SAP® Adaptive Server® Enterprise 16

New Features Overview

TABLE OF CONTENTS

EXTREME TRANSACTION PROCESSING: A BUSINESS REALITY

The amount of information that organizations process constantly grows, and so does the volume of human and machine driven transactions performed.

With the number of business transactions on the internet expected to reach 450 billion a day, in 2020[1] and the number of non-cash payment transactions processed in 2012 reaching 333 billion globally[2], transactional loads are increasingly measured in millions per hour. Users' response time expectations are also changing. Gone are the times when applications returned information in seconds; responses are now expected within milliseconds and microseconds. As an example, at NASDAQ an average transaction that in 2006 used to take about one millisecond to complete, now executes in less than 100 microseconds, even though the average number of transactions processed has increased from 53,000 to 500,000 per second in the same time period[3]. And this trend is not only found in the financial services sector. Google recently concluded that a 400 millisecond response time is considered too long by the average user, and other researchers determined that people will visit a web site less often if it is slower than a close competitor by more than 250 milliseconds[4].

Additionally, IDC forecasts that global telecom service providers will reach 30 billion autonomously connected end points by 2020[5], generating exponential growth in the number of transactions and the velocity at which they are generated. Fueled by an ever-increasing number of transactions, data volumes are also increasing at an unprecedented pace. In the USA alone, the size of digitalized data is forecasted to reach 6.6 zettabytes in 2020, which represents a 25% CAGR from 2012[6].

These changes create a number of new challenges for enterprises across all industry sectors. Businesses not only need to process extremely large number of transactions on vast amount of data, but they also have to return information to users extremely fast. Moreover, with costs as high as $199 per data record unlawfully exposed[7] and with breaches increasingly affecting tens of millions records, protecting the data privacy of these vast amounts of data has become a top concern of every enterprise. This adds to the overall cost and complexity of managing transaction intensive data environments that are required to cope with the evolving data needs of the enterprise.

SAP ADAPTIVE SERVER ENTERPRISE 16 – EXTREME TRANSACTION READY

SAP Adaptive Server Enterprise (ASE) has been recognized over the years for its ability to efficiently scale to support large number of concurrent transactions and deliver record performance, economically. Its widespread adoption in transaction extreme industry sectors, such as banking, has delivered a wealth of experience to SAP, allowing its engineers to deliver, with SAP ASE 16, unprecedented levels of speed, scalability, security and simplicity, to meet the increasing demand for extreme data transaction processing.

SAP ASE 16 delivers increased scalability and speed with extensive optimizations in its transaction concurrency management, query plan execution, data compression and utilization of computing resources in large SMP servers. Preliminary internal OLTP benchmarks on 80 CPU cores show a linear scalability with transaction throughputs as high as one million transactions per minute.

In SAP ASE 16, both the security enforcement and system auditability have been augmented to deliver customers more flexibility in the way they comply with specific compliance requirements. A broader integration with SAP Control Center and a wealth of other enhancements also deliver simplified management of SAP ASE installations, reducing their overall cost of ownership.

Finally, additional improvements in the integration with other SAP products, such as SAP Business Suite and SAP HANA, further simplify the management and increase the reliability of SAP-centric installations, also lowering their overall TCO.

In a nutshell, SAP ASE 16 has been designed from the ground up to deliver the speed, scalability, security and simplicity that enterprises need to meet the challenges of extreme transactional workloads, efficiently and economically.

[1] IDC, The Digital Universe in 2020, 2010

[2] Capgemini World Payments Report, 2013

[3] Data Center Knowledge, NASDAQ Steps Up Its Game in the Data Center

[4] The New York Time, For Impatient Web Users, an Eye Blink Is Just Too Long to Wait, 2012

[5] IDC's predictions for 2014

[6] IDC, The Digital Universe in 2020, 2013

[7] Ponemon Institute, Cost of Data Breach Study: Global Analysis, 2013

Speed Enhancements

Many factors affect the response time of data management systems that operate under heavy transactional workloads. But undoubtedly, the ability to reduce disk I/O, increase the amount of data managed in main memory, improve CPU utilization, and parallelize the execution of query plans have a major positive impact on performance. SAP ASE 16 addresses all these performance dimensions with enhancements in index management, compression, and query plan optimization. It also delivers specific performance improvements in the way it handles data loads for large SAP Business Warehouse installations.

Index Compression

The optimization of disk I/O operations has an enormous impact on the performance of data management systems and compression is recognized as a valuable technique to cut the number of these operations while reducing storage costs. Different database structures however require dedicated compression techniques to deliver measurable cost and performance benefits.

Previous releases of SAP ASE have delivered advanced data compression of rows, pages, database backups and LOBs, such as XML documents, image and video. SAP ASE 16 adds support for index compression. DBAs can use *sp_configure* and *set index compression* to enable or disable index compression. They can also specify index compression at the table level, using *create table* and at index level, using *create index*. Both these commands have a newly updated syntax to support this new feature. Index compression settings can then be modified using the appropriate *alter index* or *alter table* command. The *reorg rebuild* command has also been updated to support index compression and can be implemented as an online operation.

Index compression in SAP ASE 16 adds to the value delivered by other data compression techniques. It helps reduce both the number of page allocation related operations and the occurrences of index block splits, improving system performance and data access concurrency. Additionally, by allowing for larger portions of an index to reside in main memory, it accelerates data access and further reduces disk I/O.

Query Performance Enhancements

When heavy workloads operate over large amounts of data, query performance can degrade substantially. To prevent this problem, data management systems must optimize the execution of query plans by parallelizing operations, reducing the amount of data processed, and using memory and computation resources effectively. SAP ASE 16 improves over its previous releases with a number of optimizations in query plan executions to meet the requirement of demanding workloads over large data sets.

Sort Operator Performance Improvement

SAP ASE 16 improves the performance of parallel queries that include *sort* operators in two major ways. First, this new capability splits the *sort* operator into two operators: *sortbuild* and *getsorted*. *sortbuild* creates a single sort table that is used by multiple instances of *getsorted* for a parallel reading of sorted rows, reducing disk I/O operations. Additionally, *sortbuild* sends a single metadata row to multiple instances of a getsorted operator with a reference for accessing the sort-table directly. This avoids copying and propagating many rows via exchange operators, improving overall query execution performance.

To benefit from these performance enhancements delivered in SAP ASE 16, the parallel query must include a *sort* operator that appears on top of the *exchange* operator and the *exchange* operator must be a replicated *exchange*, with a single producer and multiple consumers. If these conditions are not met, SAP ASE 16 will continue to use the *sort* operator as implemented in previous releases.

Hash Join Operator Performance Improvement

SAP ASE 16 improves performance and reduces resource utilization for parallel query plans involving *hash join* operators. To achieve these results, SAP ASE 16 replaces the *hash join* operator with the *hash probe* and the *hash build* operators, and requires the utilization of a replicated *exchange* operator between these two.

The *hash build* operator builds a single hash table containing columns of the outer stream of rows that need to be joined. The *hash probe* operator finds matching rows by reading the inner stream and probing its values against the hash table produced by the *hash build* operator.

This improvement eliminates the need for multiple worker threads to build their own hash table, reducing memory consumption and the utilization of system resources. The end result is an overall query plan execution performance improvement.

To use the *hash join* improvements delivered in SAP ASE 16, a query plan must include a replicated *exchange* operator in between the *hash build* and the *hash probe* operators to allow for parallel execution. If these conditions are not met, SAP ASE 16 will continue to use the *hash join* operator as in previous releases.

Query Plan Optimization with Star Joins

SAP ASE query optimizer automatically uses star joins and considers special nested loop joins, when appropriate. SAP ASE 16 enhances the star join detection algorithm and introduces new star join execution plans, improving performance significantly. A star join is a commonly used data warehouse query that runs against a star schema database, which consists of a large table, the fact table, surrounded by dimension tables. Star joins *join* the fact table with one or more dimension tables along the foreign keys and may include filter predicates on the dimension tables.

In SAP ASE 16, the new star join detection algorithm uses *use fact_table* hint, which specifies the central fact table in a star join query. When a star join is detected, the hint triggers special query plan optimization strategies for the query. If the query processor cannot locate a good plan for the nested loop join while it generates the query plan for the star join (typically due to missing indexes or less restrictive selectivities), it chooses a nested hash join plan with a forced join order. The central fact table is placed at the innermost probing side and the placement of the dimension tables is based on the selectivities between the dimension tables and the fact tables.

The nested hash join plans enable the query processor to push multiple join bloom filters from the dimension tables to the central fact table scan. Each bloom filter probing effectively filters out rows that do not need to be joined from the fact table scan. This significantly reduces the number of qualifying fact table rows to be processed, enabling faster joins between dimensions and fact tables, and improving system efficiency. Additionally, nested hash join plans take advantage of any native partitions on the central fact table to execute parallel hash joins for star join queries, further improving performance.

Dynamic Thread Assignment

The utilization of static thread allocation in the execution of query plans may result in a suboptimal use of computation resources and reduce the ability to support extreme transactional workloads. On the contrary, the use of dynamic thread allocation allows databases to parallelize execution of query fragments, improve computing resources utilization, and achieve better scalability and performance.

In previous versions of SAP ASE, parallel queries could only use the number of threads that were available at query optimization time. Once query execution started, even if the particular query execution phase could only proceed serially with a single thread, the other worker threads would be reserved for use by the current process – even though idle. SAP ASE 16 uses dynamic thread allocation to execute parallel query plans faster and with fewer resources. With dynamically allocated threads, SAP ASE 16 uses only the number of treads that are needed to execute a given task, so that more computing resources can be available for other jobs. Additionally, it can load balance worker threads accelerating query plan execution.

This new feature of SAP ASE 16 is especially useful to accelerate the execution of queries that use joins across multiple table partitions. Specifically, it allows a single worker thread to join the first partition of one outer table to the first partition of one inner table and then continue the execution of the query plan fragment to join the second partition, and so on.

DBAs can activate dynamic thread assignment by configuring the *number of worker processes* and *max parallel degree* parameters.

Load Performance Enhancements for SAP Business Warehouse*

While growing needs for up-to-the-minute business insights impose shorter time-windows for data movement operations, expanding data volumes make massive data loading and merging operations increasingly common in data warehouse environments.

SAP ASE 16 implements two major optimizations to improve data loading performance in SAP Business Warehouse environments. First, it improves the performance of data insertion and index maintenance for *merge into* commands, leveraging both *bulk inserts* and *parallel index updates*. Second, it automatically triggers the use of the *bulk insert* mode at runtime based on insert/load performance statistics.

With these two enhancements, SAP ASE 16 effectively shortens time windows for large data loads into Business Warehouse and simplifies administration.

* Feature will be included in subsequent release later in 2014.

Scale Enhancements

SAP ASE 16 delivers unprecedented scalability with an optimal utilization of available computing and memory resources in large SMP servers. As proven by a recent internal OLTP benchmark on 80 CPU cores, SAP ASE 16 scales linearly with a throughput of over one million concurrent transactions per minute.

To achieve these results, SAP ASE 16 optimizes concurrency management with a more effective and granular use of locking and latching. It also expands parallelization in query plan execution and reduces resource contention with an optimal management of shared internal resources. These enhancements deliver the scalability required to serve even the most extreme transactional processing needs of the enterprise.

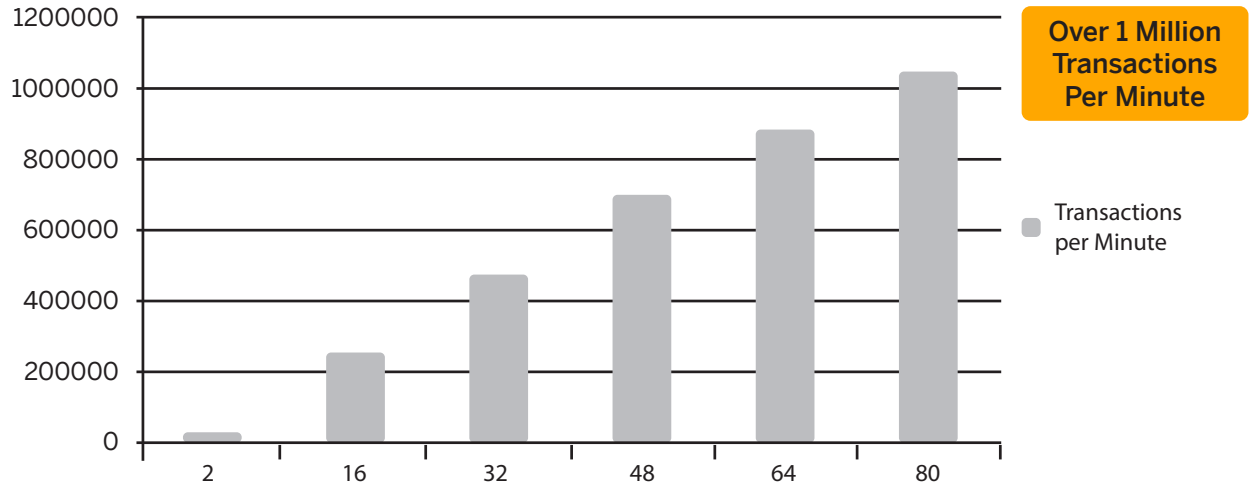

Figure 1: Internal OLTP benchmark

Run-time Logging Enhancements

Persisting concurrent transactions that operate on the same data under extreme workloads can significantly slowdown system performance. To deliver the necessary scalability, database management systems must parallelize their operations and manage contention of shared resources effectively, while maintaining the durability and integrity of their data.

As an example, SAP ASE 16 eliminates application contention and blocking with a more efficient management of access to the database transaction log. Applications with high concurrency often require data-rows locking to relieve lock contention when different users want to modify different rows of data on the same page. To ensure recoverability, previous versions of SAP ASE were forced to flush the user log cache of the first user. This premature flushing of the log cache sometimes resulted in contention for the log semaphore – so while data rows locking eliminated most of the contention by eliminating logical lock contention, there still remained some contention at the transaction log. As the number of concurrent users increased, this contention increased as well – often times becoming a bottleneck to scalability of write intensive applications.

To remediate this problem, SAP ASE 16 implements an intermediate user log cache queue, which holds the log records that should otherwise be immediately written to the transaction log when multiple users are modifying the same page at the same time. This reduces contention, allowing SAP ASE to support a larger number of concurrent users in write intensive applications.

Metadata Cache Enhancements

Internal data structures can become a serious bottleneck when a data management system operates under extreme transactional workloads. This is especially true of structures that manage contention with locks, latches or semaphores, which might unnecessarily waste CPU cycles.

To effectively leverage the computation power of large SMP servers and improve scalability, SAP ASE 16 eliminates CPU-spinning when processes access its metadata cache. The metadata cache is an internal SAP ASE structure that manages items such as object descriptors, index descriptors and similar entities. Virtually any process that accesses a database table has to refer to the metadata cache.

Each descriptor in the metadata cache uses considerable memory, and in previous releases of SAP ASE, configurations often specified a metadata cache smaller than the total number of objects and indexes. Due to high user concurrency, this cache would be exceeded causing SAP ASE to scavenge descriptors for reuse. To prevent scavenging object descriptors already in use, each process handling inflight queries had to update the descriptor's in-use counts. With high volume workloads, accessing this structure generated high contention levels and slowed down system operations. To protect frequently accessed tables, DBAs often had to bind a descriptor for dedicated use by a particular object at database startup time. Identifying these commonly used objects was a manually intensive and faulty process.

SAP ASE 16 prevents this potential bottleneck in two ways. First, it automatically maintains metadata objects of actively accessed entities in main memory and second, it uses atomic machine code operations to increment use counts. This reduces resource contention across the entire system resulting in overall scalability improvements.

Locking and Latching Enhancements

Often database management systems cannot scale linearly because internal locking and latching mechanisms prevent them from using computation resources effectively and delivering the appropriate level of transaction concurrency. The requirements of extreme transactional processing exacerbate these problems and might require a more granular and optimized management of both database locks and latches. SAP ASE 16 implements a number of new innovations that increase the granularity of locks and latches and optimizes their implementation, improving concurrency and reducing contention. Furthermore, these enhancements are automatically enabled and all clients who upgrade to SAP ASE 16 immediately experience the resulting scalability and performance gains.

Lock Management Enhancements

SAP ASE 16 includes a number of optimizations in the way it manages database locks to avoid spinlock contention scenarios. The external manifestation of these enhancements is better scalability and performance in heavy workload environments.

Specifically, SAP ASE 16 improves engine lock transfer, allowing the system to transfer blocks of locks, instead of individual locks, between global pools and engine local caches. Additionally, it optimizes the number of locks an engine can cache locally by increasing the default engine local cache size. In this way, lock reclamation and usage become more efficient and performance improves. Furthermore, a redesigned *lock verify* operation utilizes an optimistic approach that avoids unnecessary lock acquisitions, improving concurrency.

Additional enhancements in SAP ASE 16 include improved handling of lock promotion by avoiding to make a lock promotion request for each new lock request when previous lock promotion attempt failed. For Deadlock checking, a non-zero value configured for deadlock checking period now always uses a dedicated thread for deadlock detection allowing performance characteristics of DML statements to match the expectation. Finally, DOL tables can now be designated as hot tables, disabling escalation and reducing table lock contention. The combination of these optimizations allows SAP ASE 16 to further improve scalability and performance.

Latch Management Enhancements

SAP ASE 16 decreases CPU utilization for latch conflicts in environments that have very high transaction rates. To achieve these results, it decreases contention with a number of enhancements.

For example, it reduces contention on locks, latches, and data caches that result from cross-database references during periods of very high transaction volumes. Contention is also reduced on tables with a large number of partitions and on internal structures when the systems experiences high rates of create and drop tables. If under extreme transactional workloads, concurrent operations on a single page still cause data cache spinlock contention (perhaps because of hashing operations on *pageid*), SAP ASE 16 allows administrators to use a relaxed cache strategy, decrease the number of rows per page by setting *exp_row_size* parameter or use other techniques to reduce contention.

High workload scenarios can also increase contention on procedure cache, which can be caused by high concurrency of lightweight stored procedure (LWP) creation, query optimizations or sorting. SAP ASE 16 alleviates these issues by setting aside local procedure cache memory for each engine in the engine local cache (ELC). Because this memory is local, accessing it does not cause contention. SAP ASE 16 enables the use of ELC by default and it allocates 50% of the procedure cache for ELC, which is then spread across the engines uniformly. The *engine local cache percent* configuration parameter determines the size of the ELC and can be modified by administrators with the right credentials to meet the specific requirements of their workloads. The combination of these optimizations allows SAP ASE 16 to further enhance its scalability and performance.

Partition-level Locking

The granularity of the locking mechanism allows data management systems to support parallel execution of SQL statements on the same data sets and plays a crucial role in improving transactional concurrency. An optimal management of transaction concurrency is essential to effectively support extreme transactional workloads.

SAP ASE 16 expands the granularity of its locking mechanism with support for partition-level locking, improving its concurrency capabilities. This new feature allows large, high impact DML and DDL statements to execute on different partitions of the same table simultaneously. For example, a risk calculation application that uses tables partitioned by day could greatly benefit from this functionality. With partition level locking, DDLs such as *reorg rebuild* can work on one partitions, while DMLs such as insert or update can work on another partition concurrently.

DBAs use the *sp_chgattribute* command to enable partition-level locking, and the *sp_lock* command to view the partition locks currently held. Partition lock promotion is also supported to allow escalation of finer grain locks.

Relaxed Query Limits

With data volumes expanding at a faster pace than ever before, the number of data attributes to be considered in querying a database expands and query complexity significantly grows.

SAP ASE 16 addresses these needs not only by parallelizing the execution of complex queries, but also by relaxing previously existing query limits so that more data attributes can be correlated across large volumes of diverse data. For example, it increases the maximum number of tables on queries from 50 to 250 and increases the maximum number of columns allowed in order by clauses from 31 to 400. Additionally, in SAP ASE 16 the number of sub-queries supported within a query increases from 50 to 250. With these enhancements, SAP ASE 16 improves query design flexibility and enables users to extract valuable information from very large amount of diverse data, effectively.

Security Enhancements

Security enhancements in SAP ASE 16 address the needs of data privacy protection and system auditability, effectively and with minimal impact on system performance. SAP ASE 16 gives administrators more flexibility to choose how to protect their data to meet compliance and minimize business disruption. It also provides improved system auditing capabilities to facilitate the monitoring and tracing of actions performed on sensitive data, so that administrators and security personal can operate on actionable information, saving time and reducing auditing costs.

Full-Database Encryption

New security threats are created every day and new industry regulations can at any time require that new types of data be treated as sensitive and appropriately protected. Data encryption has become one of the preferred remediation techniques to protect sensitive data and it commonly complements database authentication, database access controls and perimeter security. To be effective, data encryption needs to provide the right degree of flexibility to adapt to specific compliance requirements and minimize interferences with business operations.

SAP ASE has been providing state of the art column-level data encryption for quite some time, and with SAP ASE 16, it adds support for full-database encryption. Both encrypted columns and fully encrypted databases can help companies meet their security and privacy requirements. One solution however, might be more suitable than the other, depending on specific circumstances.

Column-based encryption might be preferable when a company can easily identify what data is sensitive for their business and different database users have different access rights on the sensitive data. Its use however might have performance implications because in this scenario, data is not only kept encrypted on disk, but also in main memory. Additionally, column-level encryption requires that each sensitive data attribute be evaluated individually to validate the access rights of different users and to identify the encryption technique used on the attribute. Performance can especially be a concern when queries use common techniques such as range searches on the encrypted data field, as the encryption process will likely randomize the data such that it is no longer in alphabetical sequence. In these specific circumstances full-database encryption might perform better, especially if all database users have the same access rights (or a single database login is used). The use of full database encryption is also preferable when it is hard to identify which columns hold sensitive data.

In SAP ASE 16 full-database encryption automatically encrypts all database objects – including data files, indexes and transaction logs – and insulates existing applications from the entire encryption process because the data is automatically decrypted when it is moved from disk to memory, and encrypted when it is moved from memory to disk. Encryption is implemented at the page level and SAP ASE 16 uses a buffer manager to encrypt pages before they are written to disk and decrypt them when they are loaded into memory (Figure 2).

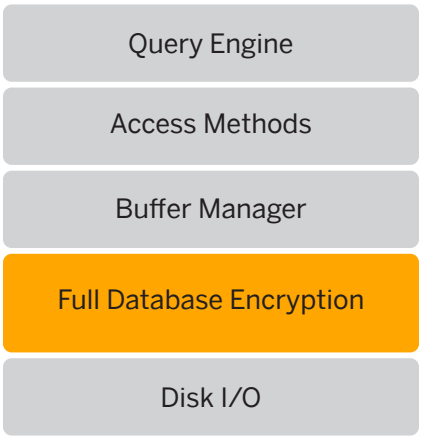

Figure 2 SAP ASE 16 Decryption/Encryption Mechanism for Full-database Encryption

To use full-database encryption is simple. First, DBAs create a key using the *create encryption key* command with a newly introduced syntax. SAP ASE uses 256-bit symmetric keys for full-database encryption. Second, once the key is created, DBAs can use it to encrypt a new or an existing database. To encrypt an existing database, DBAs use an *alter database* command. To encrypt a new database, it is enough to use the *create database* command with an updated syntax.

It is important to notice that SAP ASE 16 implements full-database encryption of existing databases as an online operation. This means that an *alter database* command that fully encrypts a database executes in parallel with other database workloads and doesn't interrupt either running queries or insert and update operations. Also, since fully encrypting a very large database can take a long time, the encryption process can be paused and resumed, giving DBAs the flexibility to perform critical system maintenance as needed.

SAP ASE 16 full-database encryption complements column-based encryption and provides a straightforward way to protect all data in the database without affecting existing applications or running workloads, and effectively delivering end-to-end advanced security.

Residual Data Removal

In most data management system operations that delete space, such as drop database, alter table, truncate table free a storage resource but do not physically erase the data written in it. This leaves an open door to security breaches because users with the right credentials might still gain access to the deleted data.

SAP ASE 16 addresses this issue with a new residual data removal capability, which automatically zeroes deleted data and doesn't leave any residual data fragment behind. This functionality operates at different levels of granularity – database, session or table level – simplifying administration and enhancing the overall database security.

Full-Text Auditing

Deploying technology that reliably tracks all actions performed on enterprise data has become a necessary best practice to comply with security regulations. This is also invaluable during auditing processes that follow a security violation, since companies need to assess the situation and take the appropriate remediation actions quickly to reduce potential damages.

SAP ASE 16 expands its auditing capability with the introduction of full-text auditing. Specifically, it records full-text audit information for the *select (into), insert, delete* and *update* commands. The table in Figure 3 provides examples of the granularity of this auditing capability, which displays parameter names and values for regular data, but masks encrypted values to comply with security best practices.

This new capability in SAP ASE 16, simplifies data auditing, reducing the time required to track actions performed on data. This empowers administrators and security personnel to expedite security auditing and make informed decision about remediation actions.

Audit event	Audit records before SAP ASE 16	Audit records in SAP ASE 16
insert with constants	sa_role sso_role oper_role sybase_ts_ role mon_role; INSERT; ; ; ; ; sa/ase;	sa_role sso_role oper_role sybase_ts_ role mon_role; insert mytab values(100, „audit"); ; ; ; ; sa/ase;
update with variables/parameters	sa_role sso_role oper_role sybase_ts_ role mon_role; UPDATE; ; ; ; ; sa/ase;	sa_role sso_role oper_role sybase_ts_ role mon_role; update mytab set c1 = @ var3 where c1 = @var1 and c2 like @var2; ; ; @var3 = 500, @var1 = 100, @var2 = audit; ; sa/ase;
Insert with encrypted column. Encrypted columns are obfuscated in audit records.	sa_role sso_role oper_role sybase_ts_ role mon_role; INSERT; ; ; ; ; sa/ase;	sa_role sso_role oper_role sybase_ts_ role mon_role; insert mytab1 values(@ var1, @var2); ; ; @var1 = ****** , @var2 = audit; ; sa/ase;

Figure 3 Comparison of audit results in SAP ASE 16 and in previous releases

Simplicity Enhancements

Among the various cost factors associated with the deployment and operations of a database environment, IT staffing (cost for RDBMS admin, server support, help desk support for database, and training) is the highest, accounting for 75% of a database TCO[8]. Simplifying and automating management of demanding data management environments is extremely important for enterprises that have to cope with expanding data sizes, transaction throughputs and user populations.

SAP ASE 16 delivers simplified database administration with analytic and intuitive graphical tools that reduce the complexity of setting up and managing large SAP ASE environments. It also adds a wealth of enhancements that simplify the way SAP ASE interacts with other applications. With SAP ASE 16, business applications can transparently access data in SAP HANA and implement data manipulations on SAP ASE data tables without requiring code modification or complex system setups. This lowers the level of expertise and shortens the time required to manage SAP ASE deployments, reducing their overall operational costs.

[8] IDC, Calculating the True Cost of RDBMS Ownership, 2011

Additionally, to minimize the cost of business disruption and simplify the recovery from disasters, SAP ASE 16 works with SAP Replication Server to deliver higher availability and shorter recovery windows to users of SAP Business Suite, ensuring that transactions are always durable and data is accessible within minimal delays. In this way enterprises can minimize business disruption and experience "zero data loss", even when one of their data centers is brought down in a disaster.

Multi-Trigger Support

Many companies rely on packaged and homegrown applications for mission critical business processes, but to address specific business requirements, these applications often need to be modified. This can present a major challenge for packaged applications since their code typically cannot be altered without either running into supportability issues or impeding further upgrades. It is also costly for IT departments to constantly modify homegrown applications for ever-changing business requirements. Database management systems can alleviate these problems by providing mechanisms that affect the way applications manipulate data, without altering their code.

Multiple-trigger support, a new feature in SAP ASE 16, allows up to 50 triggers to be fired for a single DML statement, causing sequences of data operations to automatically execute across database tables. This makes it simpler for application developers and DBAs to implement custom data manipulation without modifying an application code.

Developers can easily specify which actions have to be executed and in which order by using a *create trigger* or *replace trigger* statement. These statements also indicate whether the triggering event is an *insert, update* or *delete*. Additionally, the multiple-trigger support in SAP ASE 16 allows developers acting on the same data to easily create new triggers, independently and in a flexible way, since the creation of a new trigger does not drop existing triggers. Each trigger is executed sequentially in the established order.

From a data integrity point of view in case of failure, a rollback transaction in a trigger automatically causes rollback of the insert, update or delete statement that fired that trigger. For multiple triggers, a rollback transaction from one trigger also rolls back the work of other triggers already fired, preserving the integrity of the DML operations. Additionally, to maintain control of the data, each given table owner can disable any or all of the triggers associated with data in the table.

Integrated Disaster and Recovery Support with Synchronous Replication for SAP Business Suite*

Continuous access to data even in case of a catastrophic disaster is a requirement for companies that use SAP Business Suite to run their most mission-critical business processes.

SAP ASE 16, working in conjunction with SAP Replication Server 15.7 SP200, extends high availability (HA) and disaster recovery (DR) support for SAP Business Suite applications with a number of new enhancements. First of all, SAP Business Suite applications running on SAP ASE 16 can now experience "zero data loss" even when the host or the data center is physically damaged. Using synchronous replication, all transactions are effectively written to both the SAP ASE transaction log and to the Replication Server Stable Queue, which is a persistent store, before being committed, and then propagated to remote sites. When a failure occurs, a new DR Agent manages the complete failover/switching of active systems. Client connections are transparently reconnected to the new primary system as soon as it is available.

If the production data center is destroyed during a disaster, the recovery site can be used without data loss, as it has all the recent transactions executed at the primary site in the SAP Replication Server Stable Queue. Once the SAP Replication Server Stable Queue is drained, the remote site will have the exact copy of the data as the primary site. There are three modes of synchronous replication: Sync, Near-Sync and Async. Administrator can employ these sync modes based on Service Level Agreements requirements. Additionally, synchronization of the standby database can be individually configured to optimize system performance and mean time to repair. This mechanism can be leveraged to deliver higher system availability and shorter DR windows.

SAP ASE to SAP HANA Data Access

The need to access data independently of its physical location is pervasive across the enterprise, but with exponential growth in data volumes, it is often necessary to process data in multiple databases. A common example is when an OLTP system needs to create a report by accessing data which is kept in multiple systems.

SAP ASE enables users to access data residing in SAP ASE and in other supported databases transparently using the federation capability implemented by the Component Integration Services (CIS), simplifying data access and reducing query latency. SAP ASE 16 expands its federation capabilities to include support for SAP HANA. For this purpose, CIS has been extended with a new native ODBC interface, which coexists with the proprietary CTLIB interface. Both these interfaces can be used simultaneously to communicate with supported backend databases.

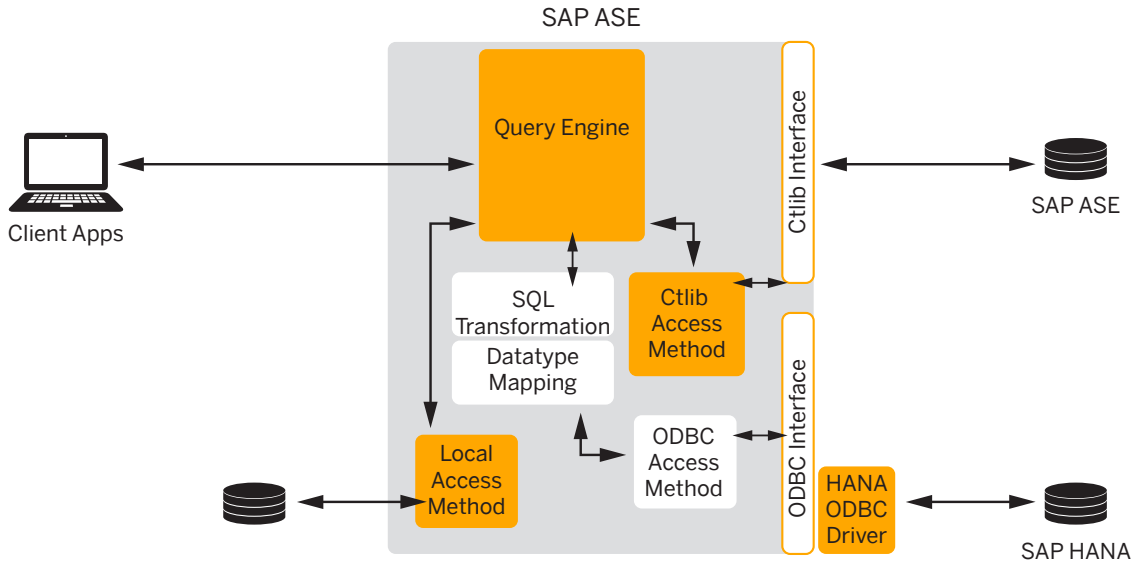

Figure 4 SAP ASE 16 data federation access to SAP HANA

To interoperate with SAP HANA, SAP ASE 16 creates local tables (referred as proxy tables), which do not actually contain data, but point to corresponding SAP HANA tables where the data resides. When a query in SAP ASE refers to data in SAP ASE tables and SAP ASE proxy tables, the relevant portion of the query on the proxy tables is pushed to the remote SAP HANA server and executed there. The results of this remote query are then brought back to the SAP ASE server and used as appropriate.

To facilitate the interoperability with SAP HANA, SAP ASE 16 implements a new server class, HANAODBC, that facilitates mapping of SAP HANA data types to SAP ASE data types. To communicate with SAP HANA, the SAP HANA client package, which includes the ODBC drivers, must run on the same server as SAP ASE 16.

The expansion of SAP ASE federation capabilities to support SAP HANA eliminates the need for time consuming data movement and makes data access transparent and faster, improving real-time decision-making and reducing operational costs.

Support "create or replace" command for database objects

Database administrators often need to replace existing compiled object definitions with new ones, while preserving original attributes such as name, object ID, auditing options, permissions or replication attributes. These tasks can be tedious, time consuming and error prone.

SAP ASE 16 addresses this issue with a "create or replace" functionality, which creates a new object if it doesn't exist or replaces an existing object with a new one, with the same name. DBAs can use this functionality by adding an *or replace* clause to the syntax of the appropriate *create* statement. When an *or replace* clause is used, the system automatically drops the original object and recreate a new one with the same name, object id, security settings and replication attributes as the original one. This saves time for administrators and avoids potentially expensive mistakes.

Enhanced SAP Control Center Support

IT departments have the imperative to reduce overall operational costs by automating and simplifying their operations. SAP ASE 16 simplifies system administration and lowers operational costs through a tighter integration with SAP Control Center (SCC). SCC significantly expands its support for SAP ASE with a combination of analytic utilities, which help DBAs make informed tuning decisions, and task automations, which reduce time and eliminate error prone activities.

Compression Advisor Utility

With SAP ASE 16, DBAs can leverage the compression advisor utility of SCC to evaluate the benefits of using different compression scenarios and identify specific database objects that would benefit from compression.

SCC compression advisor estimates the percentage of space that can be saved by compressing tables and indexes. Using SCC to examine the database usage properties, DBAs can easily identify tables that may benefit from compression. Once the candidates for compression are identified, DBAs select either a row-level compression or page level compression scenario, and optionally specify compression options for each column and index in the table. They then instruct the compression advisor to estimate the impact of compression based on these options. Estimates are based on analysis of a sample of data in the entire candidate table that is loaded in a temporary table for use during the estimation. Figure 5 summarizes the simple steps required to start the compression estimation.

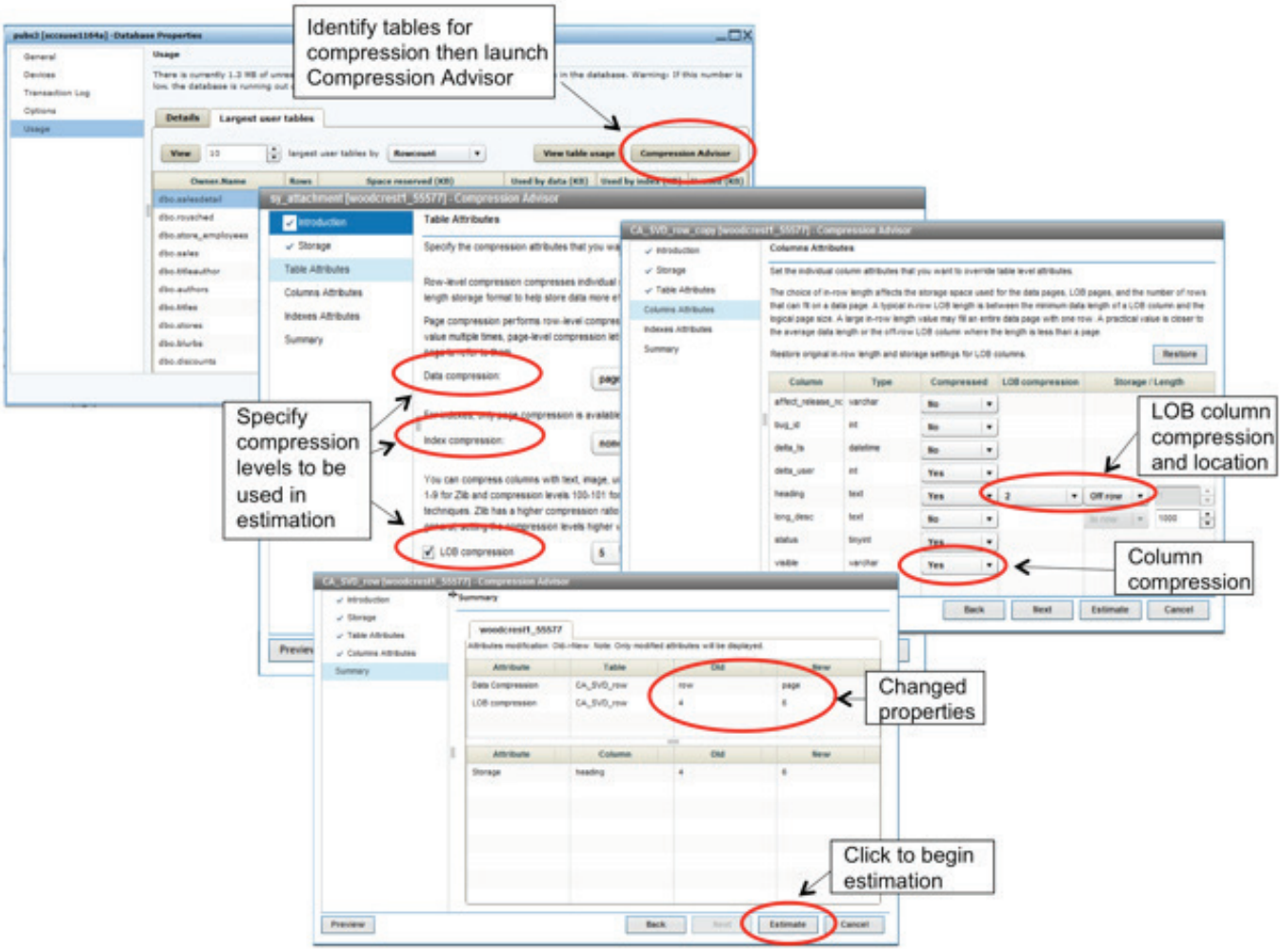

Figure 5 SCC Compression Advisor Utility

With compression advisor, administrators can make informed decisions about compression options and ensure that their operations benefit from data compression through reduced database storage and improved system performance.

Automatic Backups

With SAP ASE 16 administrators can use the SCC backup scheduling and task management feature to automatically schedule recurring backups or perform a one-time backup of both databases and transaction logs.

Specifically, this feature supports database full backup, transaction backup, and cumulative backup. DBAs can schedule an ongoing backup operation with the *backup database wizard* by simply clicking the 'Schedule' button, which activates the *add scheduled job wizard*, as shown in Figure 6.

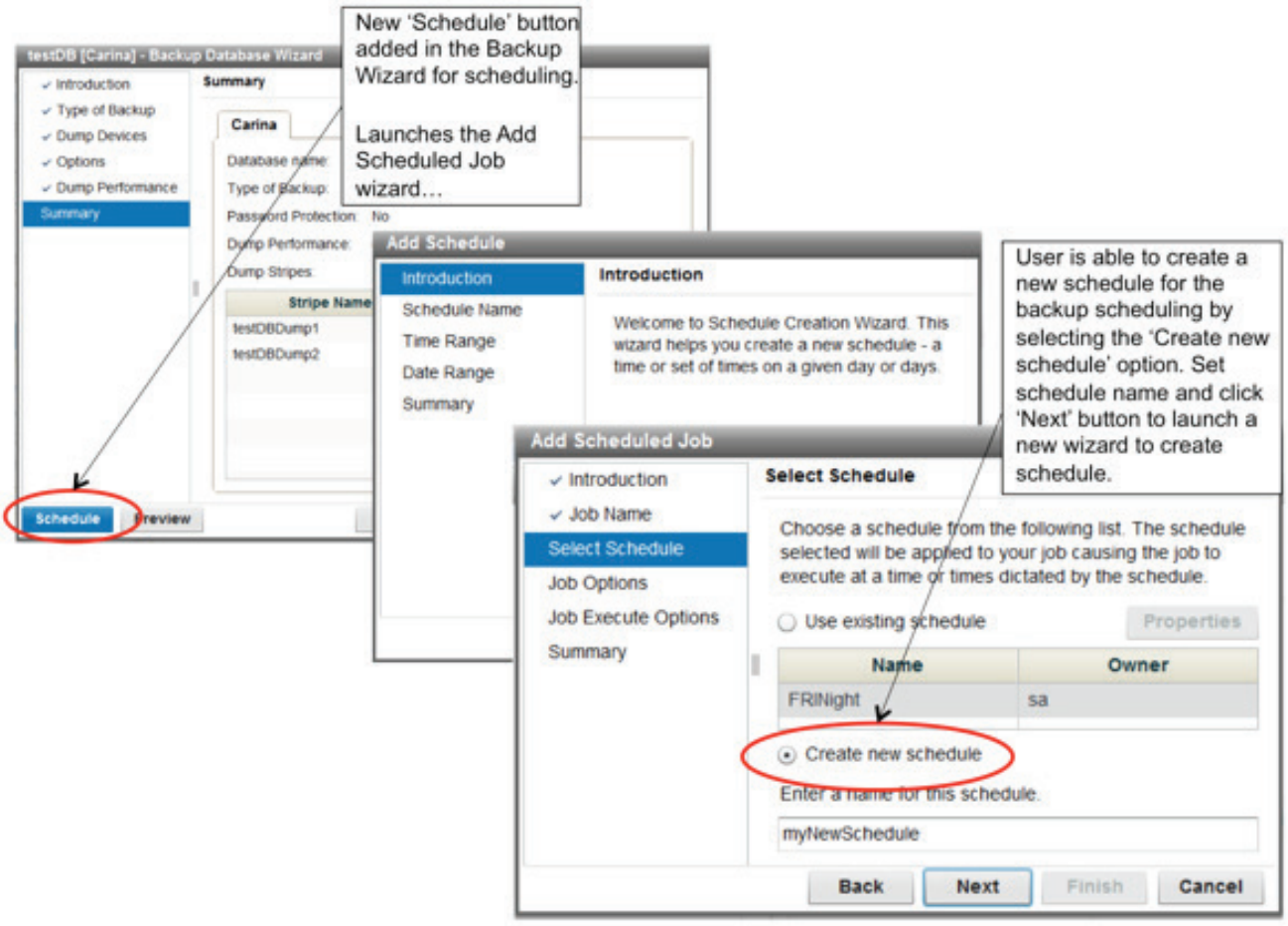

Figure 6 Scheduling a Backup with SCC Backup Database Wizard

Alternatively, DBAs can schedule a backup from the *Task Management Folder* of the SCC administration console, which activates the *add scheduled job wizard*, and then provides either a *backup* command or a *JS Template* as the input.

From the SCC admin console, administrators can than monitor and control the status of backup jobs and access historic backup information, improving productivity and lowering administration costs.

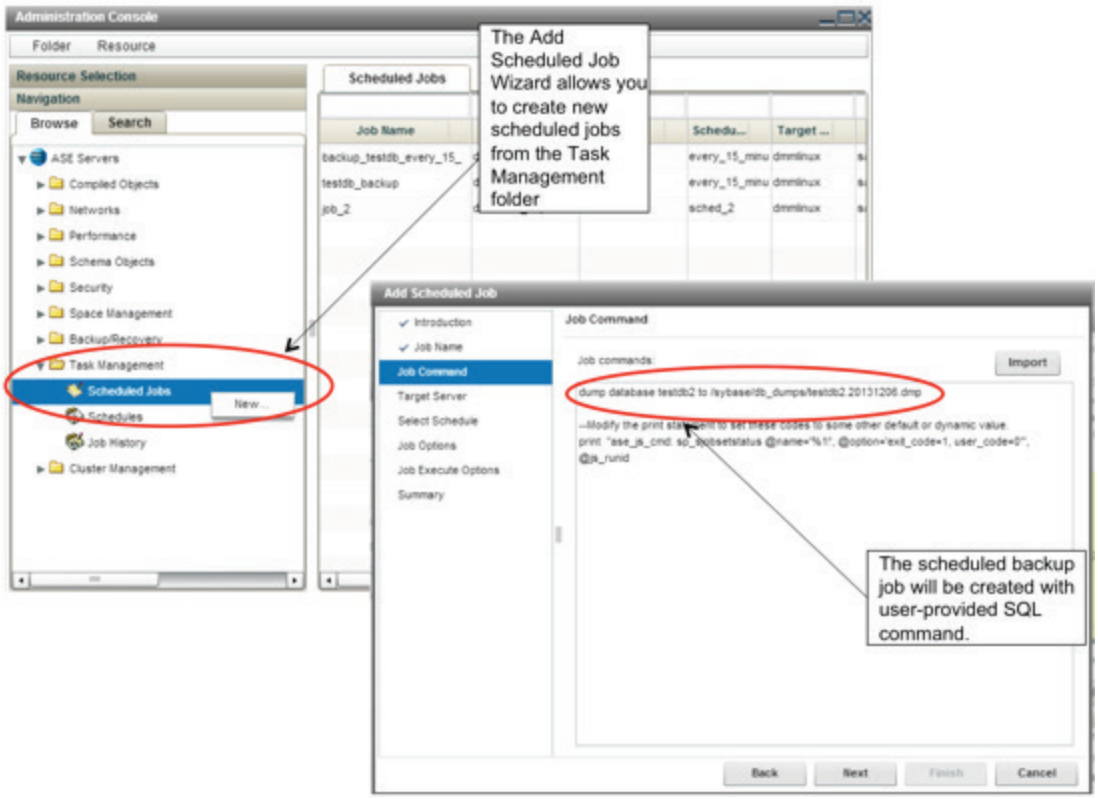

Figure 7 Scheduling a Backup with Add Scheduled Job Wizard and a SQL Command

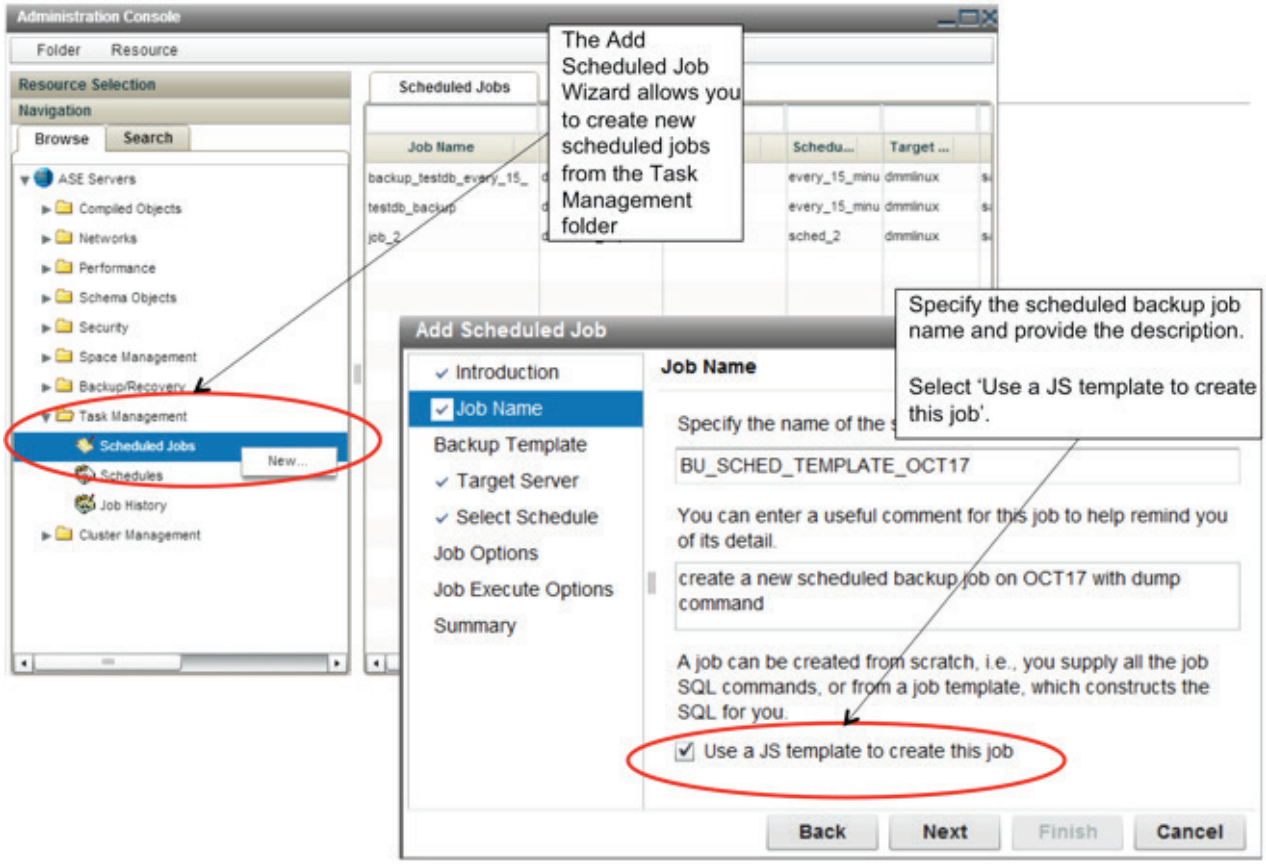

Figure 8 Scheduling a Backup with Add Scheduling a Backup Job Wizard and a JS Template

Additional SAP ASE 16 Support in SCC

In addition to the compression advisor utility and the automatic backup scheduling capability, SCC adds the following to support SAP ASE 16 features:

Feature	Support Provided in SCC
Full database encryption	Manage database encryption for new and existing databases, and key management encryption.
Create or replace command	Provide the Create wizard to implement these commands.
Partition-level locking	Enable and disable partition locking and provide 'online' option for move, merge, split, delete partition operations.
Index compression	Enhance Create Table and Create Index wizards to set index compression for table/ index/local index partition level.
Error log monitoring enhancements	Enhance Error Log monitoring support to import/ export error flags to multiple servers for ease of configuration and use.
Server configuration alert	Allows users to configure alerts to fire when the percent utilization of any of the monitored configurable resources (sp_monitorconfig) exceeds a configured threshold.
Management of Large scale deployment	Facilitate the management of large-scale SAP ASE deployments supporting up to 250 managed resources.
Secure Store	Allow removal of residual data when a database, table or index is dropped
Data Cache Spinlock Contention Monitoring	Enhance Monitoring View for SAP ASE with metrics for data cache spinlock contention.

Additional Manageability Improvements

SAP ASE 16 implements a number of additional manageability improvement to simplify the administration and reduce the cost of managing SAP ASE deployments. The following table lists examples of these changes.

Feature	Description
Configuration History Tracking	SAP ASE 16 allows the tracking of configuration changes made to a given server. For each change, it captures information such as the configuration option that has been changed, the old and new values of the option, the identity of the user who made the change, and when the change was made. SAP ASE 16 tracks a large number of configuration properties, including server-wide configuration parameters, database options, data cache and data pool properties, engine threads, and changes to the server configuration file. The history of configuration changes data is stored in the sybsecurity database. To view the configuration changes, administrators can use reporting features of the *sp_spconfighistory* , which presents the history of configuration changes in an easy to read format. Alternatively, they can access the configuration history data directly using the *ch_events* view that is part of the *sybsecurity* database.
Cyclic Redundancy Checks for Dump Databases	SAP ASE 16 adds a cyclic redundancy check for accidental changes during database backup operations. For example, these changes can happen to raw data as a result of either database or transaction dumps created with compression. This functionality is used to verify that the compression blocks can correctly be read and decompressed. DBAs can activate this feature by adding the *verify*= parameter to the *dump database* command.
Calculating Transaction Log Growth Rate	SAP ASE 16 adds the ability to calculate the transaction log growth rate for a specified time period. The *sp_logging_rate* command displays the minimum, maximum and average rate of the transaction log growth in gigabytes per hour. Using this information, DBAs can predict future log extensions that might be necessary and provide input for data movement interfaces, such as data replication or EDI tools.
Monitoring Threshold-based Events	SAP ASE 16 allows administrators to configure, record and list SAP ASE Resource Governor threshold violation events. This new feature uses the *monTresholdEvent* table to record and report the thresholds for events and records all violations of configured resource limits. Examples of events monitored include the *tempdb* usage per transaction, the total runtime of a query or transaction, the number of rows fetched or written by a query, the number of rows read by a query, the total CPU time of a query, and the estimated plan cost of a query. To record events to the *monTresholdEvent* table, administrators set the report action for resource limits using the *sp_add_resource_limit* procedure. Different actions can be configured for different resource limits..
Statistics Stickiness Enhancements*	In previous releases of SAP ASE certain update statistics options ware attached to a column once they were executed for the first time as their 'stickiness' bits were automatically set in the catalog. A new feature in an upcoming release of SAP ASE 16, will provide a way to report and unset this stickiness behavior.

Utility Enhancements

To simplify administration SAP ASE 16 enhances the capabilities of a number of exiting utilities. The following table lists a selection of these changes.

Utility	Enhancement Description
ddlgen	SAP ASE 16 enhances the *ddlgen* utility with the ability to create DDL of fully encrypted databases and database encryption keys. For creating a DDL of an encrypted database, DBAs uses *ddlgen* and uses the extended type *-XODE* with the *TEK* parameter to generate the necessary DDL.
sybmigrate	SAP ASE 16 enhances the *sybmigrate* utility with the ability to migrate fully encrypted databases. To encrypt the target database with the same encryption key as the original one, DBAs have to first use the *ddlgen* utility to generate the DDL for the master key or dual master key, and the database encryption key of the original database. Then they execute the resulting DDL on the target server to generate a database encryption key with the same raw-key value as the one of the original database.
sybrestore	SAP ASE 16 enhances the *sybrestore* utility for restoring an SAP ASE server after master database corruption in a fast, automatic, reliable and comfortable way. The restore utility, using *srvbuild* on Unix and Linux platform or *sybatch* on Windows platform, executes all the required process steps, from rebuilding an SAP ASE instance to restoring all user databases.
dbisql	In SAP ASE 16 *dbsql* delivers an improved Excel import/export capability through a generic ODBC plug-in and offers more options to control results and message output. Additionally, it offers an improved Connect Dialog, which can now encrypt password by default, start a discovered/down SAP ASE, and connect to the last-used database, instead of default database.

CONCLUSION

SAP ASE 16 has been designed from the ground up to deliver the speed, scalability, security and simplicity enterprises need to meet the challenges of extreme transactional workloads, efficiently and economically. Internal benchmarks show linear scalability in large SMPs, delivering the transaction throughput and response time required to support the most demanding transactional workloads. Additionally, it provides the security and data reliability essential to minimize business disruption and protect the privacy of enterprise data effectively. Finally, it lowers the TCO of large deployments with a wealth of analytical and graphical tools that simplify the management and lower the expertise required to manage even the most demanding SAP ASE environments.

WWW.SAP.COM