



# Redpaper

Frans Buijsen  
Dave Jewell  
Andrew McDonald  
Bernie Michalik  
William Portal  
Elisabeth Stahl

Chris Cooper  
Martin Jowett  
Richard Metzger  
Mattias Persson  
Richard Raszka  
Damian Towler  
Klaus Waltmann

## Performance and Capacity Implications for a Smarter Planet

IBM® uses the phrase *Smarter Planet*™ to describe its vision for the future, in which the world's systems become increasingly instrumented, interconnected, and infused with intelligence to better manage the environments in which we work, play, and live.

Real examples of current and emerging technologies exist. These technologies are starting to enable "smarter" solutions for vital domains, such as energy, transportation, food production, conservation, finance, and health care. For such an ambitious vision to become a reality, there are many technical challenges to address, among them being how to manage performance and capacity for the supporting IT systems.

This IBM Redpaper™ publication discusses the performance and capacity implications of these solutions. It examines the Smarter Planet vision, the characteristics of these solutions (including the challenges), examples of addressing performance and capacity in a number of recent Smarter IT projects, recommendations from what has been learned thus far, and discussions of what the future may hold for these solutions.

# Introduction to a Smarter Planet

Our cities and our world are complex systems that require a profound shift in management and governance toward far more collaborative approaches. With Smarter Planet, intelligence is being infused into the systems and processes that make the world work. Leaders in business, government, and society around the world are beginning to grasp the potential of intelligent systems to facilitate economic growth, improved efficiency, sustainable development, and societal progress. Building a smarter planet relies on the ability to:

- ▶ Instrument the world's systems
- ▶ Interconnect them
- ▶ Make them intelligent

Some examples of Smarter Planet applications include:

<b>IBM Smarter Energy™</b>	Smarter grids use sensors, meters, digital controls, and analytic tools to automate, monitor and control the flow of energy across operations from power plant to plug.
<b>IBM Smarter Banking™</b>	By applying unprecedented computing power to perform advanced analytics using technology such as stream computing and deep computing visualization, credit and risk evaluators can rely on actionable insight and intelligence.
<b>IBM Smarter Healthcare™</b>	An intelligent approach to health care is one that uses existing information to gain new insights into patient care and organizational performance. Health care providers, researchers, and directors can work smarter by creating and analyzing comprehensive and holistic views of patient data.
<b>IBM Smarter Cities™</b>	Smarter education, smarter health care, smarter water and energy use, smarter public safety, smarter transportation, and smarter government are all part of the smarter city.

These “Smarter” solutions exploit the reality that information technology (IT) is also changing the way it is accessed, the way it is applied, and the way it is architected. As a result, the opportunities for innovation have never been greater. Enterprises in every industry can use breakthroughs in technology to create new business models, devise new ways of delivering technology-based services, and generate new insights from IT to fuel innovation and dramatically improve the economics of IT. New technology innovations signal that we are entering a new era of computing. Smarter Computing integrates technical innovation areas such as big data, optimized systems, and cloud computing to evolve the computing model. Even with Smarter Computing, for such an ambitious vision to become a reality, there are many technical challenges to address, among them how to manage performance and capacity for the supporting IT systems.

This paper considers the following questions.

- ▶ What are some of the performance and capacity challenges posed by Smarter Planet solutions, and how do we propose to manage performance and capacity risks?
- ▶ What have we been able to learn from recent Smarter Planet projects about addressing these challenges, and what recommendations follow from what we have learned?
- ▶ What does Smarter Planet mean for the future of managing IT performance and capacity?

# **Performance and capacity challenges introduced by Smarter Planet**

Industry by industry, process by process, a smarter planet, while global by definition, happens on the industry level. It is driven by forward-thinking organizations that share a common outlook: They see change as an opportunity, and they act on possibilities, not just react to problems.

Since introducing the Smarter Planet concept, IBM has collaborated with more than 600 organizations worldwide who have each taken their place in making this vision a reality.

Data is being captured today as never before. It reveals everything from large and systemic patterns (of global markets, workflows, national infrastructures, and natural systems) to the location, temperature, security, and condition of every item in a global supply chain. Through social media, billions of customers, citizens, students, and patients tell us what they think, what they like and want, and what they are witnessing, in real time.

Intelligence, not intuition, drives innovation. But data by itself is not useful. The most important aspect of smarter systems is the actionable insights that data can reveal.

For example, an endemic weakness in almost all IT solutions and their ongoing upkeep is the availability of timely, accurate, and complete workload volumetrics from the user right down to the individual IT solution components. Even if throughput measurements are available for some IT components, the correlation of these with patterns of business activity (PBAs) usually requires substantial and manual human effort. Anyone responsible for IT billing and chargeback on a “user pays” basis will be familiar with the complexity of this challenge.

## **Workload management**

As the planet gets smarter and becomes the “internet of things”, we need to move towards smarter workload management and performance assurance paradigms that are automated and policy based. This is not a new idea. From the beginning, IT systems have been self-instrumenting and self-governing in terms of resource allocation and performance optimization. Operating system mechanisms for the sharing of scarce memory and CPU are prime examples. So, what is different now?

The difference is the “level of smartness”. In the past, almost all IT regimes for workload management and performance assurance have worked at the hardware, firmware, operating system (and hypervisor), and middleware layers. Measurements and adjustments are made at the IT infrastructure level. Any correlation with patterns of business activity (PBAs) and Business Demand Policies (BDPs) required a high degree of bespoke customization to take workload measurements from the business usage level and make adjustments at the IT application and IT infrastructure level.

With the introduction of so many more inter-operating parts that are all generating masses of instrumentation data, we are witnessing the emergence of integrated business level workload measurements and business determined demand policies to drive direct pass-through automation of IT resource allocations to meet business throughput and response time requirements. If there is a temporary and unexpected peak of business activity, we have the automated instrumentation and policies to know which workloads to favor and the means to guarantee priority access of finite IT resources to support the most important business functions.

Most of these solutions have required customized development and tailoring. However, a high degree of reuse of Smarter Planet IT solutions for performance assurance is emerging within and across particular industry domains. In all cases, there is the deployment of some device that gathers critical business level workload throughput data. It may be smartcards in cards on toll ways or water flow meters in an irrigation channel. This data is used in conjunction with policies for workload prioritization to directly allocate and regulate IT resources for optimum performance and capacity utilization.

A smarter planet is forcing us to move beyond the traditional methods of collecting business throughput data, both current and future, by manual means. It is also coercing us to move beyond manual and periodic analysis of this data to pre-determine the allocation of IT resources to meet business throughput and response time needs. There is simply too much data and it changes too dynamically for manual analysis and adjustment.

We must move to a policy based regime with automated collection of business level throughput data that considers pre-determined business demand policies and automatically adjusts IT resource allocation and workload throttling. For capacity provisioning purposes, we need to define utilization thresholds (or “headroom”) based on the lead times for implementing IT infrastructure so that automated alerts are raised when these thresholds are reached. That way, intelligent, timely, and informed decisions can be made for capital intensive IT infrastructure investment that is aligned with business plans for growth and transformation.

In this context, a good reality check for assessing the maturity of Cloud Computing services is to consider the sophistication and granularity of the service to adjust the allocation or throttling of IT resources based on a data feed of business throughput and priority. A primitive form of this variability is practiced by internet service providers when they reduce the bandwidth capability of customers who exceed their data transfer quota.

In a cloud computing scenario, there will always be finite resources at one or several points in the service chain. Aligning service quality with business priority and financial consequences will become essential as billions of interconnected devices compete for resources. We have already seen that in an online competition between real human users and robotic devices, the robots win every time and hog resources unless artificial ‘think times’ are included as a part of the robotic transaction execution.

## Big data

The path from idealistic concept to practical reality is not always an easy one. When creating a Smarter Planet solution, using input data that was not meant for this purpose provides the application designer with many challenges. These challenges include managing large amounts of data, integrating diverse processing systems, providing the information from that data to the user in a timely fashion, and lowering costs while doing so.

Many of these new Smarter IT applications involve the use of *big data*. Organizations are faced with a challenge today: How to effectively manage and use the massive amounts of data being generated by their existing IT solutions, with more data to come in the future.

Big data challenges include:

<b>Volume</b>	Businesses today are dealing with a tidal wave of data, amassing gigabytes, terabytes, or even petabytes of information. The generation of terabytes of data per hour during peak operations is becoming quite common. This data includes web pages, web log files, click streams, search indexes, social media forums, instant messaging, text
---------------	---

	messages, email, documents, consumer demographics, and sensor data from active and passive systems.
<b>Velocity</b>	Being able to perform analytics on thousands of transactions per second is becoming mission critical. Analytic modeling, continual scoring, and efficiently storing this high volume of data have also become critical.
<b>Variety</b>	Harnessing structured, semi-structured, and unstructured information to gain insight by correlating them together has become a key business requirement for many organizations.
<b>Vitality</b>	Problems and opportunities are not static. Big data analytical models and predictive models need to be updated as changes occur so organizations can seize opportunities as they come.

Examples of applications that collect a large amount of data at a high rate of speed include charging road tolls, collecting meter data for energy management, monitoring locations, aspects of a moving train, and tracking items in transit.

Big data is one example of both the challenge and opportunity associated with the Smarter Planet vision. The successful implementation of these solutions require the ability to gain actionable knowledge and insight from massive amounts of data in a timely manner. For these kinds of systems, the management of big data is an ongoing requirement.

## Solution complexity

Typically, Smarter IT includes orchestration and integration of multiple and often diverse systems that each have their own performance, capacity, and scaling characteristics. Smarter IT must also manage not only horizontal integration between solution technologies, but also the deeper solutions stacks introduced by computing frameworks, such as web services, service-oriented architecture (SOA), and cloud computing.

From a performance perspective, this solution complexity makes it more critical to understand and measure the end-to-end and component level contributions to utilization and response time. For example, it may be relatively easy to count how many events (for example, bank account withdrawals) are processed within an hour. However, from a performance perspective it is also important to understand how many messages are passed between system components, which components take part in the communication, and how many different workflows or connected systems are involved to process an event.

## Performance expertise

Designing and tuning Smarter IT can require multiple capacity and performance subject matter experts (SMEs) who specialize in:

- ▶ Application and application server performance
- ▶ Database and database server performance
- ▶ Network performance
- ▶ Application architecture, design, caching, and coding patterns and practices that contribute to getting good performance from the implementation technologies chosen
- ▶ Performance of back-end systems such as CICS® or IMS™
- ▶ Performance of solution components that might be a solution in their own right, such as document management components, manufacturing part list handling, stock trade settlements, and other services on multiple platforms

- ▶ Choosing the method and format of data delivery to fit the purpose of the data
- ▶ Capacity planning

Smarter IT demands that capacity and performance SMEs deploy varying tuning strategies and tactics. Various components of a system exhibit different behaviors when handling increasing volumes of service requests, approaching thresholds at which they degrade, or when exposed to different servicing characteristics and limits.

## **Customer expectations**

Businesses are looking to consultant and solution organizations to provide better flexibility, improved operating costs, and “greener” options for lower costs. Solutions such as Business Process Management (BPM) and more powerful and lower energy CPU and hardware options have helped deliver some of these benefits. However, the custom software solutions built on these sophisticated technologies have failed to deliver on the benefits that they have offered. Successful projects in the past have and undoubtedly will incorporate a multi-layered approach to cover the topics addressed above:

- ▶ Performance requirements management
- ▶ Design for performance
- ▶ Monitoring throughput, response times, and service level agreement (SLA) metrics
- ▶ Test performance requirements

To address these challenges and expectations, measurement processes must be implemented to allow us to manage the performance and capacity of these solutions.

## **Managing IT performance and capacity for the Smarter Planet**

For configuration, sizing, and tuning of composite solutions, there is a need for reliable, repeatable, and traceable measurement results. These results shed light on how the solutions components perform while supporting the business transactions used to measure the performance requirements.

## **Performance Engineering**

Through use of Performance Engineering Methodology, performance and capacity challenges have been met with performance requirements management, design guidelines, and checklists, which are all common to a large number of projects. However, it is the definition, measurement, and monitoring of those requirements, the translation of business transactions to the resulting technical transactions, the setup of appropriate test scenarios and environments, and the interpretation of the measurements, that pose an increasing challenge.

Integrating skills of performance and capacity SMEs from various business areas is central to the Performance Engineering Methodology. This integration provides the ability to identify, consolidate, and focus on all key nonfunctional business requirements, in terms of response times, volume of activity, and capacity of a system. Balancing all three elements enables a business to provide better service to clients at reduced operating costs while using less capacity, which leads to a “greener” solution. Without performance engineering, deploying an overall, intelligent, solution-wide performance and capacity management program is nearly impossible.

In addition, development of meaningful and achievable business service level agreements (SLAs) becomes hard, if not impossible, to manage.

To manage the performance and capacity of large and complex systems, it is essential to understand the behavior of the involved components under high load. For example, any part of the solution that decreases in throughput could lead to severe congestion and might drive the solution to an even more critical state. Proper threshold and throttling settings need to be considered to maintain the health of the overall solution. Smarter IT can only be considered truly intelligent if some essential self-regulating mechanism is incorporated that controls the stream of requests inside the system in a way that overall degradation can be avoided.

When performance engineering is not considered during the solution of a system, negative impacts occur. Late consideration of performance during solution test phases can lead to additional hardware requirements (due to increased CPU, memory, and I/O resources consumed), impact the time to deliver, and incur additional costs to the business. In fact, careful performance monitoring throughout the project life cycle is critical because performance analysis in later stages or as a one-time task may lead to a deeper architectural changes that cannot be accomplished in time or solved by buying hardware.

Successful performance engineering principles need to be applied at the beginning of the software development life cycle, from feasibility to deployment and maintenance, and used during the entire life cycle. The requirements definition provides the functional needs and constraints on the performance of the system.

Designing and building performance and capacity capable Smarter IT solutions composed of several diverse components requires skills to foresee all potential inter-relations, as shown in Figure 1.

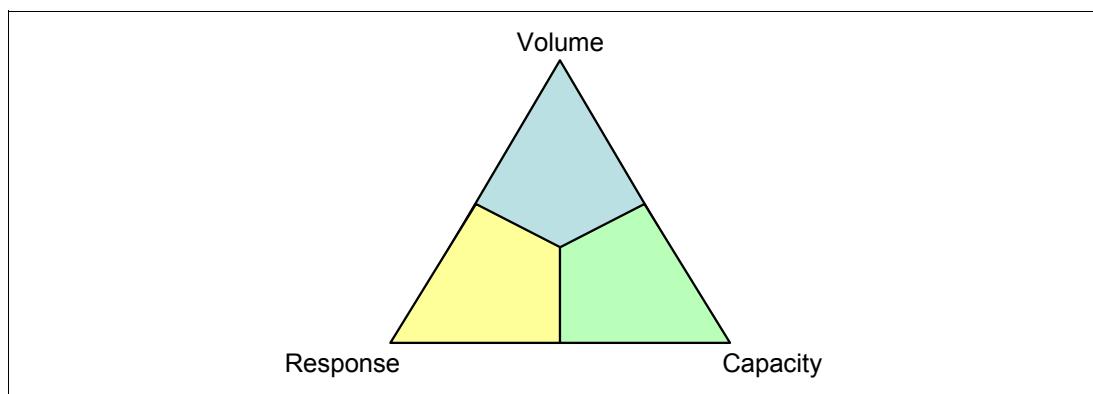


Figure 1 *Balancing performance elements*

In addition, similar skills are required for daily operational aspects. Integrative performance management cross several classic IT silos and bridges to business departments more than usual. Business process owners or even business area leads can become partners of (IT) performance management, as the intelligent solution's performance figures can relate directly to or represent key business performance indicators.

During the early design phase, both functional and performance considerations of the system should be addressed, allowing future refinement and improvement of the design, which better enables the system to perform quickly at volume and to a specified capacity. The Performance Engineering Methodology includes identifying common performance anti-patterns, high impacting transactions/systems, and risk mitigation for hardware and software systems as part of the solution.

## **Capacity planning**

Capacity planning or sizing of the necessary hardware can become a challenge as well, as putting together the sizing information for the Smarter IT components requires sufficient knowledge about the expected volume of requests and their distribution over time. This analysis is performed from an external point of view, but also should include all the interactions between the components within the solution.

Modeling becomes important during this phase to ensure that both the performance and capacity of the solution can be determined and predicted for current and future workloads. It is important to perform unit testing of all components and code being constructed during the implementation phase to ensure that they satisfy the requirements, and that testing results are used as feedback into the performance model. In the testing phase, the solution is tested functionally and for performance to validate and verify that it adheres to the functional and non-functional performance requirements of the system. After the solution is deployed into a production environment, it is monitored and reported on to verify that the performance requirements of the system are being met.

Key success factors for performance engineering include commencing performance engineering early in the life cycle and having a dedicated team focusing on performance with SMEs focusing on technology or systems. Focusing on application performance during the software development life cycle can yield major savings of the initial capacity costs and the subsequent operational savings. Most importantly, early focus on performance helps deliver a viable solution that performs well and achieves the intended business objectives.

Performance Engineering is a path that should be carefully considered when becoming involved in performance matters of Smarter IT.

## **Instrumentation**

There is a standard published by the Open Group that aims to enable solution components to provide valuable insights into the behavior of a solution. This standard is called *Application Response Measurement (ARM)*. Implementing solution components according to ARM provides valuable insights into the behavior of a solution. Measurements based on ARM enables a project to identify how the response time of a business transaction at the user's site is breaking down into response times at the application server, at interfaces towards third-party systems, and at the database level by persisting and correlating the appropriate time stamps. In some commercial off-the-shelf applications, as well as in number of infrastructure components, such as application servers, HTTP servers, and even databases, ARM standards are already supported.

Note that ARM is just one example. In fact, ARM implementations in existing software products have varying degrees of maturity. Identifying the degree of ARM support in the different components of your solution is a sensible first step. This identification provides projects with an idea of where additional augmentation makes sense to approach the goal of an end-to-end application response measurement. The adherence to ARM in projects has provided traceable breakdowns of business transactions from the http level to individual work flow, message queuing, or database transactions. Experience shows that this approach enables engineering to identify key component limits as well as potential performance flaws before the solution is deployed into production. Thus, the solution's configuration can be adapted over time to reflect these findings. Otherwise, these flaws would go unnoticed and potentially render the solution unusable.

## **Composite application monitoring**

In addition to performance engineering, smart solution specific composite application monitoring, with its ability to correlate and aggregate response times from the various components of the smart solution level, is going to play a growing role in the future.

The next section of this paper focuses on some real world examples that address the challenges of Smarter IT projects.

## **Five examples of Smarter IT projects and challenges**

The following examples of Smarter IT projects and domains from recent experience can shed some light on what a Smarter Planet might mean for performance and capacity management

### **Traffic systems: Road Charging**

In the future, IBM Smarter Traffic™ will not only be concerned with providing and funding roads and related infrastructure to support a growing number of drivers, but also finding ways to address issues such as road congestion and increased emissions. This section describes such a system being implemented in a major city, along with the associated performance and capacity challenges that must be addressed.

#### **System scope**

The Road Charging system was commissioned by a local government body responsible for most aspects of transport in a capital city. The system supports the congestion charging and low emissions schemes. The system was delivered as a replacement for an existing system and business operation run by another supplier. The original system was expensive to run and could not easily support future flexibility requirements. There were also concerns over scalability to support higher volumes for future schemes.

IBM was asked to deliver the new system as three separate sub-systems, one for each of the organizations:

- ▶ A core sub-system to support vehicle detection, event collection, and settlement against payment.
- ▶ A sub-system to support registration and product purchases over multiple channels.
- ▶ A sub-system to support issuing and managing penalties for failure to pay.

The requirement was that the solution be delivered within a short time frame and be largely package-based. It should be scalable to support all transport schemes across the capital and other national road pricing schemes.

Figure 2 shows the main Road Charging system components and the interaction with the external systems and users.

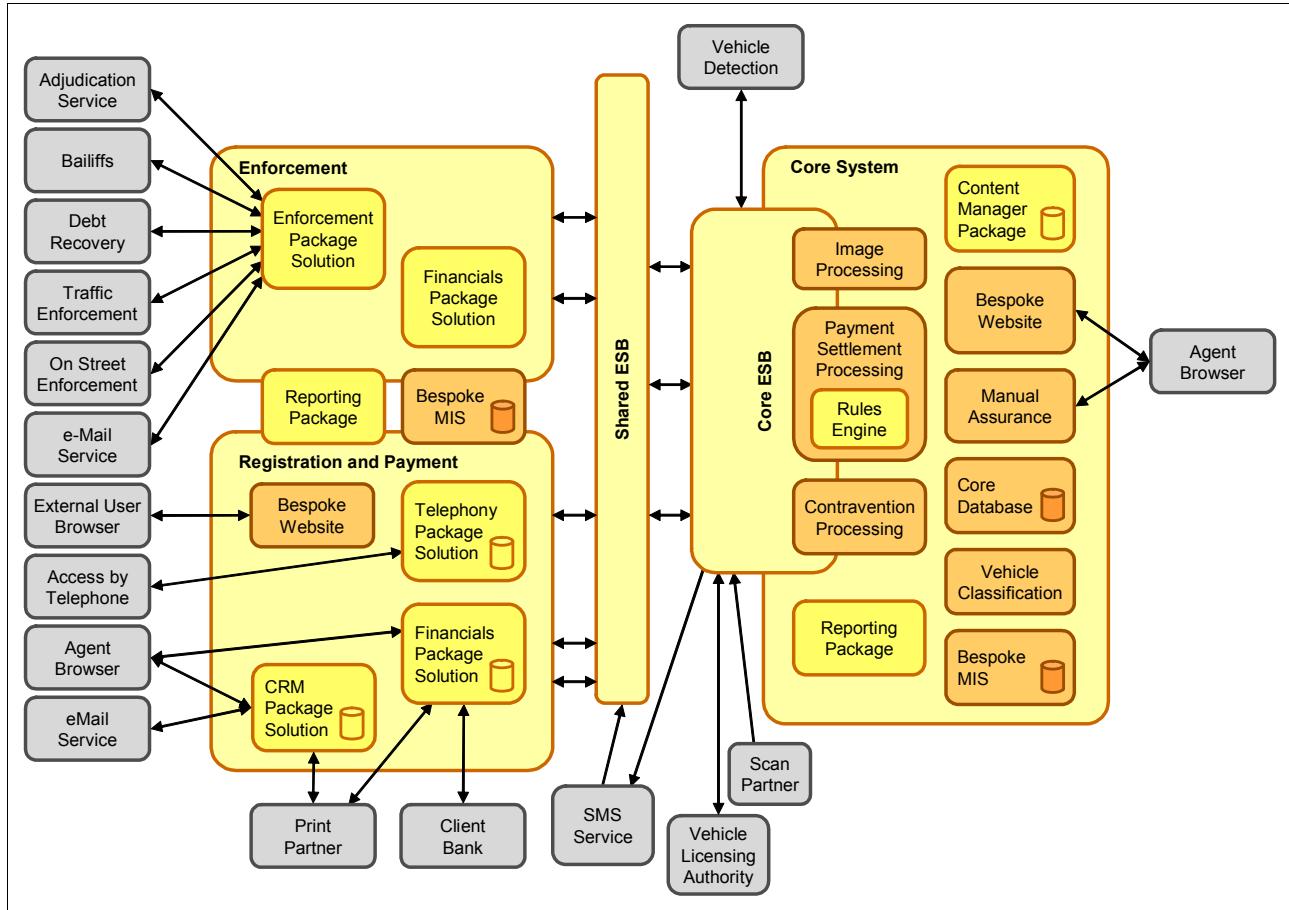


Figure 2 Road Charging components and relationships to external components

The key areas of processing are:

- ▶ The Core system receives vehicle detections from the roadside cameras. Up to three million events may be received per day, with 100 KB images attached and stored in the Content Manager package.
- ▶ The Core system settles charges and runs complex rules processing to group vehicle detections together and applies the correct charges related to the vehicle, the owner, and the scheme.
- ▶ The Financials package must be able to create and send 15,000 statements for account users each night.
- ▶ The Registration and Payments sub-system supports 70,000 payments per day over five settlement channels.
- ▶ Each night, daily changes are extracted from the system and loaded into the Core MIS database. A significant number of reports are run against this data.

## Infrastructure

The solution is implemented on an IBM Power System and Windows platform with infrastructure deployed at two main sites, as shown in Figure 3. The core sub-system runs active/active across two sites with synchronous data mirroring. The Registration, Payment, and Enforcement sub-systems run active/passive across two sites, again with synchronous data mirroring. The system is used by two main groups: agents working in remote call centers and members of the public accessing the system over the Internet.

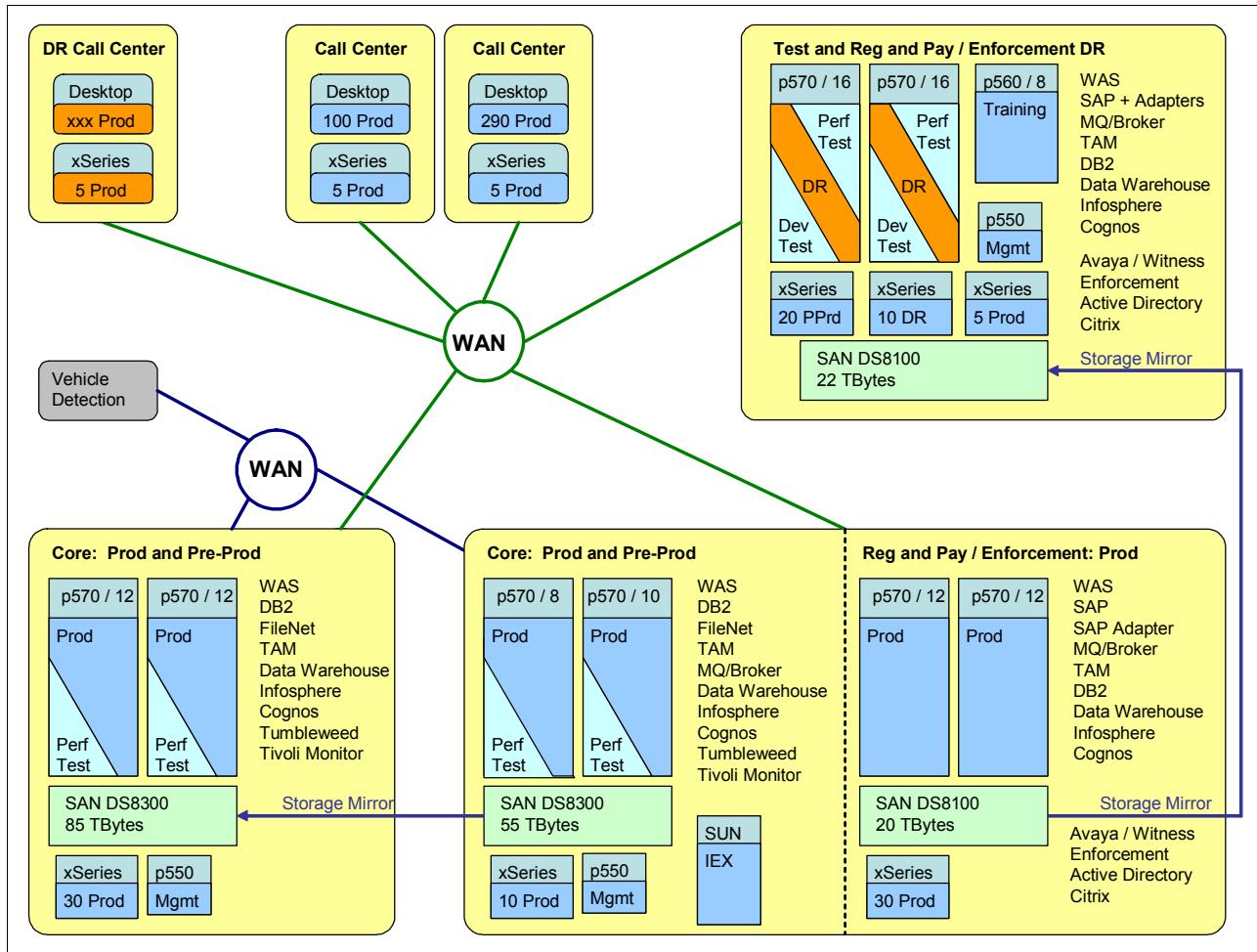


Figure 3 Road Charging server and SAN infrastructure with wide area network context

Data is synchronized between the sub-systems so that the core system maintains a full copy of the data.

## Performance and capacity challenges

The customer requirements posed a number of performance and capacity management challenges in both the Build phase and the Run and Operate phase.

### ***Build phase***

The key challenges for the Build phase are listed below. These challenges resulted in additional design evaluation work (workshops and proof of concept) as well as additional testing.

- ▶ Modeling distributed systems

The solution was designed with three sub-systems, with links to a number of external systems. This is typical of SOA-style applications and much of the functionality was designed as MQ-triggered services or as web service. Services support both scheduled and online processing, which introduces complexity when trying to model the performance of the system because it is difficult to characterize the workload. In many cases, services call other services, which execute across multiple nodes. Complex models are needed to estimate the processing capacity and latency. As the designs for one service are often detailed across several (middleware specific) design documents, it was time consuming to gather the information to model the solution, especially online user-interfaces, which use many services. A decision was made to model only the key processes (take a payment and register an account) and to extrapolate for the remainder, which provided good insight into the characteristics of the system and the expected latency.

- ▶ Brownfield systems and customized packages

The integration and use of customized packages increases performance and capacity risk because they are hard to size. Brownfield systems may not be well understood. Standard package sizing approaches from the vendor are often specific to a standard usage and, when customized, cannot be reliably used.

In the Road Charging systems, the standard sizing approaches were user-based and activity-based models. However, with significant usage of the packages driven by services accessing APIs, the standard sizing approach could not be used, which reduced the confidence in our package estimates. Significant contingency was added to the estimates and planned focused testing.

Another challenge with using packages was the difficulty of loading performance test data. Security restrictions meant that data could not be copied from production, which in any case may not match the desired profile or volume. In our case, it was necessary to load data items for test each one individually using a package API, which was significantly slower than using a utility. The solution supported large numbers of users and an even larger numbers of readings. Significant time (weeks or months) was required to generate and load test data.

- ▶ High workload and scalability requirements

The requirement to receive, store, and process three million messages per day, each containing five vehicle images, was a challenge. Complex rules-based processing involved matching vehicles with similarly interpreted number plates, detected at the same camera within a short period of time. The processing needed to be completed in near real time to avoid the build-up of queues. Although this risk was high, it was possible to break down the processing into units and to estimate the cost and duration. The initial modeling resulted in a significant design change to increase the parallelization and the batching to allow processing to complete in the time required. The solution design needed to cater for high throughput and established approaches and design patterns.

On the Road Charging program, we also used the performance testing facility at Montpellier France to conduct a proof of concept (PoC) for a content manager. This PoC provided assurance that the proposed package would support the workload.

- ▶ Flexibility

The flexibility requirement was met by using rules-based processing, which was a key performance focus. Rules-based processing has the potential to introduce significant processing impact. The ILog JRules rules performance was assessed using test harnesses. In most cases, the rules performed well, although in some cases, tuning work was carried out to optimize the rules processing. In all cases, good performance was eventually achieved.

- ▶ Availability and data mirroring

A high availability requirement, including disaster recovery sites, increased the complexity of the middleware configuration, the storage solution, and the management of the storage. It also introduced additional performance and capacity work because of the need to access data across the WAN and the additional latency due to synchronous latency. These risks were assessed and tests were carried out to characterize performance. The data mirroring used Metro Mirror over a 1 Gbps fiber link that did not add excessive latency to data access. Accessing databases over the wide area network (WAN) did add some latency. Processing was load-balanced to either the site with the database or to the site accessing the database remotely across the WAN. Unequal load-balancing was used to match the throughput achieved on each site. This resulted in a balanced latency irrespective of where the processing was completed. For solutions involving data mirroring, the storage design must be considered carefully and significant additional storage is required. A master and two copies of the data are typically required to support backup of the mirrored data.

### ***Run and Operate phase***

The final challenge relates to the Run and Operate phase, mostly because of a *large virtualized estate*

Performance and capacity planning after implementing a large and virtualized distributed infrastructure is complex. The project focused on only three main resources: CPU, storage, and network. But even so, these three resources presented challenges. Our focus was on the production environment, but with some consideration for disaster recovery and performance testing. The program delivered a monthly performance and capacity report to the customer.

The production environment consisted of 75 IBM Power System LPARs on six servers and 95 IBM System x® servers or blades. Disaster recovery and performance testing together made up a similar number of images.

The CPU is managed at the server level (Power 570, System x, and blades) and at the LPAR level. The aim is to ensure that processing capacity used matches expectations and that variations can be addressed by tuning or making changes in CPUs. The actual business processing workload is compared with infrastructure utilization from System Resource Manager (SRM). For key components, extrapolations are calculated to estimate usage at target workload.

Storage is managed for key data stores only. Alerting is used to identify smaller shortfalls, which can be made up from contingency. Models have been produced to estimate storage usage based on business processing completed and data retention policies. Usage information is again from SRM. The models are used to ensure that storage usage is understood and controlled. SAN latency is being reviewed.

The network is managed by a third party, AT&T, who provides data on network utilization for the WAN. This data is delivered monthly and provides data to allow the program to extrapolate usage at the target workload.

We considered the capacity to support both normal operation and operations following a server or a site loss. These alternative scenarios added complexity to the modeling but have been useful in identifying areas where tuning would be beneficial.

A critical consideration in preparing the necessary performance and capacity management reporting for an environment of this scale is data query and extraction techniques. In this environment, several techniques are used. For application transaction volumes, data is extracted from the MIS database by SQL queries and Excel is used to visualize the data. Crystal Reports is used to query and group data from an IBM Capacity Management Database known as SRM database and then prepare graphical charts and tabulated lists. Other vendors supply summarized management data for the technologies they are responsible for, such as mobile phone payment channels and network utilization.

A configuration database has been built to record the mapping of operating system images (server partitions) to business processes and business functions, which has enabled utilization by process or function to be targeted. It also distinguishes between production and non-production environments.

There is an ongoing evolution of Performance and Capacity reporting. It is beneficial for run teams to have access to the design authorities and to the people who operated the design/build phase testing and their documentation and models.

## Energy: Meter Data Management

In the future, Smarter Energy will dynamically gather information about energy usage to help manage both the distribution and consumption of energy more efficiently. This section describes the field of Meter Data Management (MDM), a key component of the Smarter Grid, along with its inherent performance and capacity challenges.

Meter Data Management for intelligent meter implementations requires the deployment of a solution capable of collecting, processing, and reporting on the data of thousands or millions of meters on a daily basis. MDM systems (MDMS) that do this task must be secure, reliable, highly available, efficient, and timely to keep up with the high volume of data that continually arrives. IBM provides MDM solutions for clients that combine specialized software with IBM hardware and services that result in systems that meet their needs.

From a capacity and performance viewpoint, there are a number of characteristics of such systems that need to be accounted for, including:

- ▶ Accuracy

Accuracy can be looked at in a number of ways. When designing MDM systems, the accuracy of the meters will have a big impact on the amount of capacity required. The more data intervals that each meter provides each day, the more capacity that will be required to store the data. For example, two million meters reporting every two hours can provide as much data as one million meters reporting hourly over the course of a day, which is the same as 250,000 meters reporting every 15 minutes. Likewise, intervals measured to one point after the decimal point will require less storage capacity than intervals measured to two or more places after the decimal point.

- ▶ Reliability

The reliability of meters also affects the performance of the MDMS. Meters that provide gaps in their data may drive the need for estimation by the MDMS to fill in these gaps, and such estimation requires additional processing from the MDMS to provide the customer a consistent view of their usage. Likewise, meters or meter collection systems that provide data out of sequence will also drive significant estimation by the system.

- ▶ Processing windows

The need to process meter data in a timely manner also drives performance considerations. For example, what capacity is required to process the meter data in the time allotted before the next wave of data arrives? What happens if there is a problem processing the data and there is a need to catch up? Meters are continually providing information to downstream systems. These systems need to be designed to accommodate not only day-to-day results but also exception results.

- ▶ Higher availability

While there are significant storage considerations to take into account when developing MDM systems, there are also significant network considerations. Not only does the meter data have to be collected, processed and reported on, but it also needs to be backed up and possibly replicated to another location due to business continuity requirements. There is potentially a lot of data to move around, and the network needs to have the speed and the bandwidth to do this task effectively.

- ▶ Auditability

Customers and their clients want to see information about the data coming from their meter. How they access this information and how much of it they want to retrieve will drive additional storage and performance requirements on the MDMS beyond the day to day transaction processing of meter data. You need to determine what types of queries and reports will be required, and how much data needs to remain on hand to satisfy these requirements.

These are some of the characteristics that organizations who are designing, building and managing an MDMS need to keep in mind to have a system that meets and exceeds the performance and capacity challenges that they will face.

## **Utilities: Energy consumption and conservation**

In the future, smarter utilities also mean empowering business and home energy users to manage their energy and water consumption more effectively. This section describes pilot solutions being employed in the Netherlands.

### **System scope**

IBM Netherlands created a Central Platform for Smarter Solutions (CPSS) to facilitate easy implementation of smarter solutions in an existing infrastructure environment. The infrastructure was designed to allow for a wide and flexible range of different solutions.

The current infrastructure is mostly intended for and used in pilot projects and proofs of concept, but has been designed so that the same system architecture can be scaled up to much larger implementations.

Currently, pilots have been run for the use of smarter systems in:

- ▶ Retail: Energy consumption and conservation in supermarket buildings.
- ▶ Home: Smart meters that collect and keep energy consumption in homes.

## System design

The architecture for CPSS was defined at a high level by identifying the main functional characteristics, as shown in Figure 4.

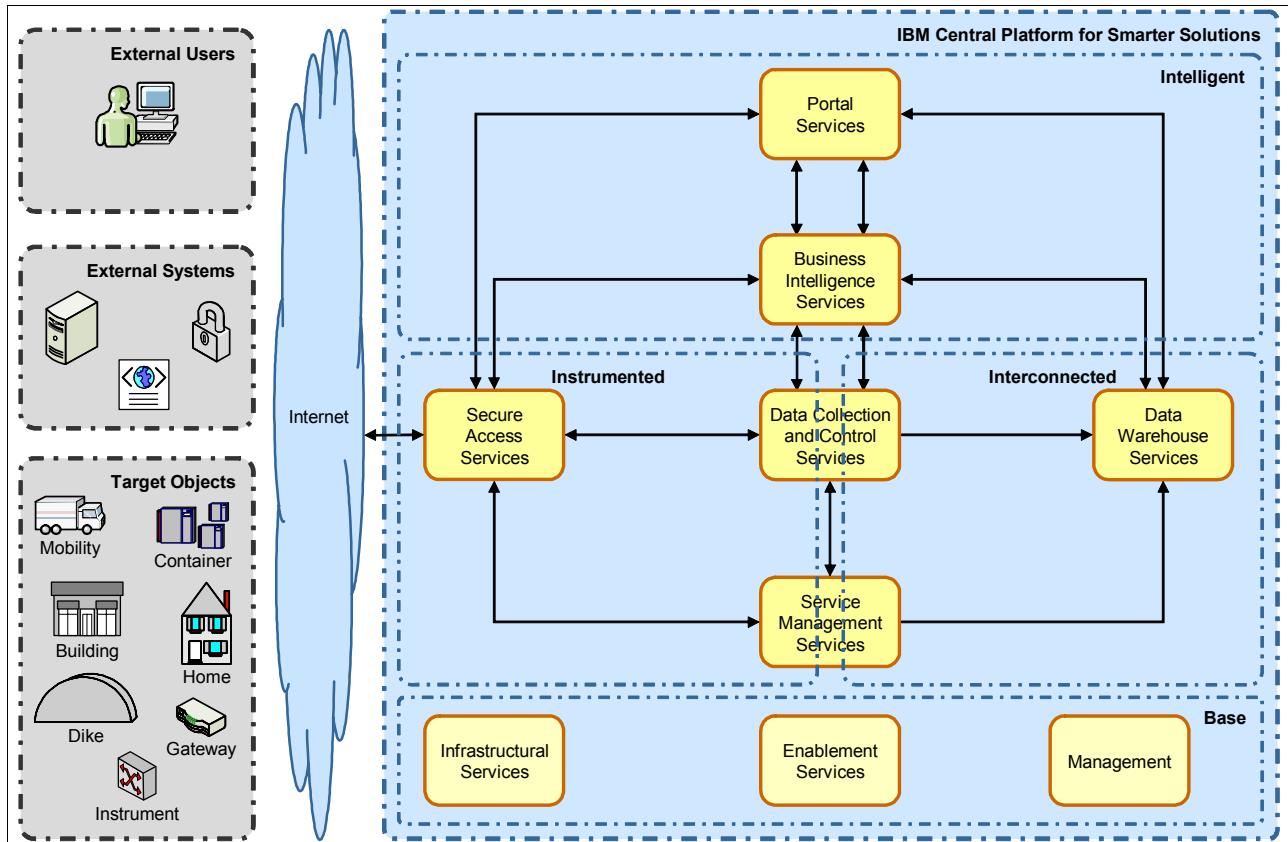


Figure 4 Central Platform for Smarter Solutions – functional architecture

Figure 4 illustrates the main functional components that occur in almost every smarter solution:

<b>Target objects</b>	Smart measuring devices that are put into many objects. These devices must, in some way, be connected to the Internet to allow for the collection of data in a central place.
<b>Data collection and control</b>	The heart of the solution, where communication with the target objects is done to collect and process measurements.
<b>Data Warehouse</b>	The central database(s) where measurements are stored and collated for further processing.
<b>Business Intelligence</b>	A tool aimed at analyzing the collected data and deriving meaning from it.
<b>Portal</b>	User access to the smarter solution, mostly to look at the data and perform analyses on it.
<b>Service Management</b>	Tooling to manage and maintain the multitude of target objects, which are not being used in a classic controlled environment, which makes managing them a harder task.
<b>Secure access</b>	A security layer to guard the environment against inappropriate access from the Internet, to which it is directly connected.

## **Implementation**

The CPSS platform was implemented in a small environment for pilot purposes, using VMware images running on just two physical systems. All functions within the platform were fulfilled using mostly standard IBM software products:

- ▶ Data collection and control: Implemented using IBM WebSphere® Sensor Events (WSE)
- ▶ Data Warehouse: Implemented using IBM DB2®
- ▶ Business Intelligence: Implemented using IBM Cognos® Business Intelligence
- ▶ Portal: Implemented using IBM WebSphere Portal Server
- ▶ Service Management: Implemented using IBM Maximo® Asset Management and IBM Tivoli® Service Request Manager®
- ▶ Secure access: Implemented using a Cisco firewall and IBM Site Protector

## **Capacity and performance considerations**

Due to the small scale of all current projects, performance and capacity limitations have not played a big role yet. There are currently about 700 devices connected to the system, which is not yet generating large amounts of data.

However, there are plans to extend the platform to an operational situation, where the number of smart meters connected to the system would number in the hundreds of thousands. The design has been made in such a way that this can be facilitated relatively easily, using the scalability features of the products:

All the WebSphere Application Server based products (WebSphere Sensor Events, WebSphere Portal Server, and Maximo Asset Management/Tivoli Service Request Manager) can be extended to multiple nodes using the standard WebSphere Application Server facilities. DB2 has its own facilities for scalability.

In practical situations, the biggest capacity demand will be placed on WebSphere Sensor Events in cooperation with DB2, because these two applications have to be able to process an ever-growing number of devices continuously sending messages.

By comparison, the capacity demand on products such as WebSphere Portal Server and Cognos BI will be much smaller, because users typically do not spend a lot of time on the system. They will just take an occasional look and leave the system alone for the rest of the time.

For this reason, we have done some capacity investigations specifically for the WebSphere Sensor Events and DB2 combination. Without any form of optimization or tuning, the current simple environment was capable of handling roughly 250 messages per second. It is likely that this processing can be improved significantly through tuning and optimization, but this number is good enough to make some realistic capacity estimates for larger systems.

The capacity demand on WebSphere Sensor Events/DB2 is almost completely determined by two factors:

- ▶ Number of devices attached to the system
- ▶ Number of messages each of these devices is sending per hour.

The second of these numbers is one that can be partly chosen by the implementation, so there is opportunity for tuning on that side of the system as well.

For example, if you are using smart meters that send a message with their current status every 10 minutes, you will get six messages per hour per device. With a WebSphere Sensor Events/DB2 capacity of  $250 \times 3600 = 900,000$  messages per hour per CPU, this gives the system the capacity to handle 150,000 devices for every CPU.

However, if devices are sending messages every 10 seconds, a single CPU can support only 2500 devices.

So the overall design of the system must look closely at how many messages each device really needs to send, since sending a lot of superfluous information (house temperatures do not change every 10 seconds) will put a big burden on the system.

## Rail systems: Live Train

Smart Mass Transit solutions will play an increasingly important role in the future as we build mass transit systems to meet the transportation needs of our societies in a viable and sustainable manner. This section describes a specific rail solution using live data to manage its operations more efficiently, along with its associated performance and capacity challenges.

The Live Train solution allows bidirectional communication between on-board train systems and the central control room. The concept involves the use of on-board train information linked to a Global Positioning System (GPS) location service to supply a “live” status of the train’s true position as well as on-board train sourced information regarding the current quality of the service provided by the train. This data includes on-time running, delays, track faults, toilets in operation, doors opening on time, and brakes in good condition. Data would be uploaded real time from a general packet radio service (GPRS) on-board train transceiver.

The pilot system produced a high volume of data on the status of on-board systems cross-referenced with train position information. In contrast, the network pipe provided by the GPRS network was relatively thin, slow, and costly, which created a number of performance and capacity challenges that need careful architectural thinking. A key consideration was to separate the data that had to be available in near real time to determine the correct status of service quality from the wealth of data that was collected to intelligently diagnose problems in a “smart” way.

The team considered the following performance and capacity considerations in the overall end-to-end architecture:

- ▶ Defer delivery of non-time-critical data.

This task was achieved by ensuring that the On Train Meter Recorder (OTMR) only transmitted a data packet when a predefined condition was met. The remaining data would be kept for historical trending and would be uploaded using Wi-Fi when the train was stabled at a depot each day.

- ▶ Data refresh rates must be appropriate.

The 30 second interval data from the train is transmitted to a central control room. Most of the time, this data was a latitude/longitude reference with a destination address as the only additional data in the packet. No OTMR data was transmitted while the train was outside the depot unless required. This resulted in a small data packet being sent over GPRS.

- Visually relevant presentation of data.

The data that was displayed had to be correct. The balance between wanting more information for the user (so they are able to do more and gain more value from the information quickly) versus the ability to deliver that information efficiently had to be kept in focus.

The BodenSee Telematics gateway<sup>1</sup> was used to provide a rich user interface with just a trickle of information from the on-board train data. Also, from a positional reference perspective on the screen (especially a human's ability to react to swiftly changing information), data refresh rates are critical. BodenSee can provide an easy-to-view plot of the position of the train as well richer information when available.

- Make the system intelligent.

The Live Train solution in Figure 5 provides more than just train positioning and OTMR data. Live Train was also about self scheduling fixes and optimizing the use of the asset (as long as it was safe and would meet service quality targets). The OTM data would be available to initiate triggers in business processes. The key is to make sure that the incident management elements of Live Train could be managed from the same data feed, that is, no additional data is being demanded from a real-time perspective. Data classification was critical. To meet performance objectives, the data processing rule was "If not needed, do not process".

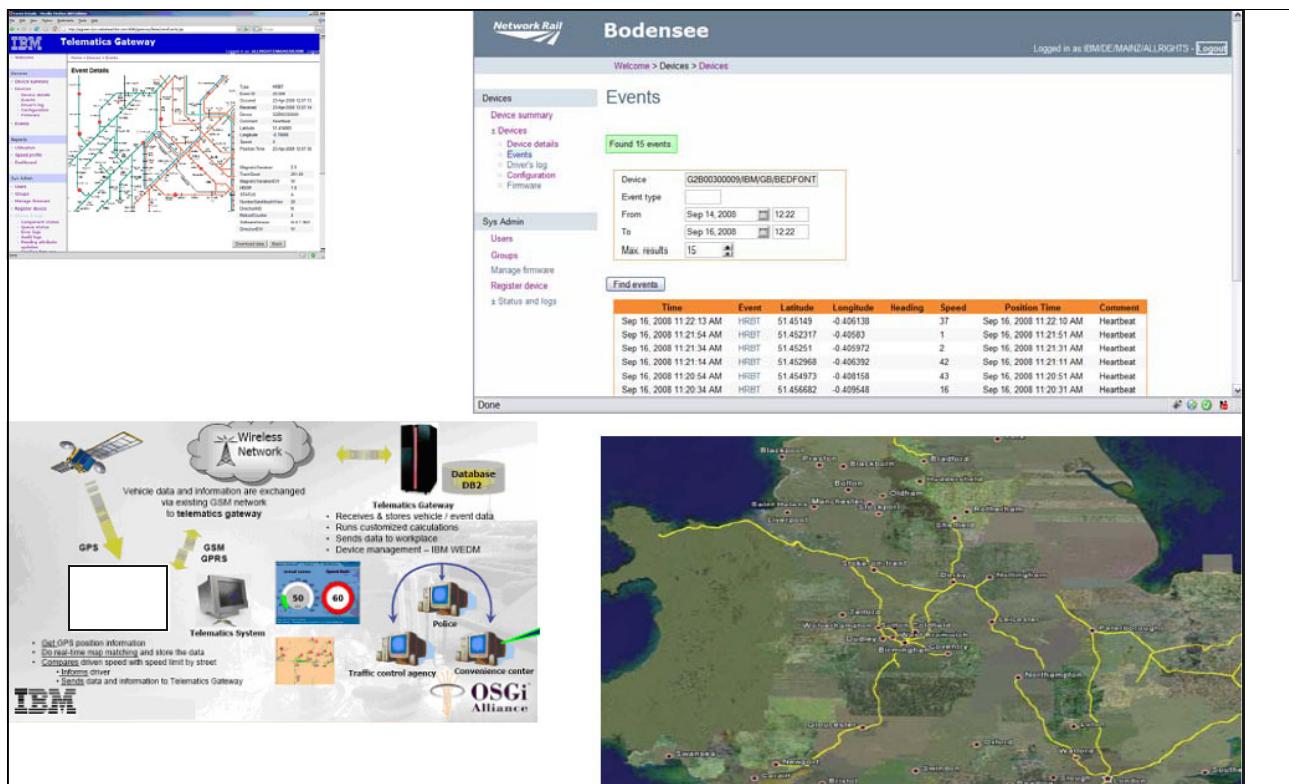


Figure 5 The BodenSee Telematics solution at work

<sup>1</sup> [http://transportationfortomorrow.com/final\\_report/volume\\_3\\_html/04\\_public\\_sessions/contentf07a.htm?name=101806\\_presentation\\_lamba](http://transportationfortomorrow.com/final_report/volume_3_html/04_public_sessions/contentf07a.htm?name=101806_presentation_lamba)

The illustrations included in Figure 5 on page 19 are, clockwise from top left:

- ▶ The front screen of the Telematics gateway showing a schematic of the network.
- ▶ An event feed from the OTMR device on a train.
- ▶ Map matching of positioning plots to a geo-spatial view.
- ▶ End-to-end diagram of the Telematics solution.

## Solution design

One focus of the solution design was to make sure that data transmissions would be as small as possible, and that each bit in the byte-sized message would be relevant. A fair amount of the intelligence and display processing can be inferred. If the track is straight and the train is running at a constant speed, its next reference point can be anticipated. Most of the detailed historical data that can be used in analytics processing is transmitted by the more traditional Wi-Fi upload approach at the depot.

The correct data for near real-time display and for analytics is transmitted separately. “Fit for purpose” is therefore an appropriate design guideline. Intelligent systems are often constrained in unique ways and architects need to think about the variety of data delivery methods to provide a full service. The standard delivery mechanism or infrastructure does not fit all requirements.

## Transportation logistics: Radio Frequency Identification

The intelligent transportation of goods to people has significant ramifications for our society. This section describes the role of Radio Frequency Identification (RFID) solutions in making our distribution systems more intelligent, along with their associated performance and capacity challenges.

RFID systems allow the identity of physical objects to be recognized by IT systems through an exchange of radio signals. RFID systems are important for smarter systems. In many smarter systems, they are the primary source of data: the eyes and ears of the system. Examples include smart bag tags used by airlines, windscreen mounted “tags” used in toll road systems, and consignment tracking devices used in parcel delivery.

The management of the performance and capacity of RFID systems includes a number of significant considerations in addition to those traditional IT systems and other smarter systems due to the impact of the object’s environment on the object. This impact can be difficult to predict without testing. Environmental factors that impact RFID systems include humidity, temperature, physical impact, reflection, water, radio interference, and snow. Understanding the environment where tagged objects operate and how they behave is necessary to establish a sound performance test.

Performance engineering for traditional IT systems suggests that the performance characteristics of the business systems should be understood before development. In the same way, performance engineering for RFID systems requires a full understanding of the RFID aspects of the system. Their performance characteristics should be developed before the RFID system is developed. The components of the system are carefully selected to meet the requirements and to manage the unpredictable impact of the environment on performance. To support performance testing, performance tuning is carried out in real life conditions. RFID performance test and tuning is performed early in the delivery process. Common performance metrics include maximum distance between tag and antenna (read-range), proportion of successful tags read (accuracy), maximum number of tags that can be read (volume), and time to perform the read (speed).

After these considerations have been established, the expected benefits of the deployment can be secured and IT application development can commence.

Figure 6 represents a high-level component-model for the RFID system. It highlights five important areas to consider while performing RFID performance testing and tuning.

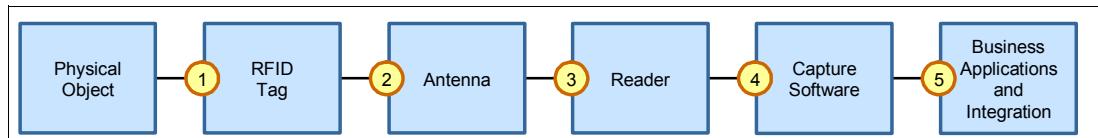


Figure 6 RFID component model

In Figure 6:

1. Tags come in different shapes and sizes and can operate on different frequencies, with or without a built-in battery. Different tag technologies are suitable for different requirements; paper tags, ruggedized tags, and tags for metal objects are examples. Tags have varying sensitivity and can be vulnerable to the orientation of each tag's attachment to the object. Selecting the RFID tag involves assessing different tags (potentially from different vendors) in the test environment where performance metrics can be established statistically for evaluation purposes.
2. The antenna interacts with the tag by transmitting and receiving radio waves. The antenna is either mounted on a hand-held, mobile, or fixed reader. For a hand-held and mobile scenario, the test involves trying different hand-held types from a range of vendors to see which combination of antenna and tag yield the best results. Typically, the location of the tag in combination with the movement direction and angle of the hand-held or mobile device is tuned for best performance. For a fixed scenario, the locations of the antenna(s) and type needs to be carefully tuned so that the angle facing the tag direction is optimal.
3. The antenna is connected to a reader. Depending on the application, there may be up to eight antennas. It is important to tune the number of antennas so that the optimal read coverage is generated for a particular area. Shielding technologies may need to be considered to minimize interference from other close-by readers. The reader has several configuration parameters to tune the performance that are controlled by the capture software.
4. The capture software controls the read process by controlling the reader. It can sometimes run on the reader itself (always for a hand-held) or on a computer/server connected to the reader. The main focus is to configure the radiated power so that the most optimal read range is met while not reading other objects (bleedover). Other parameters that can be configured are the antenna polarization, number of random slots the tag should use to place an answer, the duration of the read, and tag filtering. All of these will have to be adjusted and tuned until the best results are achieved. If the reader also has additional I/O devices attached (such as a light tree), the cost of controlling these devices needs to be considered and factored in when doing performance tuning.
5. At the Business Application and Integration stage, the performance of the RFID equipment has been engineered and optimized, and focus shifts to the Performance Engineering and Capacity Management of the related business system. A critical volumetric for this work will be the workload generated from RFID capture.

Good performance requires that no unnecessary application functions are executed. Many RFID systems have the potential to capture a great deal of data. Tag data should be collected only when it is needed. If that is not possible, it might be effective to filter unnecessary tag reads in the capture software.

If the business system does not require real-time data, batching tag reads for a defined time period can also be used to avoid the business application receiving messages/calls for each individual tag read.

We know from experience that successful RFID performance tuning requires subject matter expertise. It requires experience with various types of RFID tags, readers, and antennas so that lessons learned from similar environments can be used. It requires specialist skills and experience to drive RFID performance tuning through all the areas defined above. It requires a network of tag, reader, and antenna manufacturers for the component selection process. A robust software platform is required to run the capture, and business applications need to be selected or written to meet business needs. IBM WebSphere Sensor Event software is an example of a robust, flexible, and scalable platform.

The next section summarizes our recommendations based on our experiences with these and other projects.

## Recommendations

Smarter Planet is a key focus for IBM, and intelligent applications are becoming common. It is therefore important to build up leading practice recommendations for how performance and capacity challenges should be managed. The experience of IBM is that performance issues identified late in the life cycle are likely to delay implementation of a program and significantly impact costs. Based on the examples above, there are a number of recommendations for managing the performance and capacity of large and complex Smarter Planet projects.

### Risk-based focus

In the Build phase, it is a best practice to adopt a performance risk-based approach to identify issues with the middleware, packages, and application architecture. The complexity of such a solution means that it will be almost impossible to acquire the time and resources necessary for complete coverage. As long as fundamental architectural decisions and approaches are assured, more localized issues can generally be resolved with contingency. Contingency may take the form of focused tuning effort, or additional resources such as CPU or storage.

Specific tasks should be planned to identify risk areas and then to review them in workshops. Workshops should involve the component owners and performance architects. The workshops document identified issues and plan work within the program to address any items that could result in significant issues and ultimately prevent delivery. Proofs of concept or specific testing can be planned early in the development life cycle to address specific risks, as is being done with the utilities energy example. It is far cheaper to address performance risk areas early on than to leave the risks to appear in test where unexpected delays and significant costs can result. Tests should be planned to validate that risk areas are resolved.

Performance risks that are most important in Smarter systems include:

- ▶ Seasonal and daily peaks
- ▶ Workloads caused by error conditions, such as network failure that leads to a flood of work
- ▶ Delayed delivery of real time data due to non-real-time data

## Continuous performance testing

Many smarter systems apply information technology in ground-breaking ways to data that was not previously destined for IT systems. For this reason, it may not be possible to accurately estimate or model the system being built, so a continuous performance testing approach may be beneficial.

In this model, applications are tested as they are built to identify performance issues early and to allow as much time as possible for them to be remedied, in addition to the usual verification against requirements.

In some instances, this approach could include building application prototypes or proofs of concept to allow testing during design. This approach might be particularly beneficial in systems that will be broadly deployed.

## Defining a tooling approach

Smarter IT involves both processing complexity and high workload. It is best to spend time agreeing to a tooling strategy to support performance engineering. This strategy should cover a number of areas.

- ▶ Modeling tools or approaches are required. Modeling may be complex because smarter services may execute on multiple servers or partitions, possibly at several sites. Application workloads must be modeled running concurrently and with dependencies. Consideration should be given to meeting SLAs. On the Road Charging program, analytical models in Excel, an IBM model known as RUMPUS, and vendor sizing models were all used. Online processing is generally easier to model than a daily schedule. Models are useful to plan performance and capacity in the Build phase and for Capacity Planning in the Run and Operate phase.
- ▶ Data generation and load tools are required for performance testing. As a best practice, performance tests should be conducted on an environment with data volumes matching steady state volumes. In some cases, with data retention periods of two years or more, these tests can involve a significant amount of data. Typically, Smarter IT will have several key data stores, all of which must be loaded with consistent data. Systems can be particularly complex to load because of the large volumes. In Brownfield cases, it is necessary to test a migration or cut-over, which requires suitable data. Planning for data growth is also an important consideration.
- ▶ Workload must be generated for performance testing. Testing requires a workload that simulates at least the highest expected user- or system-generated activity levels and all of the critical transactions. Tools such as IBM Rational® Performance Tester could be used to generate workload of the correct throughput and the profile.
- ▶ Performance test results must be captured to enable analysis. To capture the data necessary to provision a correctly sized production environment, performance data must be captured both during performance testing and on an ongoing basis. In the IBM AIX® environment, IBM NMON and perfmon can be used to capture server utilization. A bespoke repository can be created to hold the results or, for NMON, the NMON2RRD tool is available to hold the results.
- ▶ For performance and capacity management, tools are useful to understand server resource usage and business workloads. Tools such as IBM Tivoli Monitoring and IBM Tivoli Composite Application Manager provide useful information that can be accessed directly through Tivoli Enterprise Portal, through the Tivoli Data Warehouse, or with a reporting application such as Cognos. With a large estate with hundreds of partitions and servers, the effort required to plan and manage IT resources is significant.

- ▶ Plan tooling to support the Run and Operate phase. Tooling must be planned to support delivery of performance and capacity reports for a large-scale system. Consideration must be given to data repositories and how data is extracted and reported. Reports can be developed showing business data held in management information components or data provided by monitoring tools or scheduling tools; data may also be provided by third parties whose services contribute to the end-to-end solution. Tools such as Cognos can be used to run reports periodically and provide standard information for a performance and capacity report.
- ▶ Configuration management is critical in a Smarter Planet deployment, especially those environments with large and complex environments. It is important to have an up-to-date view of the configuration of the environment, both current and historical. This view will become more important as dynamic cloud technology becomes more pervasive.

Appropriate tooling needs significant attention at the start of a Smarter Planet project. The wrong tools can leave a project flying blind and the effort and expenditure required to establish control over a complex environment can be considerable.

## Focusing on key volumetric drivers and key components

In the Run and Operate phase, with a large estate and complex processing, it is best to focus on the key drivers of system performance. In many cases there will be one or two key business volumetrics (possibly measured by channel) and these should be used to model the business and manage performance and capacity. A complex approach requires significantly greater effort to implement and maintain and may not provide much better results. For example, on the Road Charging program, the key volumetrics are vehicle detections, payments split by channel, and statements issued. Many other more detailed metrics depend upon (and are proportionate to) this indicative set and so do not need to be modeled separately. Focusing on a small set of indicative volumes for key applications reduces the work required for performance and capacity management. In many cases, almost all the resources are used by a small subset of the components.

## Importance of data management

Most Smarter Planet solutions receive and process high transaction rates and large volumes of data each day. We saw these items in the energy data collection examples. The solution must be able to support:

- ▶ The ingestion and persisting of high volumes of data.
- ▶ High transaction rates against high performance and availability requirements.
- ▶ The management of large data volumes. Purging and archiving of data will be necessary as well as normal backup and data management activities, potentially concurrently with normal processing. A good example of this is the Live Train solution, where data involving a completed trip can be archived or purged.
- ▶ As transaction rates and data volumes increase, more attention must be given to the data management.

## Patterns for reliable performance in smarter systems

Many existing patterns apply to smarter systems and there are many smart-specific patterns that are applicable, as shown in our example of RFID item tracking. Some of the more important items are:

- ▶ Many smarter systems are more prone to unexpected workload peaks than other systems. For this reason, reliable performance depends on a strategy to manage an overflowing workload.
- ▶ Where a system has both real-time and non-real-time data, it is important to distinguish between them and to provide a mechanism to provide an appropriate quality of service to each one. Similarly, the strategy to manage a data flood may be different for each workload.

## Integrating Performance Engineering Methodology from the start

Successful Performance Engineering Methodology principles need to be applied at the beginning of the software development life cycle, from feasibility to deployment and maintenance, and used during the entire life cycle. The requirements definition provides the functional needs and constraints on the performance of the system.

When performance engineering is integrated into the solution of a system, negative impacts are avoided, such as additional hardware requirements (due to increased CPU, memory, and I/O resources consumed). Performance Engineering Methodology can have a positive impact application schedules and costs.

## Conclusion

A smarter planet is forcing us to move beyond the traditional methods of collecting business throughput data, both current and future, by manual means. It is also coercing us to move beyond manual and periodic analysis of this data to pre-determine the allocation of IT resources to meet business throughput and response time needs. There is simply too much data and it changes too dynamically for manual analysis and adjustment.

We have already seen many projects learn the painful lessons about requiring smarter regimes of measuring business workloads with direct and automated adjustment of IT resource allocation and throttling. Often, these lessons have been discovered when the solution goes live. A chaotic period ensues for weeks or months while corrective action and re-design take place.

Fortunately, we have also seen examples that pursue an engineering approach to achieve service quality and resource optimization from the outset. Typically, the latter examples are based on proven, hardened methods, such as the *Theory of Constraints* in *The Goal: A Process of Ongoing Improvement*, 2nd Rev. Ed., by Goldratt, (Business) Demand Management from ITIL Service Strategy (Version 3) for determining Patterns of Business Activity (PBAs) and Business Demand Policies, Performance Engineering, such as IBM Performance Engineering and Management Method (PEMM), and Capacity Management from ITIL Service Design (Version 3).

Over the past few years, we have talked about what it takes to build a smarter planet. We have learned that our companies, our cities, and our world are complex systems, and systems of systems. Advancing these systems to be more instrumented, intelligent, and interconnected requires a profound shift in management and governance toward far more collaborative approaches.

To achieve reliable, scalable, and performant IT solutions in this scenario, we need early, ongoing, and direct relationships between the people planning the business direction, the people building IT solutions as “fit for purpose”, and the people who run those same IT solutions in a production environment that is interconnected to billions of other components. We also need proven techniques to follow and tooling that facilitates the automation of business workload measurement collection, demand policy consideration, and adjustment of IT application and infrastructure resource components.

## Other publications

This publications is relevant as further information sources:

Goldratt, et al, *The Goal: A Process of Ongoing Improvement*, 2nd Rev. Ed., North River Press, 1992, ISBN 0884270610

## Online resources

These websites are also relevant as a further information source:

- ▶ A Smarter Planet  
<http://www.ibm.com/smarterplanet>
- ▶ Big data  
<http://www.ibm.com/software/data/infosphere/hadoop/>
- ▶ How Big Data Will Impact Business  
<http://www.ibm.com/software/ebusiness/jstart/bigdata/infocenter.html>
- ▶ BodenSee Telematics Gateway  
[http://transportationfortomorrow.com/final\\_report/volume\\_3\\_html/04\\_public\\_sessions/contentf07a.htm?name=101806\\_presentation\\_lamba](http://transportationfortomorrow.com/final_report/volume_3_html/04_public_sessions/contentf07a.htm?name=101806_presentation_lamba)
- ▶ IBM WebSphere Sensor Event software  
<http://www.ibm.com/software/integration/sensor-events/features/>
- ▶ Open Group Application Response Monitoring (ARM)  
<http://www.opengroup.org/management/arm/>
- ▶ Tivoli Composite Application Manager for Transactions  
<http://www.ibm.com/software/tivoli/products/composite-application-mgr-transactions/>

## The team who wrote this paper

This paper was produced by a team of specialists from around the world.

**Frans Buijsen** is an IT Architect in the Netherlands. He has 15 years of experience in the systems and service management field. He holds a degree in mathematics from Leiden University. His areas of expertise include systems management, service management, and performance and capacity management.

**Chris Cooper** is a certified IT architect specializing in Infrastructure Design and Systems Management. His current role is the UK Travel & Transport industry architect with 14 years of IT industry experience. He is a graduate of UWE, Bristol and a member of the IET.

**Dave Jewell** is a Consulting IBM Specialist in the US. He has worked at IBM for over 32 years in a variety of IT development, testing and performance roles, and has specialized in performance testing, analysis and engineering roles over the last 20 years.

**Martin Jowett** is a Distinguished Engineer and Executive IT Architect specializing in the area of Performance Engineering and System Architecture and Design. Over the last fifteen years, he has worked as lead performance architect on many of the largest and most challenging of the IBM UK Systems Integration Programs and as performance consultant to many of the most significant IBM accounts, both within and outside the UK. He is currently Lead Performance Architect within GBS UKI. He is a recognized leader across the world wide IBM performance community, currently co-lead of the worldwide Performance & Capacity Community of Practice, with a membership of nearly 2000 practitioners, and global owner of the Architecting for Performance Class. He has played a leadership role in both the development and deployment of Performance Engineering related content within the IBM Global Services Method and accompanying education courses. He is a member of the IBM Academy of Technology.

**Andrew McDonald** practices and teaches Performance Architecture. He is based in Sydney, Australia. He has nine years experience in Performance Engineering and 22 years in the IT industry. He holds a degree in Computer Science from Charles Sturt University in Bathurst. His areas of expertise are Performance Architecture and Performance Triage.

**Richard Metzger** is a Senior Software engineer in Germany. He has 16 years of experience in performance engineering and business process management products. He joined IBM in 1984 after obtaining a Master of Science degree in Mathematics/Physics from the University of Heidelberg in Germany. He has held various positions in support, international projects and development, and has contributed to various publications, studies, and conferences.

**Bernie Michalik** is a certified IT architect based in Toronto. He has over 20 years of experience in performance engineering on both the application and the infrastructure side of IT systems. He holds a degree in Computer Science from Dalhousie University in Nova Scotia. He is currently working on a large scale Smarter Planet project for a Canadian client.

**Mattias Persson** is a RFID Solution Architect in the Nordics working for IBM Global Business Services®. He has several years of experience with RFID and a background as a Performance Software Engineer in IBM Labs. He holds a Master of Science degree in Computer Science and his area of expertise include RFID, Performance & Capacity, SOA, and BPM across multiple type of industries.

**William Portal** is an IT Architect in the UK. He has 12 years of experience in performance engineering and operational architecture and over 25 years in the industry. He holds a degree in Mathematical Economics from the University of Birmingham in the UK. His areas of expertise include telecommunications and the public sector.

**Richard Raszka** is a Certified IT Specialist working with IBM Global Business Services in the Australia/New Zealand Architecture practice. He holds degrees in Mechanical Engineering and Mathematical Sciences from Adelaide University and an Master of Business Administration in Technology Management from Monash University. Richard has extensive experience in IT since 1985, working in the industries of engineering software, defence simulation, telecommunications, insurance, and banking. Richard is a technologist who has been involved in innovative “green” fields complex systems integration solutions using middleware technologies, such as WebSphere, integrating new systems with existing systems. He specializes in the disciplines of Performance Engineering/Architecture, Application Lifecycle Management (ALM), Complex Systems Integration, SOA Design and Governance, Enterprise Architecture, and Integration Architecture. He has co-authored a IBM Redbooks® publication on IBM Rational Application Developer and another on SOA Governance.

**Elisabeth Stahl** created the project to write this paper. She is Chief Technical Strategist for the IBM Systems and Technology Group and has been working in systems performance for over 25 years. She holds a Bachelor of Arts degree in Mathematics from the University of Pennsylvania and an Master of Business Administration degree from NYU. Elisabeth is a The Open Group Distinguished IT Specialist, an IEEE Senior Member, and member of the IBM Academy of Technology.

**Damian Towler** is a global thought leader in IT Service Management (ITSM), recognized by his election to the IBM Academy of Technology. Within ITSM, he specializes in Business Strategy Demand Management, IT Application Performance Engineering, and IT Infrastructure Capacity Management. He attended the University of Queensland and holds an Arts degree with a double major in Philosophy and single major in Computer Science. Since 1984, he has worked within IT as a programmer, analyst, database administrator, IT service manager, and educator. He has covered the entire IT solution life cycle, including business requirements, architecture, design, testing, implementation, running, and optimizing. His deep industry experience includes the resource sector, telecommunications, banking, transport, and utilities.

**Klaus Waltmann** is a senior certified IT Architect based in Germany. He has 20 years of experience in IT solutions, of which he spent 15 years working on the performance of large integrated telecommunication solutions. He holds a degree in Computer Systems Engineering from the cooperative state university Baden-Wuerttemberg. His areas of expertise include application of performance engineering from identifying key volumetric drivers until performance testing as well as analysis of bottlenecks in live solutions with a focus on UNIX platforms.

## Acknowledgements

The authors would like to thank the following contributors and reviewers for reading draft versions of this paper and providing valuable comments:

Dave Bachmann  
**STSM, IBM Security Performance, IBM Austin**

Robert R. Carter  
**Master Certified IT Specialist, IBM Boulder**

David Challoner  
**GTS Performance and Capacity Specialist, IBM United Kingdom**

James Chaloner  
**Associate partner in GBS and UK rail leader**

**Stewart Garnett  
Performance Architect, IBM United Kingdom**

**Pam Isom  
GBS Cloud Solutions Practice, IBM Denver**

**Edwardo Martinez  
Senior IT Specialist: Performance and Capacity Management, IBM San Francisco**

**Grahame Oakley  
Systems Management Specialist: Performance and Capacity Management, IBM Australia**

**Hassan Shazly  
IBM z/OS® Software Developer: IM.Enterprise Content Management, IBM Columbia**

**John Shepherd  
Performance Architect, IBM United Kingdom**

**Kevin Yu  
Consulting IT Specialist, IBM Canada**

Thanks to Mike Ebbers and the editing/publishing team of the IBM International Technical Support Organization (ITSO) in Poughkeepsie.

## **Now you can become a published author, too!**

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:  
[ibm.com/redbooks/residencies.html](http://ibm.com/redbooks/residencies.html)

## **Stay connected to IBM Redbooks**

- ▶ Find us on Facebook:  
<http://www.facebook.com/IBMRedbooks>
- ▶ Follow us on Twitter:  
<http://twitter.com/ibmredbooks>
- ▶ Look for us on LinkedIn:  
<http://www.linkedin.com/groups?home=&gid=2130806>

- ▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:  
<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>
- ▶ Stay current on recent Redbooks publications with RSS Feeds:  
<http://www.redbooks.ibm.com/rss.html>

# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.*

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

## COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

This document REDP-4762-00 was created or updated on July 14, 2011.

Send us your comments in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:  
[ibm.com/redbooks](http://ibm.com/redbooks)
- Send your comments in an email to:  
[redbooks@us.ibm.com](mailto:redbooks@us.ibm.com)
- Mail your comments to:  
IBM Corporation, International Technical Support Organization  
Dept. HYTD Mail Station P099  
2455 South Road  
Poughkeepsie, NY 12601-5400 U.S.A.



## Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

IBM, the IBM logo, ibm.com, Let's Build A Smarter Planet, Smarter Planet and the planet icons are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®	Rational®	Smarter Healthcare™
CICS®	Redbooks®	Smarter Planet™
Cognos®	Redpaper™	Smarter Traffic™
DB2®	Redbooks (logo)	System x®
Global Business Services®	Service Request Manager®	Tivoli®
IBM®	Smarter Banking™	WebSphere®
IMS™	Smarter Cities™	z/OS®
Maximo®	Smarter Energy™	

The following terms are trademarks of other companies:

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.