Capacity Planning using Enterprise Performance Vision™ for z/OS

White Paper

Version 1.0

Demand Technology Software

September 2004

INDEX

1. INTRODUCTION 3

2. REQUIRED TOOLS 3

3. METHODOLOGICAL APPROACH 4

3.1. Assumptions 4

3.2. Defining the “baseline” 4

3.3. Evaluating the CPU usage natural growth 7

3.4. Evaluating the resources correlation indexes tendency 9

3.5. Evaluating the planned growth 10

3.6. Forecasting the CPU demand 13

3.7. Forecasting the other resources demands 13

4. SUMMARY 14

1. Introduction

From the various disciplines of Capacity Management, Capacity Planning has always been the most unloved. All companies manage Service Levels in different ways, and consequently the performance of their systems and applications; most of them have Cost Accounting and Chargeback procedures, but only some of them have a specific Capacity Planning activity running regularly.

The major reasons for this situation are the following:

- Capacity Planning is the only Capacity Management discipline that doesn't attain only to the past and present, but also to the future of systems and applications;
- even when based on metrics and methodologies, the forecast of something that will happen in the future is often affected by significant errors ;
- the accuracy of the result is highly dependent on the skill and experience of those that make the plan; these skills are difficult to find and very expensive;
- specific tools for Capacity Planning are generally very expensive too, and not really user friendly;
- Capacity Planning projects require long periods of time before some result can be produced.

Even with all these difficulties, Capacity Planning activities are always in someway performed, also in those organisations where there isn't a specific group responsible for the discipline, by:

- system programmers, based on their judgement on critical issues to come and avoid (more often already existing);
- managers, with respect to their budget;
- vendors, sometimes more keen to their own goals and objectives.

The systems programmers approach could be the most valid from the technical point of view, but not really effective because it's a based on a reactive approach and on the analysis of sporadically collected data.

In the following we will introduce a cost effective Capacity Planning methodology that is designed to guide the user to perform rapid Capacity Planning studies, reducing forecasting errors and solving most of the problems mentioned above.

2. Required tools

Most of the information needed to perform a Capacity Planning study is already available in the Enterprise Performance Vision for z/OS (EPV) product.
You only need a spreadsheet; EPV is natively integrated with MS-Excel, in addition one can export the information anywhere by using the browsers copy and paste functionalities.

3. Methodological approach

3.1. Assumptions

The Capacity Planning methodology we suggest is aimed to discover the amount of capacity (CPU) needed to support the planned workloads; indications on other critical resources such as memory, I/O and disk space can be obtained by tracking the evolution of a set of indexes that correlates the utilisation of these resources among them.

This methodology is based on three major assumptions:

- the most important resources of a system (CPU, memory, I/O, disk space....) must be balanced to allow a system to work at its best;
- the correlation between these resources will vary in time, depending on the architectural and software evolutions;
- in a specific point in time the ratios between the resources are meaningful also for new comparable applications.

We don't model the response time of the applications, because we think that if the resources were planned correctly the response time should be affected mostly by the applications and systems tuning efforts and will not be a Capacity Planning issue..

The methodology is based on the following steps:

Step 1: Definition of a "baseline"
Step 2: Evaluation of the natural growth of CPU usage
Step 3: Evaluation of the tendency of the correlation indexes between resources
Step 4: Evaluation of the planned growth
Step 5: CPU demand forecasting
Step 6: Other resources demands forecasting

3.2. Defining the "baseline"

For each system you need to find a starting point (baseline) which best represents the CPU usage level; the natural and planned growth should be applied to the baseline to estimate the CPU demand.

In order to do that you have to analyse a period long enough to be meaningful and representative for the "mission critical" workloads running in the system.
It is also recommended that you choose a recent period that contains the latest optimisations performed on your systems and applications (or a specific peak period).

Depending on the characteristics of your workloads, you may find it useful to exclude Saturdays, Sundays, holidays or shifts.

The method used to identify the value that best represents each system is based on the analysis of the average hourly CPU utilisation in MIPS. This data, sorted by descending MIPS utilisation, is

provided in the **SYSTEM PERCENTILE STATISTICS** view included in the **SYSTEM DAY TREND VISION** module.
The following table (Table 1) is an example of this view.

PERCENTILE OVERVIEW						
SYSTEM	PERCENTILE	%CPU	MIPS	EXCPDASD/SECS	EXCPTAPE/SECS	READYAVG
SYSA	100	63	981	9.491	1.344	5,2
SYSA	99	47	737	6.686	1.068	4,0
SYSA	98	44	681	5.776	919	3,6
SYSA	97	42	652	5.353	827	3,5
SYSA	96	41	631	4.596	765	3,4
…..	…….	……..	……..	…….	…….	…….
SYSA	2	9	137	245	0	1,4
SYSA	1	7	108	203	0	1,3

Table 1

The data should be graphically represented through a scatter plot, as in the following example.

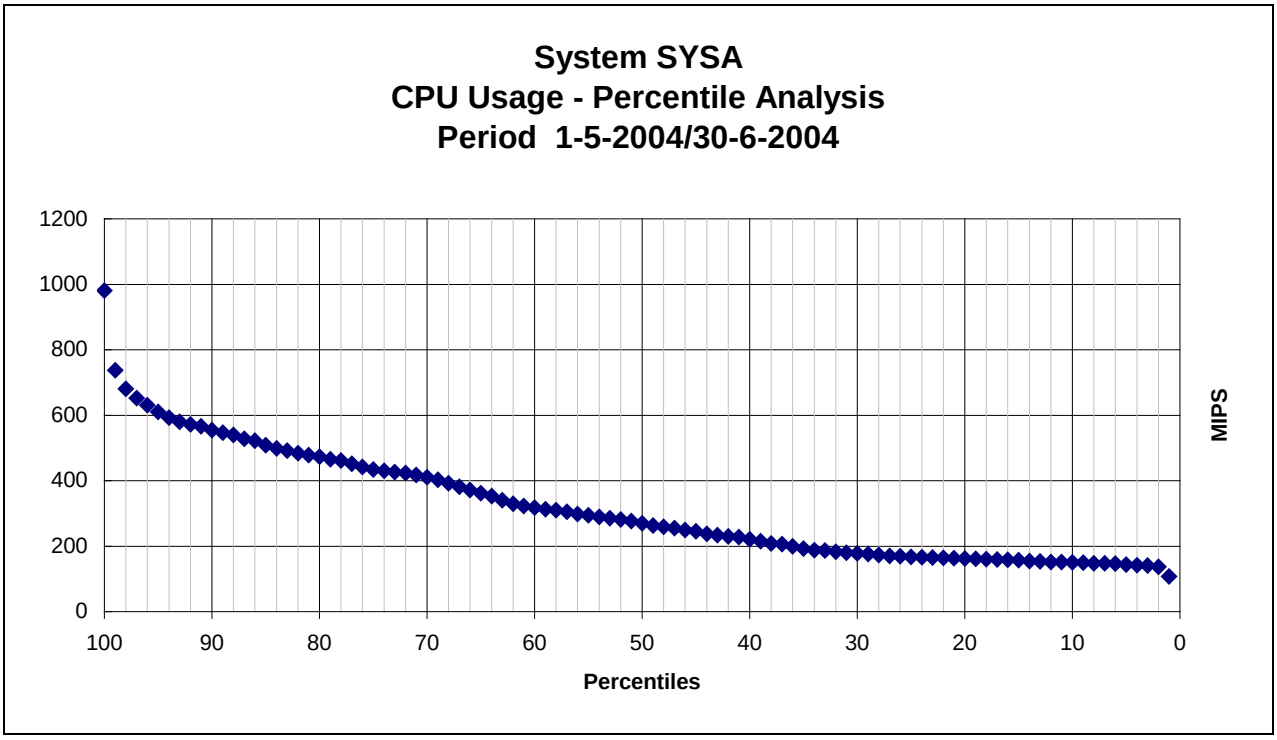


Figure 1

The choice of using a percentile as the baseline has a clear and practical meaning; this is the level of tolerance you're willing to accept in respect to the system saturation.
If you choose the 95th percentile it means that you're accepting a situation where applications may suffer from insufficient CPU in 5% of the hours in the analysed period.

In the example in Figure 1 we've a peak value about 1.000 MIPS when at 95th percentile the CPU utilisation is only about 600 MIPS.

A different example is shown in Figure 2, where there's more homogeneity in the consumption; the peak value is about 2.100 MIPS and the 95th percentile is about 1.800 MIPS.

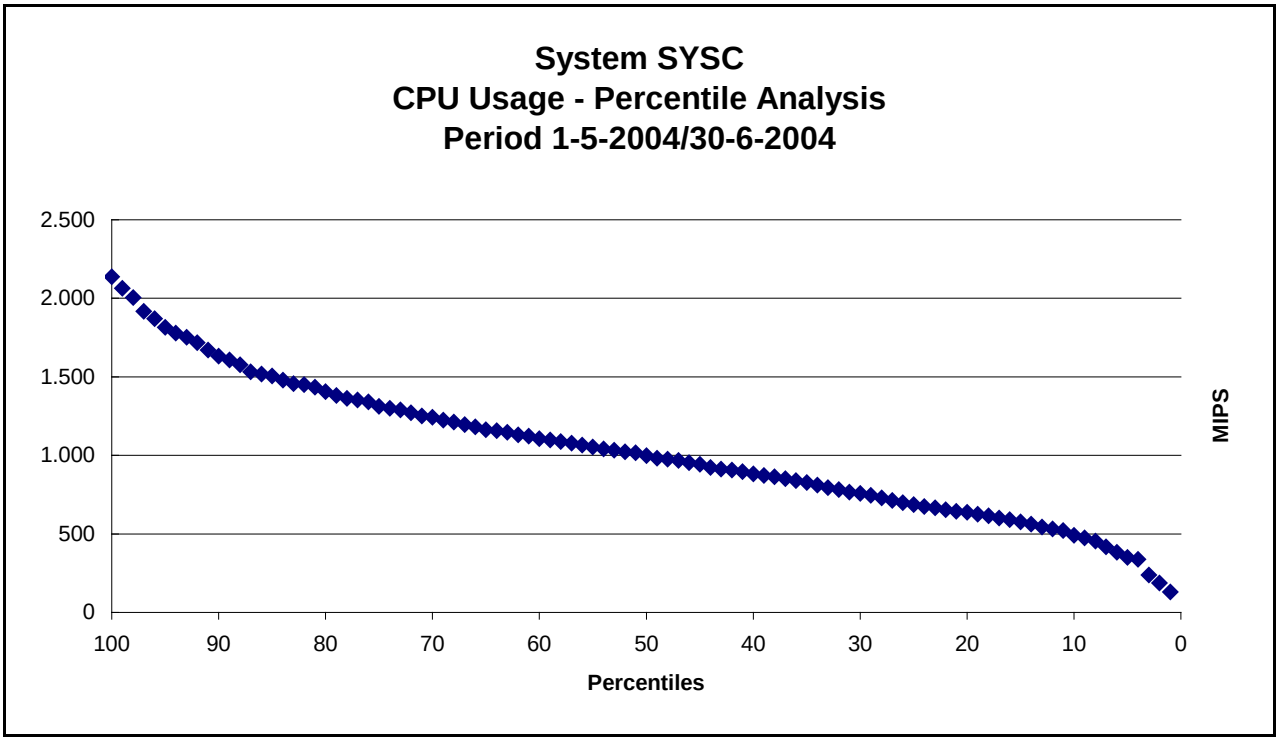


Figure 2

We know that z/OS manages the priorities of various workloads very efficiently in order to provide adequate service levels to mission critical applications even at 100% CPU utilisation. In order to choose the best percentile you must understand the workload composition in the analysed timeframe. EPV provides many views that can be used for this kind of analysis (both at daily and monthly level.[1]).

The curve analysis allows the verification of an eventual latent demand, which is normally showed when the first part of the curve is parallel to the Y axe and is close to the maximum available CPU capacity on the analysed system (see Figure 3).

[1]Workload Vision provides detailed information while the Workload Day Trend Vision and the Workload Month Trend Vision allow the workloads evolution analysis at daily and monthly level .

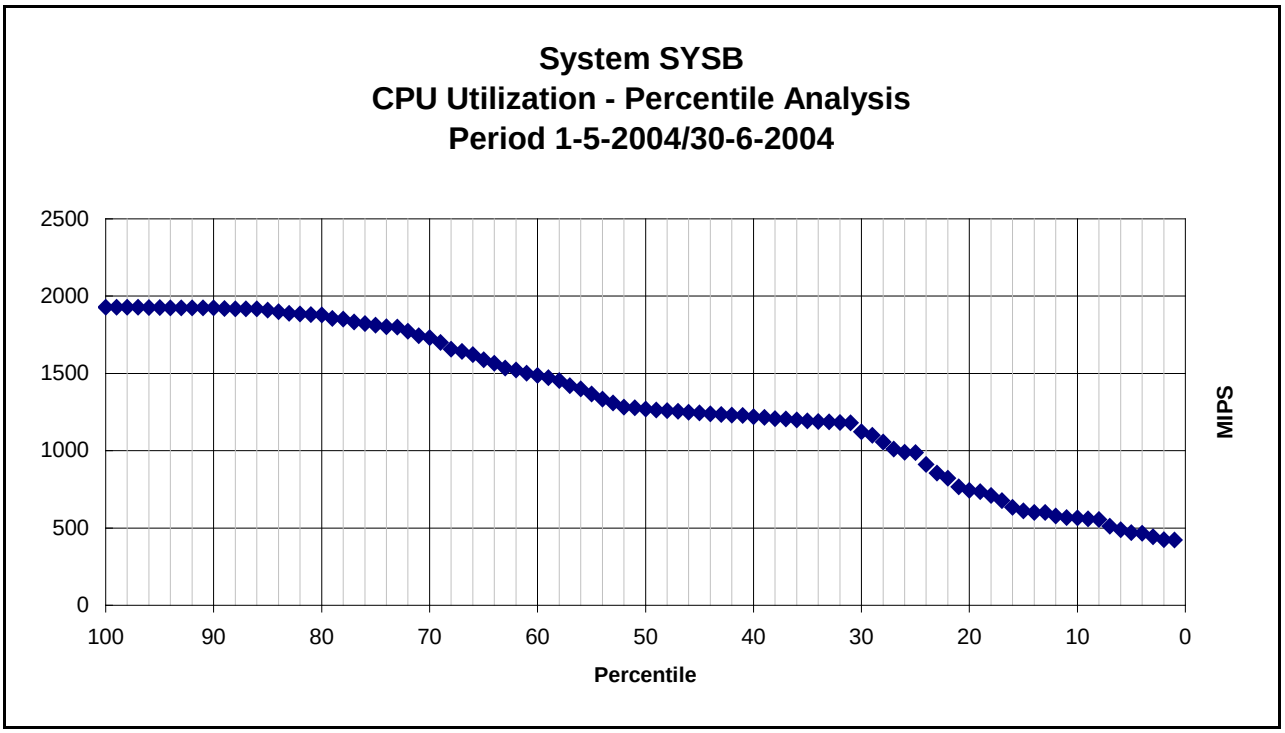


Figure 3

In this example the baseline will be 100% of the capacity plus the estimated latent demand.[2]
The information needed to estimate the latent demand is available in the **SYSTEM MIPS DEMAND DAY PROFILE** view included in the **SYSTEM DAY TREND VISION** module.

The last important thing to remember is the coexistence on the same machine of more production systems with non-corresponding usage peaks. In this situation you should define a baseline for the whole machine while looking at all systems together. This additional analysis is available in the **SYSTEM MIPS DAY PROFILE** view included in the **SYSTEM DAY TREND VISION** module.
Starting from this baseline you should then define partial baselines for each partition.[3]

3.3. Evaluating the CPU usage natural growth

The natural growth is the tendency of every system to use additional capacity over time, even in the absence of new application releases.
This tendency is measurable in most of the production environments, and is due to:

- data base growth;
- workloads growth;
- small application releases or added functionalities to applications already in production;
- software evolution.

[2] Also in this situation we suggest to use the percentile analysis technique to estimate the latent demand.
[3] To establish the specific baseline for each partition you could use the weight assigned to every system or the measured utilisation of each system .

A good metric to evaluate the natural growth is the total CPU service units usage on a monthly level.
Also this metric is available in EPV; Table 2 provides an example of the **TOTAL SERVICE UNITS MONTH TREND** view included in the **SYSTEM MONTH TREND VISION** module.

	CPU Million Service Units						
	SYSTEM						
DATE	SYSA	SYSB	SYSC	SYSD	SYSE	SYSF	TOTAL
jul-04	31.132	133.100	109.989	73.000	27.778	16.999	391.998
jun-04	32.367	133.860	111.882	71.895	25.096	16.761	391.861
may-04	39.688	134.542	116.808	83.434	31.102	14.794	420.368
apr-04	37.814	132.168	107.590	70.573	27.674	14.789	390.608
mar-04	32.725	130.507	123.103	78.012	30.644	11.724	406.715
feb-04	26.370	124.942	110.939	66.124	22.610	8.095	359.080
jan-04	26.088	125.929	102.790	58.151	15.413	11.543	339.914
dec-03	27.017	128.811	99.965	55.671	17.952	10.309	339.725
nov-03	25.481	127.853	98.826	61.284	15.716	7.845	337.005
oct-03	24.669	127.807	98.847	55.756	17.225	6.438	330.742
sep-03	25.053	128.006	101.342	59.008	18.142	7.072	338.623
aug-03	23.368	126.154	69.965	50.683	14.961	6.877	292.008
jul-03	27.272	129.967	89.774	58.524	18.399	7.718	331.654
jun-03	25.436	126.319	89.282	59.969	15.930	9.531	326.467
may-03	18.701	127.246	108.653	63.485	20.101	5.015	343.201
apr-03	17.730	123.503	100.408	56.783	17.815	3.949	320.188

Table 2

For each system a regression analysis needs to be performed on the monthly totals. The difference between the initial value and the last value of the regression, divided by the number of intervals gives you the percentage of the natural monthly growth.

The percentage of natural growth is generally around 1% to 2% each month. Greater values should be investigated as they could hide other things as a release in production of a new application or some major changes in an existing one.

If this is the case you should “clean” the monthly data by subtracting the amount introduced by these changes or you could decide to use a different timeframe where such a phenomena didn’t occur. The aim is to obtain a period without major application changes in order to calculate your system’s natural growth correctly.

Figure 4 reports an example showing the evaluated natural growth for the SYSC system.

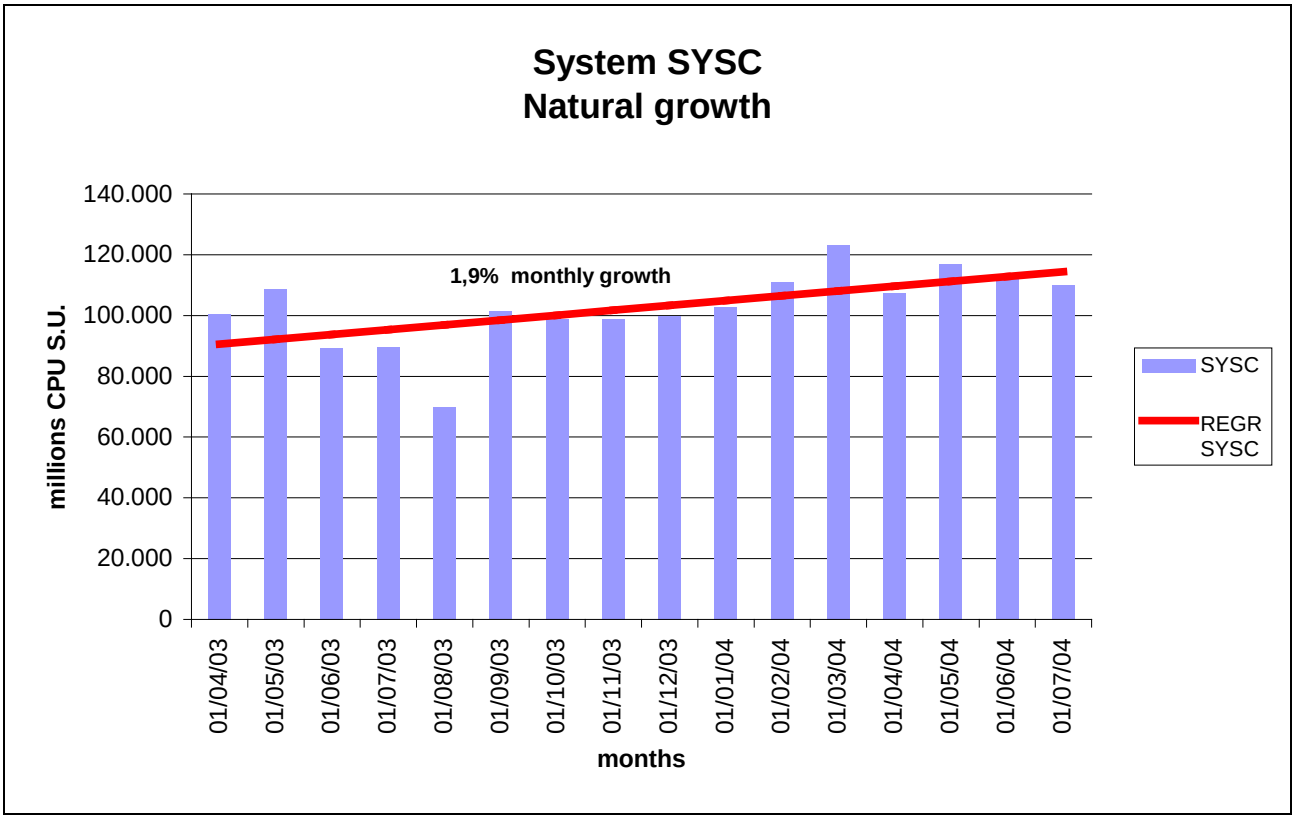


Figure 4

3.4. Evaluating the resources correlation indexes tendency

Based on our previous assumptions (see 3.1), by tracking the ratio between the different resources utilisation over time, it's sufficient to know one of them in order to calculate the others.

Practically, you must track some indexes that are used as coefficients. Depending on your workloads characteristics you should consider to exclude Saturdays, Sundays or shifts.

The first index, called **IOS**, provides the disks access density. It's the ratio between the disk I/O operations per second and the allocated disk space in GB.
As in most cases the disks are shared between the systems, EPV evaluates just one index for all systems.
This index tends to diminish over time because the allocated disk space tends to grow faster then the number of disk I/O operations.

The second index is called **IOC** and provides the ratio between the disk I/O operations per second and the MIPS usage.
This index is evaluated for each system.
This index tends to diminish over time because MIPS usage tends to grow faster then the number of disk I/O operations.

The third index is called **MMC**, and provides the ratio between the memory usage in MB and the MIPS usage . This index is also evaluated for each system.
You've to be aware that the MMC index for small systems (such as development, test or experimentation systems) tends to have greater values that are affected by the minimum memory usage needed by the z/OS operative system itself.
This index tends to grow over time because the memory usage tends to grow faster then CPU usage.

All these indexes must be tracked on a monthly base. These values help us understand the relationship between the different resources and to foresee future evolutions. EPV provides all these indexes in the **IOS MONTH TREND, IOC MONTH TREND** and **MMC MONTH TREND** views included in the **RESOURCE MONTH TREND VISION** module.

Table 3 shows an example of the IOC MONTH TREND view.
Different values among the systems are normally due to their workload characteristics.
Within each system the values are mostly homogeneous.[4]

	IORATE/MIPS					
	SYSTEMS					
DATE	SYSA	SYSB	SYSC	SYSD	SYSE	SYSF
jul-04	2,14	3,17	3,19	3,95	3,40	2,46
jun-04	2,09	3,19	3,06	3,85	3,54	2,58
may-04	1,85	3,18	2,88	3,79	3,39	3,03
apr-04	1,93	3,19	2,88	3,81	3,63	2,37
mar-04	2,11	3,12	3,11	3,85	3,58	2,67
feb-04	2,19	3,20	3,21	3,88	3,71	2,69
jan-04	1,99	3,22	3,01	3,80	3,73	2,49

Table 3

3.5. Evaluating the planned growth

The planned growth is the portion of growth that you can estimate, in addition or in subtraction to the current load, due to:

- current applications modifications;
- release of new applications;
- significant variations in current workloads;
- technological and architectural evolution.

[4] The current value of the index can be obtained through the average values of the last 3-4 months. To estimate the tendency in the long term you should perform a regression analysis using at least 24 months of data.

The first item also includes optimisation activities for production applications.

The second item is the most relevant and potentially the one that can introduce the biggest errors in Capacity Planning studies.

To minimise these errors you should follow this logical flow:

- define the forecasting period;
- identify the applications that will be released in this period;
- identify for each application: the production environment where it will run, dates and percentage of release on each date;
- estimate together with the development teams the amount of disk space (MB) needed until the end of the forecasting period.

We use the disk space as a starting point because it is generally the only metric that the development team can forecast rather closely to reality, this will help us avoid the biggest errors in our plan. In fact it's very uncommon to be able to estimate a reasonable CPU consumption estimate for applications still in the development phase. The disk space estimate should be checked as soon as possible with accurate tests. Using the disk space estimate, we can calculate, through our indexes, the number of I/O operations that the application will produce, and then the application MIPS usage.[5]

The following tables provide an example of the methodology described above:

New Applications	Production environment	Date of release	Percentage of release	GB
Application A	SYSB	09/04 12/04 03/05 06/05 09/05	40% 10% 10% 10% 30%	400
Application B	SYSC	09/04 02/05	50% 50%	200
Application C	SYSA	12/04	100%	100
Application D	SYSC	11/04	100%	100
Application E	SYSC	10/04	100%	300

Table 4

The next thing to evaluate is the amount of disk space used at the end of the forecasting period.
To simplify our example we will estimate that 50% of the disk space will be used.
The amount of used GB disk space can then be multiplied by the index IOS estimated in the last 4 months.

[5] Be aware that the amount of space indicated by the application teams is normally only the part strictly related to the application operational data; you should also consider additional space needed for fragmentation, free space, backups, flash copy, Disaster Recovery, ….

	IORATE/DASD GBYTE
	SYSTEMS
DATE	TOTAL
jul-04	1,18
jun-04	1,14
may-04	1,25
apr-04	1,24
average	**1,20**

Table 5

The result shown in Table 5 is disk access density; by multiplying the IOS index average by the used GB it's possible to calculate the number of disk I/O operations per second estimated for the new applications.
For each system hosting the new applications you must then estimate the value of the IOC index. Also for this estimate you should use the average of the last 4 months (see Table 6).

	IORATE/MIPS					
	SYSTEMS					
DATE	SYSA	SYSB	SYSC	SYSD	SYSE	SYSF
jul-04	2,14	3,17	3,19	3,95	3,40	2,46
jun-04	2,09	3,19	3,06	3,85	3,54	2,58
may-04	1,85	3,18	2,88	3,79	3,39	3,03
apr-04	1,93	3,19	2,88	3,81	3,63	2,37
average	**2,00**	**3,18**	**3,00**			

Table 6

By dividing the IOC index average by the number of I/O operations per second you will obtain an estimate of the MIPS needed by the new applications.
A summary is reported in Table 7.

New Applications	Production environment	Date of release	Percentage of release	GB	Used GB	IOS	I/O sec	IOC	MIPS
Application A	SYSB	09/04 12/04 03/05 06/05 09/05	40% 10% 10% 10% 30%	400	200	1,2	240	3,18	**75**
Application B	SYSC	09/04 02/05	50% 50%	200	100	1,2	120	3,00	**40**
Application C	SYSA	12/04	100%	100	50	1,2	60	2,00	**30**
Application D	SYSC	11/04	100%	100	50	1,2	60	3,00	**20**
Application E	SYSC	10/04	100%	300	150	1,2	180	3,00	**60**

Table 7

For each system you must build a table which indicates the amount of estimated MIPS for each new application and release dates. Table 8 shows an example of the SYSC system forecasting.

	…	09-04	10-04	11-04	12-04	01-05	02-05	…	Total by Application
Application B		20					20		40
Application D				20					20
Application E			60						60
monthly total		20	60	20			20		120

Table 8

3.6. Forecasting the CPU demand

For each CPC you should build a scenario reporting the estimated MIPS usage, up to the end of the forecasting period.

The starting point for each partition is the baseline, increased with the eventual latent demand.
For each month you have to add the planned growth effects, adding or detracting values depending on your estimates (new loads, or instead optimisations).

You must not forget to add the monthly natural growth for each partition.

In addition we recommend adding a "capacity reserve" to the total CPU demand for the CPC to account for the following issues:

- the baseline estimate is based on hourly averages; you cannot predict short peaks of load, that will happen in real life;
- online systems performances start to degrade when the hourly average of the CPU usage becomes greater than 85%;
- each measurement is affected by errors, and this phenomenon is amplified when you forecast the future.

Attachment 1 provides an example of a spreadsheet used to forecast the CPU usage.

3.7. Forecasting the other resources demands

By multiplying the evaluated MIPS values by the index MMC you can estimate the memory usage that you should consider for each partition.
This estimate doesn't account for the logically swapped address spaces, so you should add some more memory to warrant good performances to your systems and applications.
How much you should add is strictly related to the workload running in each partition; it also could be a relevant value for those systems running mostly TSO and Batch applications.

By multiplying the evaluated MIPS values by the index IOC you can also estimate the total I/O per second load on each system.
This value together with other metrics available in the EPV pages, such as connect and disconnect time, write hits, etc, can allow us to estimate the number of ESCON or FICON channels needed.

The sum of I/O per second of all the partitions multiplied by the IOS index will provide an estimate of the total disk space that will be used.
The resulting value must be incremented by dividing it by the allocated over the installed disk space ratio (measured or desired).

The disk space information together with the number of channels and the I/O characteristics can help us to size the disk Storage Processors.

4. Summary

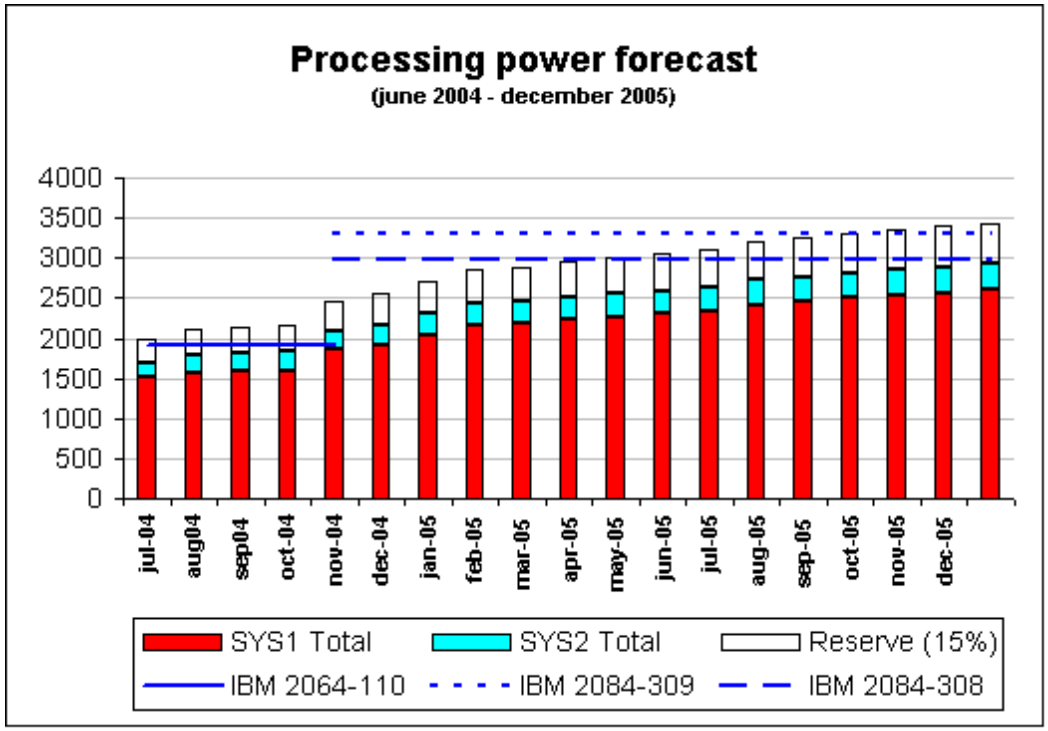
Capacity Planning is a discipline that requires the highest level of knowledge and experience in Capacity Management areas.

The usage of a clear and defined methodology as the one suggested in this paper can help plan the growth of real life systems normally more complex than the ones showed in our examples..

This methodology has been successfully used during the last 10 years, to plan the evolution of complex mainframe environments at different customer sites.
In the last 2 years all the needed views have been introduced in the EPV product and, thanks to that, it has been possible to greatly reduce the efforts needed and the duration of our Capacity Planning studies.

Attachment 1

SYS1 Baseline	1.464
SYS1 Natural Growth	1,3%
SYS1 Latent Demand	100
SYS2 Baseline	169
SYS2 Natural Growth	1,6%
SYS2 Latent Demand	0



	jun-04	jul-04	aug04	sep04	oct-04	nov-04	dec-04	jan-05	feb-05	mar-05	apr-05	may-05	jun-05	jul-05	aug-05	sep-05	oct-05	nov-05	dec-05
SYS1	1564	1518	1562	1582	1602	1859	1905	2038	2152	2180	2227	2269	2299	2347	2420	2452	2498	2530	2563
SYS1 Appl Tuning	-65					-35													
SYS1 New Appl		23			233	56	107	87		18	14		18	43		14			
SYS1 Total	**1518**	**1562**	**1582**	**1602**	**1859**	**1905**	**2038**	**2152**	**2180**	**2227**	**2269**	**2299**	**2347**	**2420**	**2452**	**2498**	**2530**	**2563**	**2596**
SYS2	169	172	225	229	233	236	265	270	274	278	283	287	292	297	301	306	311	316	321
SYS2 Appl Tuning																			
SYS2 New Appl		50				25													
SYS2 Total	**172**	**225**	**229**	**233**	**236**	**265**	**270**	**274**	**278**	**283**	**287**	**292**	**297**	**301**	**306**	**311**	**316**	**321**	**326**
Reserve (15%)	**298**	**315**	**320**	**324**	**370**	**383**	**407**	**428**	**434**	**443**	**451**	**457**	**466**	**480**	**487**	**496**	**502**	**509**	**516**
Total	1988	2102	2130	2159	2465	2553	2715	2855	2893	2952	3008	3048	3110	3202	3245	3304	3348	3393	3438
IBM 2064-110	**1915**	**1915**	**1915**	**1915**	**1915**														
IBM 2084-309					**3299**	**3299**	**3299**	**3299**	**3299**	**3299**	**3299**	**3299**	**3299**	**3299**	**3299**	**3299**	**3299**	**3299**	**3299**
IBM 2084-308					**2993**	**2993**	**2993**	**2993**	**2993**	**2993**	**2993**	**2993**	**2993**	**2993**	**2993**	**2993**	**2993**	**2993**	**2993**