Building an Adaptive Infrastructure for Education Service Providing

Matthias Mohr, Tobias Simon, Helmut Krcmar

Technische Universität München, Lehrstuhl für Wirtschaftsinformatik,
Boltzmannstr. 3, 85748 Garching, Telefon (089) 289-19530,
E-Mail: {matthias.mohr|simont|krcmar}@in.tum.de

Abstract: Educational institutions using enterprise software systems within their courses show a very particular behavior in system usage, different from that of operational businesses. This requires an adjusted operating and service model for the underlying IT systems. A service provider running and maintaining such educationally dedicated systems can meet bigger parts of these special requirements by implementing a flexible IT infrastructure which adapts to the customers' demands. We propose a concept based on a blade server architecture which allows a flexible handling of an SAP® system landscape. The technical concept is embedded into a proposed model of providing services for educational institutions.

Keywords: IT Training, System Usage, ASP, SAP, Flexible Infrastructure, Blade Server Architecture

Enterprise Software Usage for Educational Purposes

The products of German SAP AG (e.g. R/3®) are among the most commonly used enterprise software packages which support a company's processes and information flow. It therefore is utilized in several departments of many companies of different size and industry [Dave98]. The demand for SAP skills among employees is on the rise since years [WaSc99, p. 3; Lang03].

Not surprisingly the different features of SAP software are addressed within **educational courses and trainings**, e.g. by IT training companies, system integrators, consulting firms, internal training departments and even by institutions of higher education. Enterprise software applications are complex information systems that contain a lot of issues which are addressed within educational sessions, for example architecture, technology, programming and business applications [WaSc99].

However, the overhead that the operation and maintenance of complex IT systems present often complicates the use in education. One possibility to reduce this overhead is to acquire services from an application service provider (ASP) [Bece[+]00, p. 41].

Service Provisioning for Educators

Application Service Providing (ASP) is a service which allows customers to use software services provided by another company. The services are centrally operated and maintained and are contractually fixed [Krcm03, p. 310]. The service can be used without having to care about the hardware, its administrative work and the way the service is fulfilled. The computing resources are not located at the own company site and the knowledge of the system is based at the service provider. ASP comes along with a technical support package, helping the customer to ease the use of service and taking care of providing solutions, if problems arise. Emergency support is another service of work. The ASP concept can be applied to nearly any type of application and is commonly used for deploying an array of software ranging from basic office suites, such as office applications, to large enterprise resource planning systems, such as SAP [Wals03, p. 103].

A lot of commercial providers of educational resources like teaching concepts or teaching material use the term "Education Service Providing" in order to describe their offerings. We understand **Education Service Providing (ESP)** in a more technically oriented way: it is both an add-on and a modification of the traditional ASP service: Add-on means the ASP additionally offers education specific services, e.g. train the trainer packages and support for problems referencing special issues (concerning the hosted application) located in the educational sector like e.g. class controlling or system based examinations. Traditional ASP is industry-independent whereas ESP is optimized to be utilized in the educational sector, where the hosted systems have to be prepared to accomplish the requirements of trainers and students. This includes **modifying** both the hosted software itself and the **technical infrastructure** running the software. These steps are necessary because of the special usage patterns of educators.
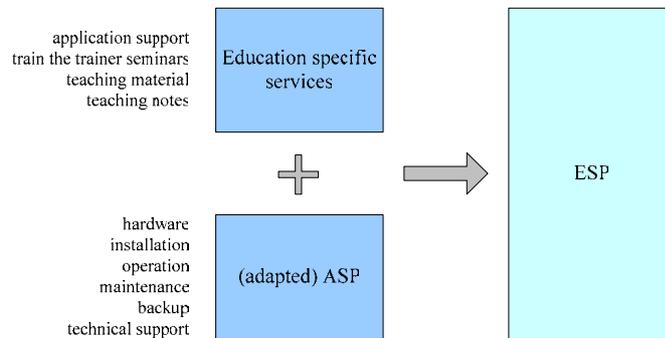
Figure 1: From ASP to ESP

With this definition of ESP, the traditional ASP characteristic „software as a service" [GrMa02] moves towards „educational resources as a service".

System Workload Patterns within educational environments: The Whiplash Effect

Educational institutions have some characteristics in attitude that result in special requirements for the underlying technical infrastructure used for the education process. Those requirements are quite different from those of operational businesses: Unlike enterprise systems used in their intended context within companies, educationally dedicated systems have to bear the peculiarity of the educators' usage patterns. There are three main characteristics that can be identified: **long-term discontinuity, short-term discontinuity and atypical workload**.

Discontinuous workload on the long run means the load on the systems varies extremely during a year: depending on holiday times and course periods the system usage varies extremely. On the short run, depending on the shape of an educational session, there is an increased demand of computing power during the hours of a course, followed by the resources running in an idle state till the next course takes place. These discontinuity phenomena can be approved e.g. by using SAP's standard workload analysis transactions like ST03N or ST06, by operating system monitoring and by interpreting SAProuter's logging files. Most of the time the underlying hosts rest and so valuable assets remain unused. But at the same time high and peak load situations must be handled: parallel courses must be possible without loss of computing power of one of the courses.

Educators also tend to an atypical usage of a system. Use cases which are usually done only once by a single person are worked through in a totally different manner: E.g. 20 students running an MRP operation at the same time or 20

students in a data warehousing course modifying the data dictionary simultaneously. This unusual behavior, which does not happen to the systems in the real world, also results in a heavy system usage. Another characteristic is the fact that educators don't like to work on used and "dirty" systems: For new courses educational institutions regularly need new systems or clients. Those copy runs also highly burden the system. In case of complete IDES systems (pre-customized SAP systems which represent a model company) this is often about a complete reinstallation.

The following table provides an overview about the education specific workload patterns described above:

	Workload Pattern	**Reason(s)**
Long-term discontinuity	**cyclical variations** within a year	holidays, lecture-free periods
Short-term discontinuity	**oscillations** during course hours	more or less heavy exercises
Atypical workload	multiple execution of **one-time use cases**, e.g. MRP runs	use cases are integral part of educational lesson
	simultaneous write access to **central components**, e.g. data dictionaries	access is integral part of educational lesson
	frequent execution of **client copies** and **reinstallations**	educators regularly need new clients or systems

Table 1: Education specific system workload patterns

Those three different types of usage patterns don't occur separately but aggravate each other by **overlapping**. For example, during summer course weeks (long-term) within a production planning lesson (short-term), a lot of students are each running a material requirements planning (MRP) transaction (atypical workload). During only a few hours, a maximum of computing power is needed whereas some days later, the system will be running in an idle state. This is why it is about a kind of "whiplash effect" which is drafted in Figure 2.
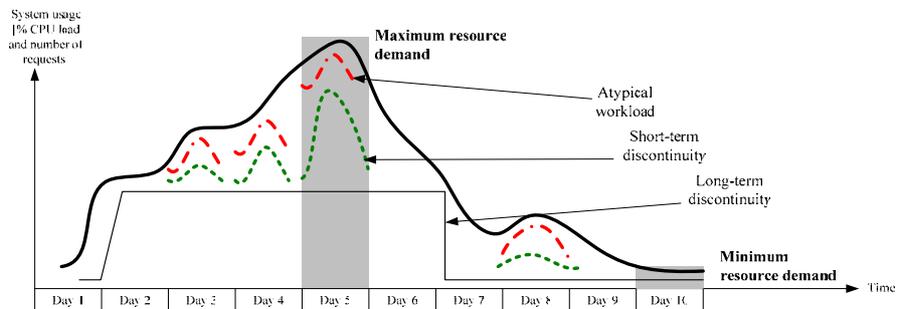
Figure 2: The Whiplash Effect (draft)

As consequence, the underlying infrastructure has to be able to handle both the discontinuous and atypical workloads which themselves cause for example

- higher average response time during customers' courses

- higher duration of administrative tasks, e.g. client copies and reinstallations

One way to face such customer access patterns would be **sizing for peak workloads**. But two problems will arise: First, it is hard to predict the maximum workload of every course and second and most important, given a completely equipped computer centre, there are only limited resources available and charging with a maximum of hardware is very cost intensive.

Another solution would be to **flexibly allocate the given resources** the way they provide each educational session at any point of time with sufficient computing power to accomplish the requests.

Resource Provisioning "on Demand"

One of the most critical resources concerning computing power for an SAP system is the application server providing a user-defined number of work processes which deal with any kind of user requests. An application server added to an existing SAP system not only enlarges the number of its work processes but also the system's overall CPU power, shared memory and buffer sizes.

A solution to encounter the problems arising from the special workload patterns would be a flexible and scalable shared (application) server concept. Servers not being used at the moment are grouped in a pool [Andr⁺82, p. 1] and taken out if resources are requested by another SAP system. If a system will need extended resources because of a training lesson taking place the next days, one or more application servers will be connected to the system enlarging its capacity, e.g. the number of its work processes. Assumed that not all systems need extended

resources at the same time, a given number of application servers can enhance different systems dynamically. The servers act as a kind of "migrant worker".

The request for additional resources can be triggered in different ways. E.g. a schedule is specifying the times the resources have to be switched from one SAP system to another. It can happen also job scheduled or at specific performance levels. Definitely, resources are allocated "on demand".

The blade technology concept, e.g. Sun Blade servers [Sun02] is supposed to provide a scalable and modular infrastructure and hence the characteristic of this technology fits to the requirements of the concept of usage on demand.

Establishing an Adaptive Infrastructure with Blade Servers

As the blade technology emerged a few years ago, a new concept of servers came up. A blade is a slim, hotswappable server which is plugged into a chassis that can hold a number of blades. Each blade server is an independent server with one or more processors and associated memory, disk storage, network controllers and running its own operating system and applications. Fewer components than in the conventional server design are necessary because the chassis provides the servers with power, fans, floppy drives, switches and network ports which are shared with other blade servers. This results in the reduction of cabling and infrastructure costs, power consumption and in less use of data center space [Sun02]. According to [Scha04], blade servers are only about fourth the price of one unit servers with an equivalent capacity.

The modular design of a blade server based architecture allows enterprises to scale their IT infrastructure very precisely and also afterwards, after the initial sizing process. There is no need to buy powerful servers and only use part of their computing power or even let them run idle for most of the time. The administrators of data centers can buy what they need today and plug in additional blades when their processing needs increase. Since the sizing of a SAP system is not trivial, the blade technology enables the SAP systems to be scaled more accurate.

Another advantage of blade servers is the central system management which allows the administrator to remote install or configure software on blades. Critical operating parameters like temperature and fans can be also monitored via web client.

In the context of an SAP system landscape, each blade server is suited to run as an independent application server. The design and the characteristics of the blade technology enable this technology to serve as hardware foundation of a concept supporting flexible resource allocation.

Load balancing in SAP systems

The application layer of each SAP system consists of one or a cluster of several SAP instances[1] which normally run on different application server hosts. When a user connects to the system, he is assigned to a specific application server which processes all the incoming requests of that user. For each SAP system there is one message server configured, which handles the load balancing (Figure 3). This message server redirects the logon request of the user to the application server that currently faces the smallest workload. As consequence, each user is logged on to one application server for the time of the whole session.

Each application server runs a dispatcher and a number of work processes. Any dialog user request is split into dialog steps. Each dialog step is assigned to a single work process for execution. The dispatcher distributes the dialog steps among the work processes on the application server. Different dialog steps of the same user request don't have to be executed on the same work processes; however the assignment of the work process has to be set for every single dialog step ("dialog work process multiplexing") [Schn04, p. 60].

SAP systems allow system wide logon load balancing. But once a user is logged on to an application server there is no further load balancing between different application servers.

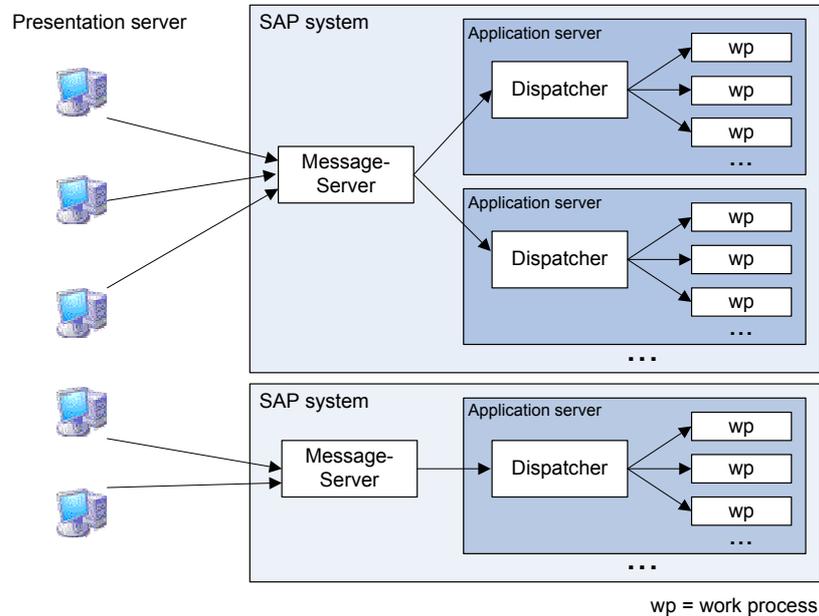[1] The highest number of application servers at customer installations is less than 30 [Rade01].

Figure 3: Load balancing in SAP systems

Logon load balancing also allows the system to handle breakdowns of single servers: In case an application server breaks down, the message server redirects the incoming logon to another application server within the SAP system cluster. Additional application servers can be installed to support a system with additional computing power.

Despite the fact that the SAP system, once it is installed, can dynamically distribute the incoming requests to application servers and their work processes, the overall technical design of the system itself is fixed. Incoming user requests cannot be distributed among application servers of different SAP systems. During the installation of an application server the specific SAP system the server will work for has to be set. It is not possible to switch the server to another system dynamically, even if there is no work to do and the server could support another SAP system which currently is under high work load.

Flexible Server Assignment within a Fixed Software Architecture

Script based application server assignment

Although, as described above, the application server architecture of SAP systems is fixed after the initial configuration procedure, there is a technique that allows an SAP application server running on a blade host to switch between several SAP systems dynamically within minutes. This technique is called *Blade Runner*[2] and it is based on a script which runs on operating system level on a blade server. Setup is easy and only small software packages have to be installed to configure a blade to run with blade runner.

The execution of blade runner starts an application server (= instance) which then connects to a user chosen SAP system. The script copies the settings of another template application server and sets up the structures the new application server needs. The template server delivers an image for the new SAP instance. This image includes file structure, configuration files for the database connection and SAP profiles. The profiles of the starting application server can even be custommized, e.g. by defining the number of different kinds of work processes. As consequence, the services of an SAP system (e.g. work processes) are not statically bound to a dedicated server. A service could run on any of the currently active application servers. Once the application server is started it acts like a normal SAP instance. If another SAP system needs the computing power of this server, the blade runner script logs off the server and attaches it to the requesting SAP system.

Advantages through flexible application server assignment

This simple scripting technique enables application servers to run for different SAP systems. In the context of service providing for educational institutions, an SAP system's **capacity can be adapted** to the specific course demands which may help managing the "whiplash effect" described above. This will result in a better system performance.

Whereas users gain in performance, the application service provider will experience an increase in flexibility. He can deal with **server breakdowns** much

---

[2] *Blade Runner* is a bundle of UNIX Solaris bash scripts and SAP ABAP™ programs developped by the Chair for Information Systems at the Technische Universität München.

more easily and faster, because a dynamic blade can be assigned to another system within less than five minutes.

The blade switching concept will be of economic interest especially in a situation where there is a big amount of SAP systems to be "supported" by a given amount of blade servers. The switching concept enables the pooled blade servers to work for different SAP systems. It enables a fast redesign of the overall SAP system landscape in order to establish a kind of **data center wide load balancing**. Assumed that not all systems need extended computing power at the same time, the service provider can run a given number of SAP systems at a given service level with less capital investments in hardware than in a situation without flexible server assignment. But, in the case of ESP, there are some preconditions:

1. The lecture periods and holiday times during a year don't overlap completely. In case the ESP's customer base is heterogeneous (e.g. universities, universities of applied sciences, schools) there's a chance that not all customers go on leave at the same time.

2. During lecture periods, not all customers conduct their courses simultaneously. There is no good reason to presume an uneven spreading of educational courses while the university is in session.

Table 2 gives an overview about the advantages mentioned above and their practical implications:

Advantage	Consequences
adaptation of one SAP system's capacity	ability to handle peak workloads, better performance
fast substitution of broken servers	less server down times
data center wide load balancing	less hardware investments necessary

Table 2: Advantages and consequences of a flexible server assignment

Case Study: HCC at Technische Universität München

The HCC (Hochschulkompetenzzentrum - University Competence Center) at Technische Universität München is an Application Service Provider (ASP) hosting more than 45 independent SAP Systems for about 40 german institutions of higher education, e.g. universities and universities of applied sciences. Those institutions include SAP applications into their curricula: a variety of courses is based on SAP software, e.g. lectures about customizing and seminars dealing with ABAP™ programming or data warehousing.

Current situation at the HCC data center

The HCC's data centre is equipped with about 40 Sun Fire V210 servers and about 96 Sun Fire B100S Blade servers. Actually (as of June 2004) each SAP system consists of a cluster of three application servers: the central instance and the database instance are running on a Sun Fire V210 server whereas the two dialog instances are running on two Sun Fire B100S Blade servers. Each V210 is connected to a Sun StoreEdge 3310 SCSI array. The dialog instances are statically assigned to their specific central instance because SAP's architecture requires so. The HCC faces a kind of dilemma: **Running a static software on a flexible hardware**: So far, the HCC does not use the blade infrastructure as originally intended. Consequently, the systems cannot adapt to changing workload situations and workload differences between the several systems cannot be equalized.

Future configuration plans: Blade Switching

The concept of flexible application server assignment using *Blade Runner* promises to appease the current dilemma. If some of the existing blades are turned flexible and put into the blade pool, the HCC hopefully can

- handle peak workload situations (during educational courses of individual customers) on single SAP systems

- easily and fast react to server breakdowns and

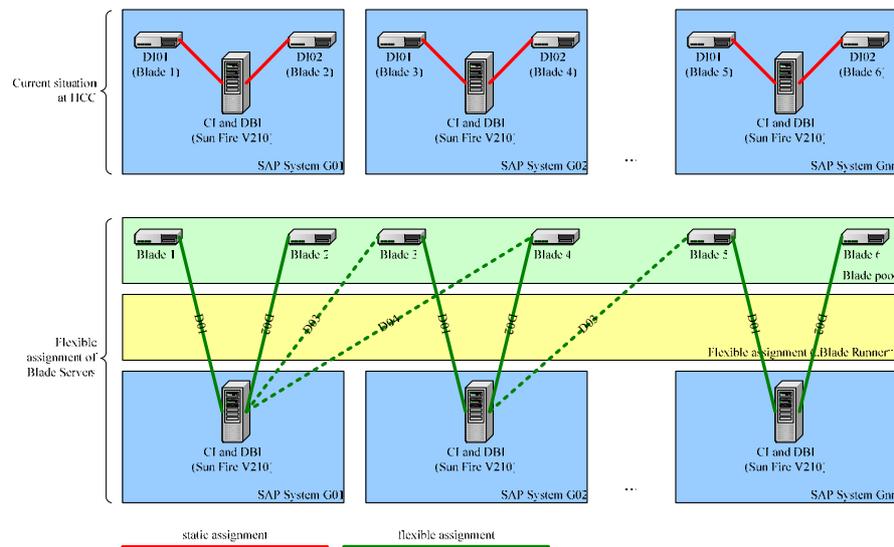- establish a data center wide load balancing between the 45 SAP systems.

Figure 4: HCC's SAP system landscape architecture alternatives

Flexible infrastructure: Measuring benefits and limits

The HCC will only take into consideration the flexible infrastructure concept if it yields an economic benefit. Since the project is still in an early phase the performance enhancement was only measured under conditions which are not yet close to course reality, i.e. on few test servers over a limited period of weeks. This is why, in the next phase, we are going to extend the time frame and propose to measure the benefits with the following test scenarios:

1. Predict the HCC customers' system usage behavior (temporal distribution of workload). Next, put this behavior into a statistical model. Then use this model to perform an **ex-ante-simulation** of possible data center wide load balancing using the *Blade Runner* concept.

2. After a period of about three months running the data center with the "old" static blade architecture measure a predefined set of important values (see below), in the first instance referring to a test environment, i.e. 1 or 2 SAP systems. An ecatt macro will produce the necessary test workload. Then implement the flexible blade concept on a few blade servers and replay the test run, again measuring the defined values. This **ex-post-measurement** will reveal enhancements in data center wide performance, if existent. In a second phase, the test environment will be enlarged to more SAP systems.

The values to be measured first of all include **average response time** as defined in [Schn02], because this is the main indicator for a system's performance from a user's point of view. For the data center wide average response time is to be optimized, the individual system average response times have to be aggregated, e.g. including a weighting. From the HCC's point of view a shorter **duration of client copies and reinstallations** would make the administrators' life easier because at an average there are about 3-4 client copies to be conducted per month. Flexibly assigned application servers are supposed to help reduce duration.

But to what extend can additional blade based application servers improve an SAP system's capacity before another component becomes a bottleneck ? Each new application server can handle more users and more requests which will produce an extended amount of database transactions and increased network traffic. In another test series, we propose to examine the **delimiting factors** like database host and network connections when adding another application server to an existing cluster.

Portability considerations

The HCC currently hosts SAP R/3 Enterprise systems, BW 3.10 systems and one IS-Banking 4.6C system. The blade runner script has been developed for this configuration but the concept itself can be applied to every kind of Web

Application Server (Web AS) based SAP systems. Furthermore, it is not restricted to application servers installed on blade servers. However, the blade technology supports the creation of a modular and scalable system landscape better than big one-unit hosts.

Productive, non education dedicated SAP systems usually don't experience such atypical and inconstant workload. But there are some examples where they do so: year-end or month-end closing and periods of time where SAP systems have to be customized, tested or migrated. The blade runner technique would equip these systems during times of higher workload with additional computing power to still provide enough hardware and computing resources to handle the incoming requests.

Conclusions

The flexible resource allocation concept we proposed in this paper is an attempt to solve the area of conflict between running a static software system (considering SAP systems as an example) and the efforts of embedding it into an adaptive hardware architecture (i.e. blade servers). We found out that this won't be a real problem because with relatively little effort on the part of the software application the latter can be turned flexible. In the case of providing SAP systems for educational purposes (ESP model) this opens up new potentials for making available a lot of SAP training systems with limited hardware resources.

References

[Andr+82] Andrews, G. R.; Dobkin, D. P.; Downey, P.J.: Distributed allocation with pools of servers. Annual ACM Symposium on Principles of Distributed Computing, Ottawa, Canada, 1982.

[Bece+00] Becerra-Fernandez, I.; Murphy, K. E.; Simon, S.J.: Enterprise resource planning: integrating ERP in the business school curriculum. Communications of the ACM 43(4), 2000, pp. 39-41.

[Dave98] Davenport, T. H.: Putting the Enterprise into the Enterprise System. Harvard Business Review July-August 1998, 1998, pp. 121-131.

[GrMa02] Greschler, D.; Mangan, T.: Networking lessons in delivering 'Software as a Service' - Part I. International Journal of Network Management 12, 2002, pp. 317-321.

[Krcm03] Krcmar, H.: Informationsmanagement. Springer: Berlin, et al., 2003.

[Lang03] Langley, N.: Demand for SAP skills on the rise. ComputerWeekly.com. 2003.

[Rade01] Rademann, C.: Introduction into mySAP.com Landscapes, 2001.

[Scha04] Scharf, A.: Blade-Server gewinnen an Akzeptanz. Monitor. 4a, 2004.

[Schn02] Schneider, T.: SAP Performance Optimization Guide, SAP Press, 2002.

[Schn04] Schneider-Neureither, A.: Optimierung von SAP-Systemlandschaften, SAP Press, 2004.

[Sun02] Sun: Improving Economies of Scale with Blade Computing, 2002.

[Wals03] Walsh, K. R.: Analyzing the application ASP concept: technologies, economies, and strategies. Communications of the ACM 46(8), 2003, pp. 103-107.

[WaSc99] Watson, E. E.; Schneider, H.: Using ERP systems in education.. Communications of the ACM 1, 1999.