



THE BEST-RUN E-BUSINESSES RUN SAP 



Volume Testing for SAP Solutions

– Generic Procedure

Best Practice for Solution Management

Version Date: November 2006

The newest version of this Best Practice can always be obtained through the SAP Solution Manager or the SAP Service Marketplace.

Contents

Applicability, Goals, and Requirements	2
Best Practice Procedure and Verification	4
Preliminary Tasks	4
Procedure – Roadmap for Volume Tests.....	5
Plan volume test project.....	5
Define Test Scenarios and Requirements.....	7
Implement Test Environment.....	13
Implement Test Tools.....	15
Run Test and Evaluate	17
Further Information	22
APPENDIX.....	23
Monitoring SAP Application Servers.....	23
Monitoring Interfaces.....	24
Types of Interfaces.....	24
Key Performance Indicators (KPI's).....	25
Monitoring.....	26

Applicability, Goals, and Requirements

To ensure that this Best Practice is the one you need, consider the following goals and requirements.

Goals of Using this Service

The goal of volume testing is to run a realistic simulation of expected workload during the implementation period of an IT solution. This simulation tests the solution's performance, sizing and scalability. Specifically, it focuses on runtimes and data throughput for critical business process paths and interacting components.

This Best Practice describes, independently of the application subject to volume testing, a generic procedure how to plan, execute, and evaluate volume tests.

You can apply this methodology not just to the complete solution, but also to specific components (interfaces, SCM liveCache size,...) of your solution. For the various products of SAP, additional specific aspects have to be taken into account for volume testing. For these details we refer you to the Best Practices for volume testing with focus on particular products which are available on the SAP Service Marketplace (<http://service.sap.com/VTO>) --> [Media Library](#).

Volume tests are performed to answer the following kinds of questions:

- Does the implementation of business processes meet your company's specific performance requirements for those processes?
- Will the implemented solution meet the company's specific performance requirements with respect to response time and throughput?
- Will the implemented solution be able to handle the maximum expected volume within the available time frame, thereby achieving the required throughput?
- Does interactive processing comply with the maximum allowed response times?
- Was the hardware sizing accurate, so that the planned hardware of the production system will be sufficient?
- Is the implemented solution scalable?
- Will there be resource-contention problems during production operation and what will be their performance impact?

In advance of productive operation, it is often difficult to determine whether the planned hardware will be sufficient to handle the planned workload. This is especially true for highly resource-consuming (or "expensive") customizing settings, user exits, or customer enhancements. Such elements usually cannot be taken into consideration in advance by sizing tools such as the SAP QuickSizer.

Scalability can be endangered by the contention for central resources due to concurrent processing activities, such as when there are exclusive lock waits. During the volume test, activities that will be running concurrently in production operation are scheduled to run concurrently in order to identify negative interactions between them.

Alternative Offering

SAP experts deliver this Best Practice in the framework of the Solution Management Optimization (SMO) service, known as the SAP Volume Test Optimization. This service covers three stages: review of the volume test plan, monitoring of the volume test and reporting of the test results. For further details concerning this service, refer to SAP Service Marketplace (<http://service.sap.com/vto>).

Staff and Skills Requirements

A volume test has to be set up in the framework of a subproject within your (SAP) implementation / upgrade project. For that purpose, you need a designated volume test project team with appropriate knowledge regarding technology and applications. For details, refer to the procedure described below.

System Requirements

The test environment must have comparable hardware to the productive hardware solution. Ideally the test should either be performed directly with the planned productive system configuration or with a suitable copy.

If you are unable to use the system configuration that will be used in your production landscape, ensure that the application parameters are set similarly and that the hardware size has at least been confirmed by a sizing estimation. This measure is also applied for the data volume. The data load (at least for the data that is directly used for the test scenario) should be brought close enough to the estimate data load on productive environment to get the most realistic condition. For further details, see the procedure described below.

Duration and Timing

The duration of the volume test strongly depends on the complexity of the tested business scenarios. It should be performed several months prior to the start of production operations and prior to introducing major changes to a productive system environment. For details, refer to the procedure described below.

How to Use this Best Practice

Read the whole document prior to the project start. Then, use it to guide you through the different steps of the project. This Best Practice document is divided into three parts:

- Preliminary activities:
This section describes key requirements, which have to be fulfilled before the volume tests can be started.
- Roadmap for volume testing:
This section describes how to plan, execute, and monitor the volume test and to evaluate its results.

The success of each phase of a volume test project depends on distinct success factors. Throughout the document, these factors are emphasized in particular.

Important steps in this document are illustrated with a hypothetical company performing a volume test according to SAP recommendations. The example is presented in boxes such as this one.

Best Practice Procedure and Verification

The Best Practice procedure described in the following applies to virtually all possible volume test scenarios. For the sake of simplicity, to illustrate the basic procedure we have chosen a simple R/3 test scenario.

For the various SAP products, which may be subject to volume test scenarios, there are additional aspects, which have to be considered when you plan and execute a volume test. For a discussion of these product-specific details, we refer you to the Best Practices for application-specific volume testing. These Best Practices are based on the *generic* Best Practice you are reading at the moment.

Preliminary Tasks

Performing volume tests follows a bottom-up approach that proceeds through the following stages:

- Performing analysis and measurement in single user mode
- Using a *forecast* to delimit the scope of volume testing
- Performing volume testing

Following this approach saves you time and money because it is less expensive to optimize procedures at an early stage, for example on the basis of an analysis of a business process step, use single user mode, rather than continuously repeating unsuccessful volume tests.

Based on the expected workload and the response times measured in single user mode, each business process step is forecasted regarding the question whether it may become time-critical, i.e., whether it will not be possible to handle the expected workload within the available time-windows. This forecast is then used to delimit the scope of the volume test so that the latter only focuses on time-critical processing steps.

In advance of a volume test, the following preliminary tasks have to be performed:

Success Factor: Development and Customizing Freeze

Before you start the final test activities, it must be ensured that all development activities have been finished including the customizing. The only exception refers to inevitable emergency corrections.

If you continue with development during the testing phase, already successfully finished tests for parts of the business process may become meaningless. In this case, later during productive operation you may encounter severe problems for already tested scenarios where critical parameters have been changed after the test runs and a retest never has been done.

Success Factor: Functional Correctness

Perform an integration test to verify the functional correctness of business process implementation. The integration tests have to precede the volume test to ensure that the volume test truly reflects the processing in subsequent production operation and prevent the volume test failing due to functional errors. The integration test requires that development and customizing activities are already completed and that the versions of all software components are frozen until the start of production.

Success Factor: Performance Analysis and Optimization in Single User Mode

Analyze the business process steps in single user mode for expensive database accesses or RFC calls, unnecessarily long database lock times or R/3 enqueue times, as well as for the consumption of database time, CPU time and RAM. Remember that you need a representative data basis in order for the measured response times and resource consumption to be representative of subsequent production operation.

Procedure – Roadmap for Volume Tests

We recommend that you prepare and run your volume tests according to the following roadmap:

- Plan volume test project
 - Define test objectives
 - Establish test plan (high-level, based on objectives)
 - Setup project plan
- Define test scenarios and requirements
 - Establish test scenarios and load profile
 - Determine requirements for test landscape (sizing / configuration / test data)
- Implement test environment
 - Implement test landscape
 - Prepare data basis (master / transactional data)
 - Migrate test data to test landscape
- Implement test tools
 - Implement load generators
 - Implement / setup monitoring tools
- Run test and evaluate
 - Run volume test and monitor
 - Evaluate test results
 - If necessary: optimization and repetition of test runs

Plan volume test project

Define test objectives

Before starting a volume test project, you have to carefully consider whether:

- In your businesses the probability of volume-critical situations occurring
- The costs and efforts related to a volume test project are justified
 - For large projects / installations, related volume test projects can take several months and cause costs of several 100.000 \$ (and even significantly more)!

At the beginning of a volume test project, it has to be agreed by all parties involved (from both business and IT department) what the actual purpose of the volume test shall be:

- A „real-life“ simulation of volume-critical situations with the goal to get a statement regarding the feasibility of the planned productive operation
- A “smoke” test to identify the limits of data throughput with your current hardware
- A volume test for a subset of applications and/or interfaces, i.e., a test, which only covers a part of your business process.

Success Factor: Criteria for Successful Volume Test

Define clear, verifiable and quantified criteria whose fulfillment will indicate a successful volume test. These criteria must be defined before performing the volume test, and be in accordance with the actual business requirements.

This in turn means that the volume test team gets in touch with the business department already in the early planning phase for the volume test. Only with the secure knowledge regarding critical business volumes and required / available time-windows, you will be able to identify the volume-critical process steps. The success criteria have to be defined in accordance with the required business process flow. In the procedure described below, we discuss this aspect in detail.

Establish Test Plan

Once you have agreed on the objectives of the volume test and identified – still at a high-level – which business projects should be subject to volume testing (for the exact procedure, see further below in the Roadmap for Volume Testing), you have to establish a basic test plan. It should contain a basic schedule for the activities required for planning, executing, and evaluation of the volume test.

Setup Project Plan

The volume test should be planned, executed, and evaluated in the framework of a formal volume test project. The latter should be executed according to the general guidelines valid for project management.

Based on the high-level test plan, you have to set up the volume test project. In particular consider:

- Timing
- Available vs. necessary budget
- Project team (also including support from other parties, like your hardware partner, the SAP project implementation team, third-party companies for test tools, and so on..)

Duration and Timing

The duration of the volume test strongly depends on the complexity of the tested business scenarios. It should be performed several months prior to the start of production operations and prior to introducing major changes to a productive system environment such as:

- Implementing a new module or new functionality
- Adding new users
- Performing a major hardware or software upgrade

It is essential that the volume and preliminary tests are performed **under live conditions, at least four weeks before the live date**. "Live conditions" means using the same data volume as in the production system. Enough time should be left in case sizing changes are necessary and the processes have to be restarted.

When planning the timetable for testing, remember to plan the preliminary testing (that is required to ensure a proper execution of the volume test) sufficiently ahead of the volume test. This preliminary testing (see below) consists of:

- An integration test that tests the functional consistency of the customized business processes. For this, no representative data is needed and the test can normally be performed by a single user.
- Optimizing the performance of individual business steps, for example, using the GoingLive optimization session with representative data and a single user.

After these preliminary tests, the volume test can be performed as a multi user test with representative data.

Success Factor: Timing

Perform volume tests during the *go-live preparation* phase of an SAP implementation project. If you are using the SAP GoingLive Service, the volume test should take place after the GoingLive Optimization session since the focus of this session is the analysis and optimization of business process steps in single user mode. The GoingLive Analysis session, which precedes the Optimization session, ensures a reasonable initial system configuration.

Success Factor: Test Team

The test team should have skills in the areas of technology and application. At least one person of the test team should work full-time on the preparation of the volume test, focusing on planning and implementing the technical framework of the volume test.

To run the volume test, create it as a sub-project within the SAP implementation project by defining a test team with appropriate skill, (see the SAP Service Marketplace (<http://service.sap.com/education>) for the most up-to-date information on training courses). The team is responsible for:

- Determining the test plan
- Planning, preparing and monitoring the execution of the test
- Evaluating the test result
- Based on test results, initiating corrective actions

A team leader should be appointed to:

- Ensure sound planning of the test in accordance with the success factors
- Perform a review of the test plan
- Manage the workflow and monitoring during the tests
- Collect and compile the test result from the test team for reporting.

Define Test Scenarios and Requirements

Establish Test Scenarios and Load Profile

For setting up the testing scenario, identify and document the core business processes. You can do this within the framework of a Solution Management Assessment Service (SMA) or Technical Integration Check (TIC) offered by SAP. The complete set of business processes should then be integration-tested to ensure data consistency. Next, all business steps should be tuned individually and an initial forecast can be made as to whether the system will be able to handle the proposed load if no interference with other business steps occurs.

For each of the core business processes, identify the relevant steps and arrange them in several test scenarios. Dependencies between the steps need to be identified and the critical success factors for the steps must be defined.

The steps you need to perform to set up the test environment are described in more detail under the subheadings below.

Introducing our hypothetical example: The BikeComp GmbH is a famous German company that creates revenue by producing and selling bicycles mainly to Asia and Europe.

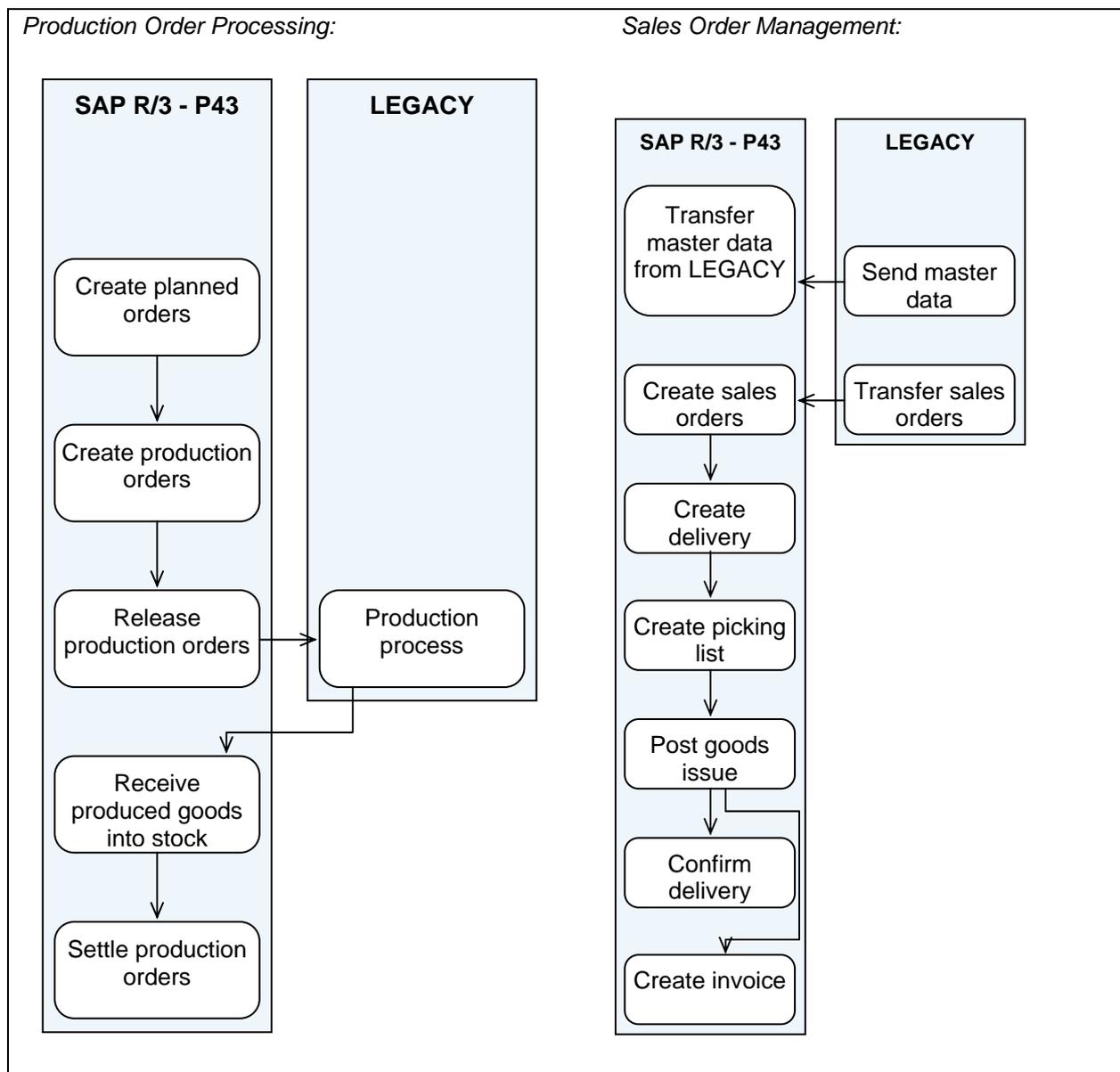
During the implementation phase, the company had a GoingLive Analysis session performed by SAP where many parameter recommendations were given, and these are now implemented into the systems. An integration test was performed to ensure that all customized business steps are consistent. As part of the GoingLive Optimization session, the business steps were tuned to show an optimum response time in single user mode.

Before the planned production start date, the responsible IT managers at BikeComp want to ensure that the system will work successfully within the constraints of the planned load and time schedule. Therefore, they set up a BikeComp Volume Test project.

Select Business Processes for the Test

The first step in setting up the test scenario is choosing the business processes to be investigated. Typical candidates are the company's core business processes as well as other processes known to be time-critical. Time-critical paths within a business process can be identified, based on the planned processing volume, the available processing time window and the response times measured in single user mode.

BikeComp GmbH receives a Solution Management Assessment service (SMA) as part of SAP's Safeguarding Program. One of the essential parts of the service is the determination and documentation of the main business processes. For BikeComp, the two main processes are Production Order Processing and Sales Order Management.



Identify Dependencies between Different Process Steps

For each test scenario, the test team should try to identify possible accesses to common objects from different process steps. These objects can include materials, document numbers or number range objects where accesses may cause lock situations during the volume test and subsequent production activities. Before the test, check whether the SAP Notes 70865, 151672, and 91580 (for material locks), as well as 37884 and 179224 (for number range objects) can be implemented in your system.

Include Interfaces

Interfaces between the different SAP and non-SAP systems can be critical both in terms of throughput and generating errors. Therefore, these interfaces should be part of the volume test. In order to include interfaces in the test scenarios they should be treated as individual business step, as they also can be connected with processing times, throughput and critical success factors.

For more information regarding interface monitoring, see below.

Success Factor: Load Profile

To identify the load profile, document the critical parameters of the individual business steps. If SAP Active Global Support has already delivered services for your solution, look for this information in the questionnaires of the SAP GoingLive Analysis and Optimization services, as well as in the service report of the SAP Technical Integration Check.

It is absolutely essential to collect all information about critical time windows as well as a realistic estimation of the volume of documents or line items processed. At this stage, you can already exclude from testing those business steps that are definitely non critical, that is, have low resource consumption and small run times.

Identify:

- (1) Peak times for the isolated business processes, focusing on time-critical process steps and volume-critical interfaces. For that purpose, refer to SAP service reports (e.g., Technical Integration Check).
- (2) Peak times for each of the identified business-critical interfaces: Identify document volume / peak times due to the sum of transferred documents from several processes. This should be done in conjunction with a SAP Interface Management Optimization service.
- (3) The global activity peak where several business processes are run in parallel. This requires the detailed knowledge of the background job schedule.
- (4) As a side-result of this analysis, you should also identify whether there are still time windows available that you need for the maintenance of the solution.

To identify peak load situations, it is essential to take into account the document volume to be processed and / or the number of concurrently working users for ALL countries / time zones / plants being operated simultaneously at the time in question.

The Interface to the bicycle production system does not need to be taken into account, as during integration testing the runtimes have been in the order of approximately 100 ms per order, which will not contribute significant load to the system. For a later volume test only the grey highlighted business steps will be considered.

Production Order Processing:

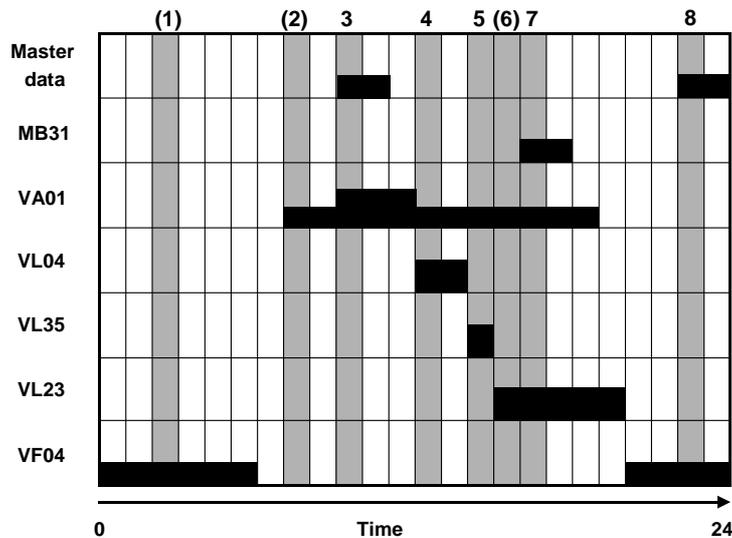
<i>Business step</i>	<i>Number</i>	<i>Processing time window</i>	<i>Peak time window</i>	<i>Peak number</i>	<i>Frequency</i>	<i>Type</i>	<i>Single user processing time per line item</i>
<i>Create planned orders</i>	<i>300 orders</i>	<i>15:00 – 15:30</i>			<i>Daily</i>	<i>Background</i>	
<i>Create production orders</i>	<i>300 orders</i>	<i>15:30 – 16:00</i>			<i>Daily</i>	<i>Background</i>	
<i>Goods receipt</i>	<i>2000 line items</i>	<i>16:00 – 18:00</i>			<i>Daily</i>	<i>Background</i>	<i>800 ms</i>
<i>Settle production orders</i>	<i>300 orders</i>	<i>18:00 – 20:00</i>			<i>Daily</i>	<i>Background</i>	

Sales Order Management

<i>Business step</i>	<i>Number</i>	<i>Processing time window</i>	<i>Peak time window</i>	<i>Peak number</i>	<i>Frequency</i>	<i>Type</i>	<i>Single user processing time per line item</i>
<i>Master data transfer</i>	<i>100.000 data sets</i>	<i>22:00 – 24:00 (Europe) 09:00 – 11:00 (Asia)</i>			<i>Weekly</i>	<i>Interface</i>	<i>800 ms</i>
<i>Create Sales order</i>	<i>220.000 order line items</i>	<i>07:00 – 19:00</i>	<i>09:00 – 12:00</i>	<i>60.000</i>	<i>Daily</i>	<i>Online, Background</i>	<i>400 ms</i>
<i>Create delivery</i>	<i>220.000 delivery line items</i>	<i>12:00 – 14:00</i>			<i>Daily</i>	<i>Background</i>	<i>200 ms</i>
<i>Generate picking list</i>	<i>200.000 delivery line items</i>	<i>14:00 – 15:00</i>			<i>Daily</i>	<i>Background</i>	
<i>Post goods issue + Confirm delivery</i>	<i>200.000 delivery line items</i>	<i>15:00 – 20:00</i>			<i>Daily</i>	<i>Background</i>	<i>300 ms</i>
<i>Create invoice</i>	<i>2.200.000 invoice line items</i>	<i>20:00 – 06:00</i>			<i>2 per month</i>	<i>Background</i>	<i>150 ms</i>

Work out a 24-hour time-grid showing the planned execution time-windows and associated processing volumes for each business step involved in the test. Identify all load scenarios that may differ from this model. The test scenarios that are noncritical or already included in another scenario can be neglected. In a last step, arrange the different test scenarios in a way that is most suitable for the dialog users and the system administrators who have to schedule or start the background jobs.

For the selected business steps, a timetable is created and the different load scenarios are determined.



Easy situations are now separated from the complex ones. For BikeComp GmbH, scenario 1 does not interfere with other processes. It will already have been determined whether the system can use parallel processing for invoice creation. Scenarios 2 and 6 are contained in other scenarios where a similar or even higher load of the same process step is generated.

Consider Recovery Scenarios

In addition to daily or peak business scenarios, we recommend you to consider also recovery scenarios for your volume test.

For this purpose, you have to identify possible scenarios where a system-down may lead to significant data backlogs to be processed both in SAP and in the legacy systems. For a typical backlog which you may encounter after maintenance windows etc., you should set up tests to estimate the time required to reduce a given document backlog (e.g., IDOCs in the R/3 inbound queue).

Based on numbers from business regarding these possible scenarios and the related backlog document volumes, it is possible to make sensible assumptions for the additional system load caused by these incidents.

Success factor: Success Criteria for the Volume Test (KPIs)

To determine if a volume test has been successful, it is essential that every business step owner defines a realistic success criterion for their step (= *key performance indicator (KPI)*). Success criteria may include the average and maximum response times for certain dialog transactions, as well as a minimum throughput per time unit. For background jobs, the criteria may include the runtime or the minimum number of parallel-running jobs.

Success Factor: Test plan

Once you have identified the critical scenarios (including success criteria) and load profile, you have to establish a test plan. This plan describes the test scenarios and the test activities to be performed. It includes the processing volumes and a detailed description how to attain these volumes. If a tool is used for load generation, the test plan describes the necessary parameterization, settings and conditions to enable you to simulate a representative load. For dialog users who participate in the test, the test plan describes what the users need to do. The test plan also contains a time schedule of all test activities.

Here is a link to an example EXCEL template:

<http://service.sap.com/~sapidb/011000358700006461232006E>, which allows you to summarize the key figures and current issues of your volume test in a well-manageable way.

Review and Sign-Off the Test Setup

At the end of the planning phase for the volume test, it is essential that the managers and business process owners review the test scenarios and critical success factors. Each business process owner should evaluate whether all significant business steps were considered and sign-off the test plan as having been correctly setup. The Appendix contains a link to a template that you can use as a guideline for the review and sign-off.

Determine Requirements for Test Landscape (Sizing or Configuration or Test data)

Success Factor: Hardware Capacity and System Configuration

Ideally, the simulated load and the hardware used for the volume test should be identical to the load and hardware of subsequent production operation. However, for various technical, organizational or economic reasons, it is not always feasible to simulate the full load expected in subsequent production operation. In addition, it is never really possible to execute the test on hardware exactly *identical* to the hardware planned for production.

The hardware and software components involved in the volume test have to be set up for optimal performance and stability. The hardware must be powerful enough to handle the planned load both during the volume test as well as during subsequent production operation.

To minimize the risk of general bottleneck situations due to insufficient hardware or nonoptimal system configuration, SAP Active Global Support supports you via the SAP GoingLive and EarlyWatch services. For already productive systems, the EarlyWatch service provides you with up-to-date recommendations regarding system configuration and parameterization.

Very important: Make sure that the hardware used for the volume test is exclusively reserved for this purpose while the volume test is run. In-parallel usage of the same hardware or system(s) for other purposes (as, for example, training system) can have disastrous consequences for the volume test and the interpretation of its results.

AVOID TOO HIGH CPU LOAD

The closer the CPU load is to a 100%, has a nonlinear effect in relation to the number of documents processed – meaning CPU load becomes important. If no application locking problems occur (for instance locks on number range table NRIV or material master data may impede the creation of documents although it would be necessary with the available CPU power), typically, one may expect the following qualitative relation between CPU load and numbers of processed documents:

CPU load in test system	20	40	80	90	95	100%
No. of processed docs in test system	100	200	380	390	393	370

This behavior (numbers - not to be taken literally - for illustration purposes) is caused by CPU-internal processes which dispatch the exhausted CPU resources, for which several work processes are waiting at the same time. Once the CPU is entirely exhausted we even may observe a negative impact on the response time per document in comparison to less loaded CPUs.

As a consequence, the necessary hardware for production derived from a volume test may be overestimated:

Imagine a KPI of 1000 documents per hour and accept the results of a volume test with a measured CPU load close to 100%. According to the table above, you would need a CPU power of $1000 / 370 = 270\%$ plus $20\% * 2,7$ (as safety buffer to limit the CPU exhaustion in production to 80%). For the required productive CPU, you end up at $> 330\%$ of the test system CPU capacity.

Now reduce the document load in your test to end up with a CPU load of 80%. This corresponds to a throughput of 380 documents. For the CPU of the production system, this would result in $1000 \text{ (KPI)} / 380 \text{ (throughput during test)} = 260\%$.

Recommendation: If it is not possible to perform the volume test on the future production system, you have to avoid the misinterpretation of volume test results described above. Make sure that during the volume test the CPU load of the test environment does not exceed 80% (i.e., 15...20% idle should be achieved).

The simulated load must be adequate for the hardware on which the test is performed. If the test is performed on a hardware that is less powerful than the hardware planned for production operation, the simulated load might need to be scaled down correspondingly in order not to create bottleneck situations. However, if you do perform the test on the production hardware with only part of the planned load, the simulated load must still be high enough compared to the available system resources to serve as a basis to project the system behavior for the full load. If the load generated in the volume test is too low, some bottlenecks might not become visible.

Implement Test Environment

Implement test landscape

In case that you have to prepare a volume test landscape from the beginning, independent of the complexity of your solution, you should consider this implementation as a subproject of your volume test project and plan and execute it accordingly.

SAP Consulting may provide you with adequate support regarding the technical setup of your test environment. For further information, we refer to SAP Test Management Consulting (<http://service.sap.com/TMC>).

Check the Technical Environment

Verify that the parameter recommendations of the last service session (that is GoingLive sessions) have been implemented. If the last report from such a service is older than two months or you have changed your technical IT environment, ask SAP for a new service session to get the latest parameter and system recommendations. It is important to apply these parameter recommendations *before* a volume test.

Prepare Data Basis (Master or Transactional data)

Success Factor: Representative Data Basis

Since database tables (especially transaction data tables) start to grow after start of production, database accesses that are not fully specified tend to get slower and more resource-expensive. If the volume test is performed with almost empty database tables, the response times in subsequent production operation will be higher than measured in the volume test. Therefore, for the volume test to

be meaningful, you need to populate the database tables with representative data that is both master and transactional data.

Success Factor: Realistic Test Data

The volume test should be a realistic simulation of subsequent production operation, and therefore the provision of suitable data is important and often is the key factor for the success of a volume test and a smooth production start.

Transactional business steps should be performed with the line item distribution that is identical to that used later. It is important for the critical success factor definition and the evaluation of the results to know the average number of line items per document. But you should also test the maximum numbers since nonlinearity can occur. For programs and reports, the processed data volume should be within the order of magnitude expected during production operation.

Another way in which test data must be realistic is that the customer numbers used, the materials and so on, must all be different and reflect the subsequent operations for the business scenario. This results in a detailed analysis of the business scenario by the business process responsible and an explicit documentation in the test scenario.

Migrate Test Data to Test Landscape

You should carefully consider which amount of data has to be transferred to the test environment. Depending on the business scenario you want to simulate, the complexity of this data migration may have the same order of magnitude as the data transfer for production preparation. Therefore, this step in the preparation procedure for your volume test has to be carefully planned, and a sufficient time slot has to be reserved for this data migration.

In this sense, the migration of test data to the test landscape may be regarded as a good last test for the final data migration (assuming that the simulated scenario covers more or less the future productive one, with respect to both data volume and complexity of the business scenarios).

Implement Test Tools

Manual Testing verses. Load Generators

The ideal test for a future productive scenario would be if you had a realistic number of end-users for your IT solution which would use the test environment as it they would for daily business.

Such a test, however, may suffer from following shortcomings:

- Availability of sufficient test users: Since the tests would have to be executed outside the regular working hours, the availability of a sufficient number of test users may become questionable.
- Insufficient description of test cases, that is, only really experienced users would be able to execute the manual test cases.
- High personnel costs: The tests would have to be made in addition to the end-users' regular work. This in turn, implies significant additional costs.
- De-motivation of testers: In case that the test run encounters an unexpected performance bottleneck, the natural consequence is that the test participants stop working because they are unsatisfied by the system performance.
- Repetition of volume tests: If the volume test runs have to be repeated (for what reason so-ever), it may become difficult to have available the booked resources for the required additional time.

Implement Load Generators

Alternatively to manual testing, you can run the volume test using an automated test tool that simulates user activities.

In comparison with real test users, employing such a test tool has many advantages:

- Even if the costs for license per simulated end-user may be sizeable, normally the price to be paid will be lower than the personnel costs of hundreds of end-users (in particular when repetitive test runs are planned or several volume tests for various roll-out phases in your project are necessary).
- Tests can be easily repeated, with the identical set of dialog data entered into the system
- The number of simulated users may be gradually increased until the maximum workload has been reached.
- Depending on the tool you decide to use in your test, there are various options to define particular user profiles and combine several types of end-users to generate the entire range of simulated end-users.

There are a number of tools available on the market, which may be employed to generate dialog workload.

SAP (free of charge): The SAP Test Workbench is integrated into the SAP Solution Manager, SAP's evaluation and implementation portal for mySAP.com. Within the SAP Test Workbench, you have access to the test tool eCATT especially developed for mySAP.com components. For more information about SAP eCATT, see the SAP Service Marketplace: go to <http://www.service.sap.com/eCATT>.

Please note: For the simulation of realistic end-user scenarios (users executing transactions at different point in times, i.e., not working exactly in parallel, and requiring "think times"), the current version of eCATT is not suitable.

SAP: SAP consulting provides support for the test setup and also has developed a tool for GUI-based load simulation called "SAP-GUI Scripting". For more details see the media library on the SAP Service Marketplace: go to <http://www.service.sap.com/VTO> → Media Library.

Certified by SAP: For testing cross-system integrated functionality, there are many external SAP testing tools on the market that are certified by SAP, for example:

- AutoTester / AutoControler from *AutoTester Inc.*
- WinRunner / LoadRunner / TestDirector / Test Suite from *Mercury Interactive*
- Rational TeamTest / Rational Robot from *Rational Software*
- QACenter from *Compuware*

For more information on our Software Partners and their respective product in this area, select the Software Partner list from the SAPNet (<http://www.sap.com/partners/directories/searchpartner.epx>). (Note that the mySAP.com logo in the list indicates that the Software Partner achieved a certification for a recent release of this interface that is valid for the mySAP.com Edition.)

If you want to run the testing tool on other mySAP.com components, you should be sure that the tool has been certified for that mySAP.com component.

Mercury LoadRunner: Due to the broad spectrum of applications (SAPGUI, web-applications) covered, this tool is the most versatile currently available on the market.

From 2006, SAP offers LoadRunner as "SAP LoadRunner by Mercury". Contact your SAP sales representative for more details and a price list.

Implement or Setup Monitoring Tools

SAP Components: For monitoring the components of the SAP components of your solution, refer to the next section. There, you find details regarding performance monitoring of R/3 application servers and interfaces. The monitors base on the Computing Center Management System (CCMS, transaction RZ20) which is available on each SAP system.

To monitor the CPU and RAM utilization of servers without SAP Basis, such as stand-alone database servers, remote SAPOSCOL needs to be installed on that server to allow the monitoring from within an SAP application server (see SAP Note 436186 and 20624).

Alternatively shell/batch script in the operating system level can be programmed and applied to collect additional information of CPU, Memory and IO activities in more detail and frequent way.

Non-SAP Components: In case that you require performance key figures for non-SAP components of your solution make yourself familiar with the respective monitoring applications.

Monitoring at Operation System Level: Each operation system provides you with tools for performance monitoring. Since the SAP CCMS does not have access to all possibly important performance key figures on OS level, we recommend you get familiar with the OS performance monitoring tools and to employ them for detection of bottleneck situations.

Technical Preparation for Assistance from SAP

If you need assistance from SAP during the volume test (as mentioned in the **Alternative Offering** in sub section above), ensure that the following connection is available through SAPNet Frontend and that a suitable user ID and password has been created:

- R/3 Support connection to the system(s) to be monitored
- Additionally, the remote SAPOSCOL must be installed on each server without SAP basis (for example, the stand-alone database server), so as to allow monitoring of the CPU and RAM utilization from an SAP application server (see SAP Note 20624 and 436186).
- If particular non-SAP or JAVA-based applications have to be monitored, make sure that the appropriate monitoring tools are installed and ready for use in the respective solution components (e.g., Wily Introscope for JAVA applications).

Before you start the tests, ensure that R/3 has been restarted to reset some of the monitoring parameters.

Regarding the interface monitoring, the remote support by SAP will be facilitated if you provide SAP with system access via pcanywhere.

Run Test and Evaluate

Run Volume Test and Monitor

Coordinating the Test

The test coordinator is a key resource in the test team who is responsible for executing and monitoring the test. The main responsibility aspects of the test coordinator are:

- Executing the test according to the test plan
- Ensuring that all actions within the test scenario are started and ended properly
- Deciding when a test scenario is finished
- Communicating with the dialog users
- Acting as the first contact for complaints and problems
- Communicating with the monitoring team
- Ensuring that the right measurements are performed at the right time
- Deciding whether the current situation requires the test plan to be changed

Monitoring SAP Application Servers

For most of the key figures relevant to monitor a volume test, SAP application server maintains statistics that can be evaluated after the end of the volume test. Continuous monitoring during the execution of the volume test is required to detect temporary bottlenecks that are not logged explicitly and that would, on average, level out. The monitoring frequency must be high enough to monitor temporary peak or bottleneck situations or temporary contention problems such as exclusive lock waits.

A monitoring log with the relevant key figures should be maintained for each test scenario to document the measured values.

For technical details, refer to the appendix "Monitoring SAP application servers".

Monitoring Interfaces

This section is intended for staff members who are familiar with the basic ideas of interface monitoring. Therefore, refer to the appendix for technical details.

Collecting Performance Statistics Data

After Volume test finish, run transaction ST03n and collect general performance statistics and performance statistic for each important transaction or report (identified in the test plan).

ST03n >> change to *Expert mode* >> *Detailed Analysis* >> *Last minutes Load* and choose the time period of the volume test. For General performance statistics, choose *workload overview* under folder *Analysis Views*. To get performance statistics per transaction or reports choose *Transaction Profile* >> *Standard*. For more detailed statistics of each transaction or reports, highlight the transaction or report you want to drill down and click button *Single Records*; double-click the transaction or job name for single records detail.

Documenting the Test Results

For reporting purposes as well as for later follow-up on issues detected during the volume test, a detailed written report is essential, as a documentation of the test results. The documentation should contain the hardware setup, the business processes and process steps, the test scenarios and also the monitoring results. Furthermore, an evaluation of the results should be performed as described below.

Success Factor: Documentation of Test Results

For each test run, the results should be documented. This includes, for example, the actual volume processed and required runtime. Relevant performance statistics are either stored in the SAP system and can be obtained from special transactions or have to be collected manually during the test run.

In addition to collecting performance statistics, document any other conspicuous events or general observations. For each success criterion defined, document the extent to which the criterion was fulfilled by the test run.

Influence of Untested Processes and Process Steps

Business processes, process steps and programs that do not interfere with any other business step or program do not necessarily need to be part of the volume test. However, the resource consumption of these components has to be taken into account if the hardware size is evaluated. For this purpose, determine the resource consumption of these programs and steps, either during the integration testing or the single-user performance optimization.

Evaluate Test Results

The result of the volume test is the determination whether the hardware is sufficient to handle the expected system load and – at the same time – the comparison of measured and required performance according to the critical success criteria. The following section describes some rules of thumb you can use to evaluate the results of the volume test from the monitored values.

R/3 Application Servers

Maximum CPU utilization: For each measurement of the hourly CPU utilization sum up the CPU utilization by user and system and enter the maximum value as 'Max. CPU utilization [%]'.
 CPU rating: If the average CPU load increases to more than 70%, some CPU wait situations may occur. An average CPU load of more than 90% clearly indicates a CPU bottleneck situation.

Memory Used:

1. Calculate an upper limit for the maximum memory usage MaxMemUsage1 as the sum of memory used by SAP shared buffers, SAP work processes and users contexts. An upper limit for the memory used by user contexts is calculated by adding up the maximum usage of main memory for SAP Roll Memory, SAP Paging Memory, SAP Extended Memory and SAP Heap Memory.
2. Calculate an upper limit for the maximum memory usage as the sum of RAM [MB] and the maximum paging rate measured within the hourly average. Note that you have to take the page-out rates for Unix servers and the page-in rates for Windows servers.
3. Enter the minimum of MaxMemUsage1 and MaxMemUsage2 as 'Memory used [MB]'.

Maximum Paging: Based on the measured hourly paging rates, enter the value for 'Max. Paging [% of RAM]'.

Memory Rating: The memory usage should be less than 125% of the RAM size. The hourly paging rates should be less than 25% of the RAM size. A memory usage of more than 150% of the RAM size or paging rates of more than 50% of the RAM size indicate a memory bottleneck.

SAP Memory Usage:

Roll area: The maximum usage ('Max. used [KB]) of the roll area, configured as shared memory ('In memory [KB]'), should be less than 80%.

Paging area: The maximum usage ('Max. used [KB]) of the paging area ('In memory [KB]' + 'On disk [KB]') should be less than 80%

Extended Memory: The maximum usage ('Max. used [KB]) of the Extended Memory ('In memory [KB]') should be less than 80%.

Lock Situations: Lock situations may occur in every system. The duration of the lock, the frequency of occurrence and the locked object type are of importance to judge whether and what actions should be taken.

Interfaces

Interfaces can be considered as critical if they have one or more of the following requirements:

High throughput requirements (Rules of thumb)

- Maximum data volume per hour > 1.000 or
- Average data volume per day > 10.000 or
- Maximum number of documents per hour × Response time [sec] > 3600
- High data volume >10.000 data records or documents x line items per day

Requirements for good response time are (rule of thumb):

- < 2 seconds for synchronous interfaces (sender waits for response of receiver)
- < 1 Minute for asynchronous interfaces (sender does not wait)

High requirements for data consistency

(Business) requirements for the time frame

Recommendations

- Use package processing (reduces the overhead)
- Use RFC Server groups for processing of incoming IDocs (load balancing)
- Use logon groups on the sender and receiver systems
- Process data in parallel with the required degree
- If possible limit the number of calls processed (that is, RFC-quotas)

Use the outbound scheduler for tRFC and qRFC (SMQS)

BikeComp GmbH has performed all tests and monitored the parameters as defined. In a last step, the final report has to be written and the hardware and business process steps must be evaluated.

Maximum CPU and Memory Utilization

Server	Max. CPU utilization [%]	CPU Rating	RAM [MB]	Memory used [MB]	Memory used [% of RAM]	Max. Paging [% of RAM]	Memory Rating
Sapaix01	87		2048	2850	139	45	
Sapaix02	66		2048	1856	91	12	
Sapdb1	77		4096	3200	78	10	

From the monitored values, the hardware capacity was evaluated and showed a possible bottleneck on server SAPAIX01, which could maybe avoided by redistributing load onto the other application server. The database server also shows a somewhat too high CPU utilization. Especially if the untested processes are taken into account, the hardware might not be sufficient.

Sales Order Management, Create Sales Order

Transaction/Program	Critical success factor	Value required	Value observed	Rating
VA01	Max. response time for average order	5 sec	4 sec	
VA01	Max. response time for 100 line item order	70 sec	90 sec	
VA01	Max. throughput per hour	4000 orders	4210 orders	

For the business process step 'Create Sales Order', three critical success factors have been defined: the response time for an average order with 10 line items, the maximum response time for a 100 line item orders and the throughput for the peak hour. The response time for average orders can be found directly in the monitoring protocol. For the orders with 100 line items, a special monitoring process has to be defined, since the value can hardly be obtained by the general statistics. Here it would be more suitable to monitor certain users who perform only these big orders and get their average response times from the statistics.

For process steps, whose critical success factors have not been fulfilled, a careful analysis of possible reasons should be done. In addition to this, some of the application recommendations of the GoingLive Optimization session should be considered, e.g. the number of group conditions in price determination could be the cause for the longer response time with more line items per order.

Optimization and Repetition of Test Runs

In case that you experience performance bottlenecks and the necessity for performance optimization, a repetition of particular parts of volume testing becomes necessary.

Recommendation:

- Before starting the test runs, you have to reset your test environment to ensure that you start the volume test with a system status comparable to the one available for the previous test run(s).
- After the reset, you should consider that the various buffers and caches (e.g., database cache, application table and program buffers) of your system landscape need a certain time to get filled again. As a consequence, when starting the test run your systems will show an initially lower performance than later on.
- Generally speaking, there is no prediction possible regarding the time required to refill the buffers and caches. Depending on the application and database, the order of magnitude for the necessary time may range from a couple of minutes and hours.
- To avoid a misinterpretation of a possible temporary performance bottleneck at the beginning of the test run, it is recommendable to increase the workload on the system gradually, until the maximum workload has been reached. As a consequence, in your performance bottleneck and post-test analysis, you have to neglect the initial system behavior and concentrate on the time period where the workload was at its maximum.
- To evaluate the success of optimization measures it is essential that you carefully protocol the changes made in the test environment.
- Use the measurement results of the last run before the optimization cycle as a baseline for comparison with the response times after optimization.

Further Information

- Strategic motivation and project guidelines for volume testing are discussed in detail in *Project Handbook Load Testing and Performance Tuning* by George W. Anderson and Michael Mißbach, published by SAP Press (English edition 2003; revised German edition 2005).
- For performance tuning of SAP solutions, refer to *SAP Performance Optimization Guide* by Thomas Schneider, published by SAP Press (4th edition 2006).

Feedback and Questions

Send any feedback by formulating an SAP customer message (short description “Best Practice Volume Testing – Feedback”) to component SV-SMG-SER at <http://service.sap.com/message> in the SAP Service Marketplace.

APPENDIX

Monitoring SAP Application Servers

CPU: The hourly average CPU utilization can be obtained from the Operating System Monitor (ST06 or OS07) after the test. Additionally, the CPU utilization should be monitored periodically during the volume test to be able to identify temporary CPU bottlenecks that level out within the hourly average.

RAM: The hourly average RAM utilization (paging rates) can be obtained from the Operating System Monitor (ST06 or OS07) after the test. Additionally, the RAM utilization (paging rates) should be monitored periodically during the volume test to be able to identify temporary RAM bottlenecks that level out within the hourly average.

For each SAP application server of the OLTP-R/3 System the following key figures need to be monitored:

- Response times and memory utilization of the programs and transactions that implement relevant business process steps
- Utilization of SAP shared buffers and SAP memory
- Number of active users.

Continuous monitoring is required to detect lock wait situations and to keep an overview of the current system activity while the volume test is in progress.

Response Times: The response times, CPU times and database times of the programs and transactions that implement the business process steps involved in a test scenario can be obtained from the Single Statistical Records (STAD). In STAD, select the display mode 'Show business transaction sums' and restrict the evaluation of the single statistical records to the time interval of the test scenario. To restrict the display to a particular transaction or program, set a filter in the screen 'Workload: Business Transaction Analysis.' To drill down to task type level, click button 'Open/close all' and select 'Both aggregations'. To download the displayed list into a spreadsheet, use the 'Download' button.

Use the spreadsheet to calculate the following key figures:

- Number of executions
- Average response time per execution
- Average CPU time per execution
- Average database time per execution
- Maximum response time of all executions.

Note that you should consider the time for the complete execution of a program or transaction, rather than look at the average response times per dialog step.

SAP Shared Buffers: Snapshot data about the utilization of SAP shared buffers can be obtained from the SAP Memory Configuration Monitor (ST02) after the test. Immediately after each test scenario, make a note of all buffers that have swapped.

SAP Memory: Snapshot data and high water marks of SAP memory utilization (extended or roll or paging or heap memory) can be obtained from the SAP Memory Configuration Monitor (ST02). Make a note of the high water marks after each test scenario.

Number of active users: The number of active users can be obtained from the Global User Monitor (AL08). Make a note of the number of interactive users and RFC users during each test scenario.

Database locks: Snapshot data about exclusive wait situations for database locks can be obtained from the Database Lock Monitor (DB01). If exclusive database lock waits occur, document the locked object, lock holder (program/transaction) and lock waiter (program/transaction).

Database Performance: Snapshot of data buffer/cache quality, SQL cache catalog/pin ratio (ST04). For further detailed analysis, click button *Detail Analysis Menu* on ST04. Prerequisite: to get exact snapshot information from ST04, reset the counter by clicking button *Reset Counter* once before volume test start.

Current system activity: The main tool for monitoring the current system activity is the Global Work Process Overview (SM66). Continuously monitor the status of the work processes. Make a note of aspects such as the following:

- Are there always free work processes available (that is, work processes with status 'waiting')?
- Are there work processes in status 'stopped'? If so, see column 'Reason'.
- Are there work processes in status 'terminated'?
- For work processes in status 'running': See column 'Action/Reason for waiting' for the current activity.
- Are there work processes waiting for database locks?
- Are there work processes waiting for semaphores?

To deepen the analysis, use the appropriate detail monitor, for example, the Database Monitor (ST04), Database Lock Monitor (DB01), Operating System Monitor (ST06), and SAP Memory Configuration Monitor (ST02).

Errors: After the volume test, investigate the System Log (SM21) for error messages and warnings that indicate potential functional or stability problems. Additionally, check for ABAP dumps (ST22) and update errors (SM13) that occurred during the volume test.

Load: For each test scenario, measure the actual system load (for example, the number of active users, number of sales orders processed). Compare this data with the load planned for the volume test.

Monitoring Interfaces

Types of Interfaces

Flat File (File upload): A flat file is transported via FTP/CFT or any means of the operating system and uploaded into the application program.

Batch Input / Call Transaction: Based on simulating the online-input of transaction codes and data.

For a lot of data, SAP default programs for instance use are delivered. Alternatively, you can use the **Transaction Recorder** (TA: SHDB).

- **Batch Input (asynchronous):** External data is read by an ABAP-program and saved in batch input sessions in the R/3 database. Any action required for transferring data with regular SAP transactions into the SAP system is recorded.
- **Call Transaction (synchronous. and asynchronous):** The ABAP-program uses the „CALL TRANSACTION USING“ – the data is completely processed “inline” the own program. The ABAP-program is preparing the data and then calls the according transaction for immediate processing.
- **Direct Input:** Since a complementation of Batch-Input or Call Transaction you can use this to takeover huge data sets. No dynpros are run in the R/3 but function modules are called to do the necessary checks and entries in the database tables.

RFC

Similar to the RPC protocol (remote procedure call) the RFC client calls a function module *remotely* on the RFC server. SAP offers additional RFC libraries as well as services and programming APIs, which enable customers to interface with R/3-, as well as with R/2-systems or external programs.

BAPIs (Synchronous and Asynchronous.)

Via this method of a SAP business object type you can access R/3 application functions. The implementing details are encapsulated so just the interface functionality is visible to the outside.

Before using a BAPI method getting access to a SAP business object an application needs to have the technical connection, either by:

- Direct use of the BAPI-Function module (connection via RFC)
- BAPI ActiveX Control (Windows 95/98 and Windows NT)
- BAPI C++ class libraries
- BAPI Java class libraries
- R/3 DCOM Component Connector (Windows NT)
- ObjectBridge from Visual Edge

The following information is needed:

- Name, Information about the interface, Import and Export parameters for data

ALE / IDOC (synchronous and asynchronous)

ALE contains the business scenarios and function modules that allow transfer data to or from an R/3 System. It is linked closely with Workflow Management within R/3. In addition, it can also trigger follow-up actions in the R/3 System:

- Transaction data, which is data from applications
- Master data, such as customer or material master data
- Customizing data used for an overall ALE view

IDoc (Intermediate Document): A data structure of SAP applications at the interface level. IDoc types form the container for the data to be exchanged. By linking SAP systems directly, IDoc's can be transmitted without a mapping into EDI standards. This is the ALE approach. IDoc's can be distributed as a Flat File or by tRFC

EDI (Electronic Data Interchange): Describes the technical structure of a document. This enables the recipient to automatically process the document by his business software.

Key Performance Indicators (KPI's)

The relevant parameters for the interfaces are according to performance, response times and data throughput.

Batch Input/Call Transaction (Direct Input)

- Processing time in peak or non peak times
- Available time window
- Maximum number of documents processed per hour
- Number of successfully transmitted transactions
- Number of erroneous transmissions

RFC

- Maximum processing time
- Average response time during peak work hours
- Maximum number of documents processed per time
- Available time window

BAPI's

The KPI's for BAPI's depend on the chosen technique, that is, RFC

- Average processing time
- Maximum possible runtime
- Maximum number of BAPI's per time
- Available time window

ALE/IDOC

- Average number of documents per day
- Maximum number of documents per hour (or number of data records)
- Average number of line items per doc
- Maximum number of line items per doc
- Number of erroneous documents
- Available time window

Monitoring

Prerequisites: All involved systems and in special the legacy systems are available and prepared with realistic and consistent test data. The test procedure and responsibilities are defined. Monitoring and Logging/Tracing is enabled. Transactional security and functional reliability is tested. Data consistency is guaranteed.

SAP offers an Interface Management Optimization service, which supports you regarding the topics:

- Performance and throughput
- Configuration
- Transactional security and data consistency
- Interface management procedures

Note: functional issues are usually not covered by this service. For more information on the SMO Service Interface Management see <http://service.sap.com/ifm>

For checking the settings as a precondition:

SM59 RFC destinations
BALE ALE administration
WEDI EDI/IDOC basis
BD87 select specific IDOC

...

Interface Requirements Template:

Interface Name	Maximum Data Volume*	Response Time Requirements	Other requirements**

* Number of documents × average line items or number of data records. Note in which time frame the documents have to be processed (per hour or per day)

** Fill in other requirements if important for the interface. For example: Very high requirements for data consistency, very high availability (business continuity) and so on.

Flat file (File upload): This has to be done by means of the operating system (scripts) or external tools (HP openview, TIVOLI, else) to check for data consistency and functionality.*

Batch Input / Call Transaction: Use transaction SM35 to monitor a batch input session. Additionally you might want to see the protocol file for the batch run in \usr\sap\<SID>\SYS\global\

Focus areas are:

- space required on the database → see DB logs for critical rising of data amount

- long running transactions → check the rollback segments (Oracle) or transaction logs (other RDBMS)

Be aware that in 'Direct Input' there is no processing protocol and no monitoring in TA: SM35, use TA: BMV0 instead.

RFC:

- Monitor the system utilization via SM66 (consumption of work processes)
- Watch the RFC monitors in SM58 or SMQx (1 for outbound, 2 for inbound, R for inbound scheduler and S for outbound scheduler) and the gateway monitor for the logged on clients (SMGW – Goto – logged on clients)

Focus on:

- Resource consumption (memory, CPU, work processes) → See operating system monitor via OS06, global work process overview SM66, availability and consumption of resources in SMQS – goto - qRFC resources
- Erroneous setup → Check the resource settings of RZ12 (RFC server groups) and general destination settings in SM59, also in SM58 – special options (for tRFC)
- Monitoring the runtime → Check the response times via STAT/STAD and the system load in ST03(N)

BAPI: The monitoring concept depends on how the BAPI is called:

- synchronous
 - BAPI ActiveX Control
 - SAP DCOM Component Connector
 - Underlying function module

The monitoring has to be done in the program that uses the BAPI's by querying the standardized return codes.

- asynchronous
 - ALE → see ALE monitoring

IDOC/ALE: First decide whether you will do active or passive monitoring

Active monitoring:

- Use report RSEIDOCM to actively monitor the processing of an IDOC
- WE06 active IDOC monitoring (as of Rel. 6.10)

Passive monitoring:

- SM66 and SMGW give you an overall view for system activities and load.
- Check the IDOC processing via transactions:
 - WE02 - IDOC overview / with selection
 - WE05 - show IDOC
 - SMQR, SM58 - monitor RFC processing
 - RZ20 - monitor ALE/EDI objects
 - SM36 / SM37 - monitor background job for errors
 - ST03 / STAD/STAT - Check performance of IDoc processing
- Watch the system for:
 - Transfer errors in the sending system
 - Application or technical errors when processing the data in the receiving system
 - erroneous documents (IDOC's)
 - Serialization errors (if serialization is required and used)
 - System and database load
 - Data growth

© Copyright 2006 SAP AG. All Rights Reserved

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, and Informix are trademarks or registered trademarks of IBM Corporation.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

The information in this document is proprietary to SAP. No part of this document may be reproduced, copied, or transmitted in any form or for any purpose without the express prior written permission of SAP AG.

This document is a preliminary version and not subject to your license agreement or any other agreement with SAP. This document contains only intended strategies, developments, and functionalities of the SAP® product and is not intended to be binding upon SAP to any particular course of business, product strategy, and/or development. Please note that this document is subject to change and may be changed by SAP at any time without notice.

SAP assumes no responsibility for errors or omissions in this document. SAP does not warrant the accuracy or completeness of the information, text, graphics, links, or other items contained within this material. This document is provided without a warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

SAP shall have no liability for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials. This limitation shall not apply in cases of intent or gross negligence.

The statutory liability for personal injury and defective products is not affected. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third-party Web pages nor provide any warranty whatsoever relating to third-party Web pages.