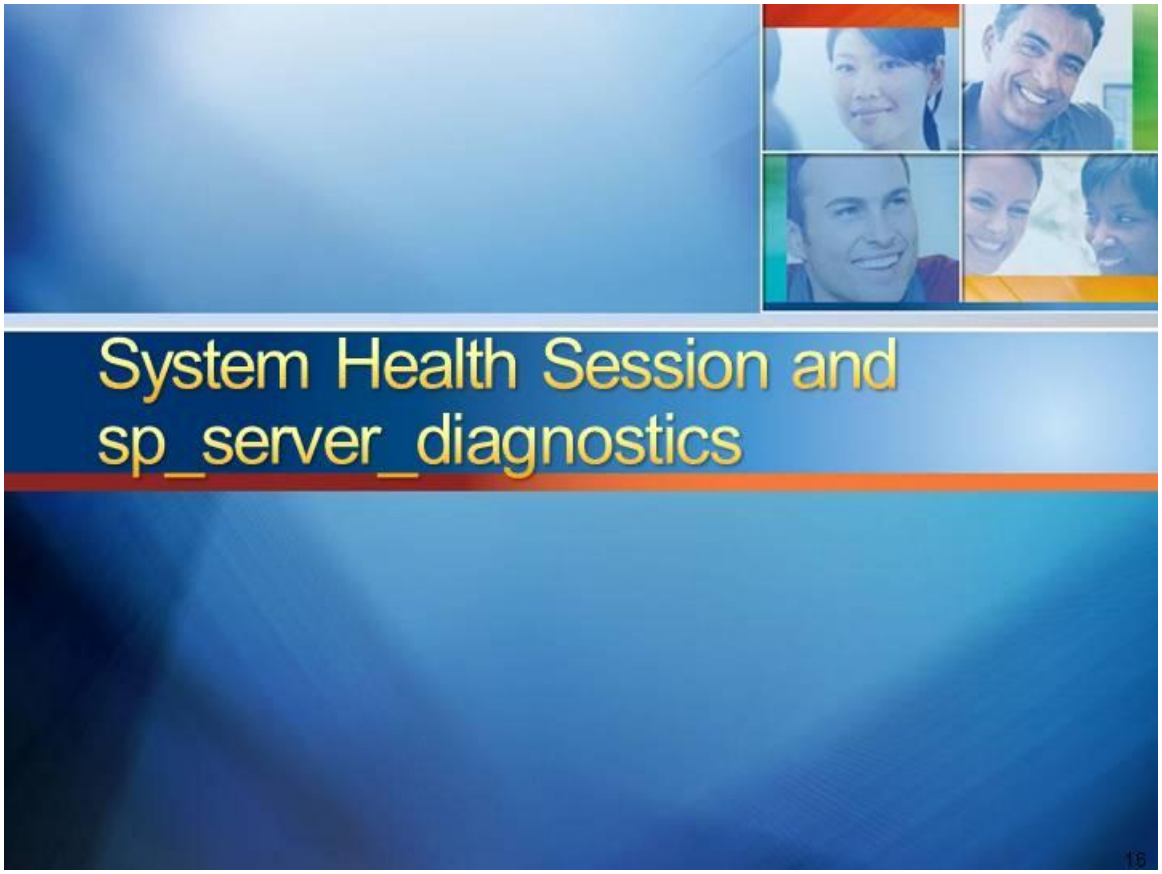
After ~5 minutes of idle time the process will terminate and be restarted when demand for it is needed again.

This is working with the SQL Server Development Studio as well. When you create a SQL Server project and press F5 in VSTS the default it use the serverless, SQL Server to support the development experience. So the developer has a fully integrated experience and it just works.

Tracing and Trace flags: Registry based and apply globally to all serverless instances

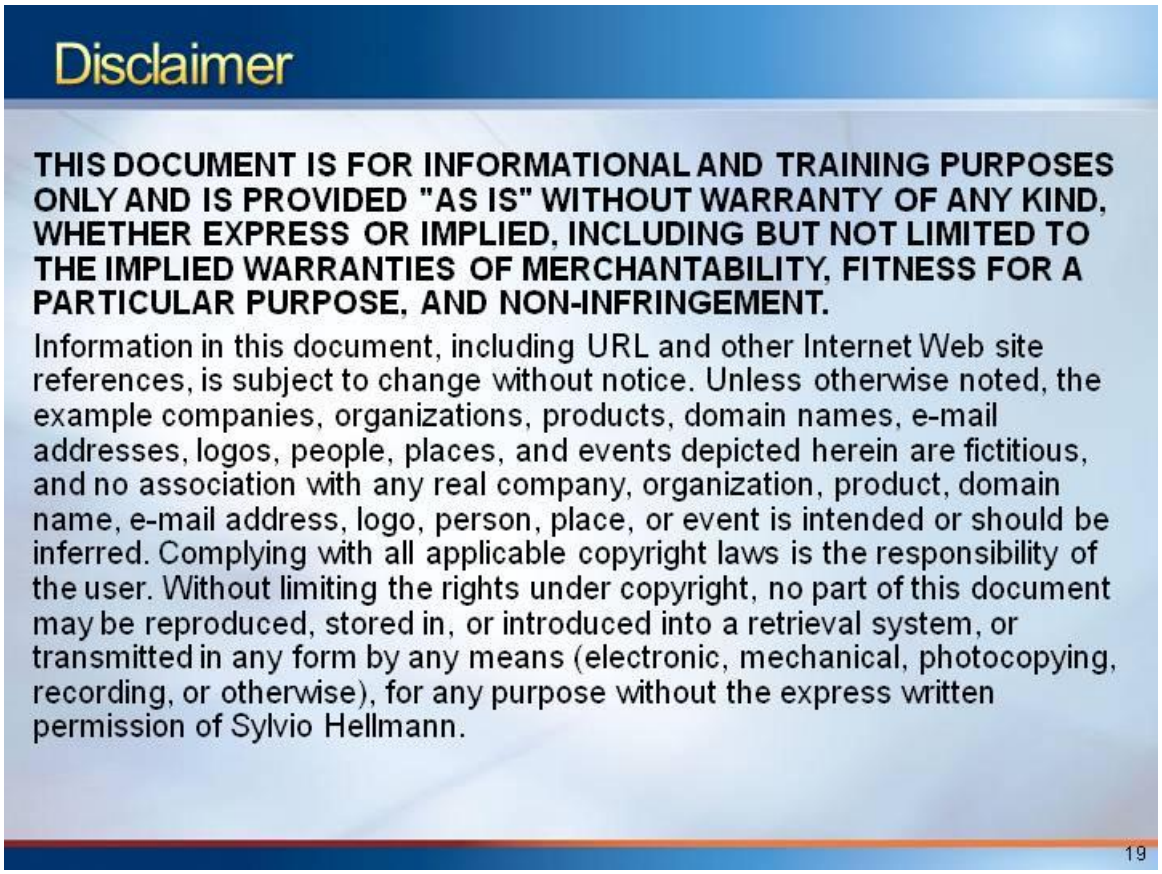
Install: Single copy of the binaries

User Based: Each user account will get a sqlservr.exe process in a sandbox approach, running under the specific user credentials




Q & A
17


Thank You!
Thanks for:
Your attention
Your support and
Your time
감사합니다
Obrigado
Dakujem
Merci
Tak
ευχαριστώ
Mulţumesc
Dziękuje
Gracias
謝謝
Takk
ありがとう
Grazie
תודה
Kiitos
Danke
谢谢
Спасибо
Dank u
Asante
dhanayawad
شكراً


Disclaimer
THIS DOCUMENT IS FOR INFORMATIONAL AND TRAINING PURPOSES ONLY AND IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, WHETHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT.
Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in, or introduced into a retrieval system, or transmitted in any form by any means (electronic, mechanical, photocopying, recording, or otherwise), for any purpose without the express written permission of Sylvio Hellmann.
19