Understanding Availability Management: Concepts and Options
(PRESENTATION WHITE PAPER)

January 2000

Mike Bonett
Systems Management Technical Support
IBM Corporation, Advanced Technical Support
Gaithersburg, MD
bonett@us.ibm.com

Preface

The information contained in this document has not been submitted to any formal IBM test and is distributed on an "as is" basis **without any warranty either expressed or implied.** The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM licensed program in this publication is not intended to state or imply that only IBM's program may be used. Any functionally equivalent program can be used instead.

Any information in this document concerning non-IBM products was obtained from the suppliers of those products or from their published announcements. IBM has not tested these 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.

Any pointers in this publications to external Web Sites are provided for convenience only and do not in any manner serve as an endorsement of these web sites.

The information in this publication is not intended as the specification of any programming interfaces.

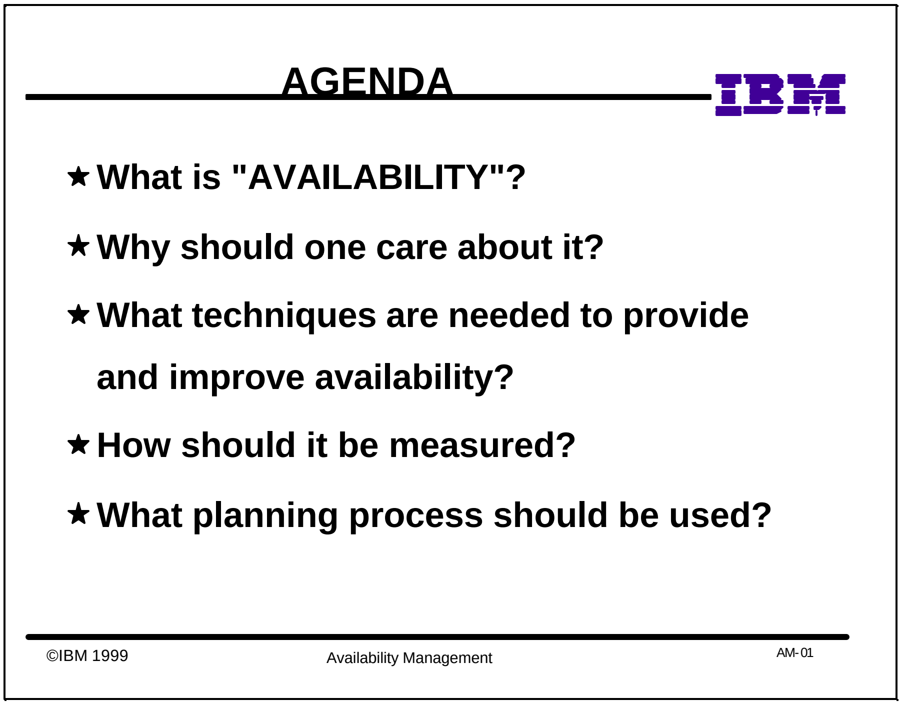
Questions or comments about this publication should be sent via the Internet to **bonett@us.ibm.com**.

Introduction

Availability can be hard to define - but everyone knows when it is not being provided. Unavailability is visible not just to the organization, but to the organization’s customers - as well as the media. Improving availability doesn’t happen just by installing hardware or software. A planned design, along with implementing specific improvement techniques, must be done to get results. This paper provides an overview of Availability Management concepts and options. It identifies the various definitions of availability, the techniques that can be applied to improve availability, how it should be measured, and the planning actions for cost-effective availability improvement.

This paper is to be used with the "Understanding Availability Management: Concepts and Options” presentation that the author has given at a number of IBM customer seminars. The charts, tables, and figures used in this paper are taken from that presentation.

Page AM-01



AGENDA

- ★ What is "AVAILABILITY"?
- ★ Why should one care about it?
- ★ What techniques are needed to provide and improve availability?
- ★ How should it be measured?
- ★ What planning process should be used?

©IBM 1999 Availability Management AM-01

Availability. This word can mean many things when applied to Information Technology (I/T). There is constant talk of using I/T within an organization to improve service, competitiveness, and/or revenue by providing better service and higher availability. However, the word can be misused so that its true meanings, applicability, and quantification of its impact can be lost. This is especially true when a crisis occurs and emotions are running high.

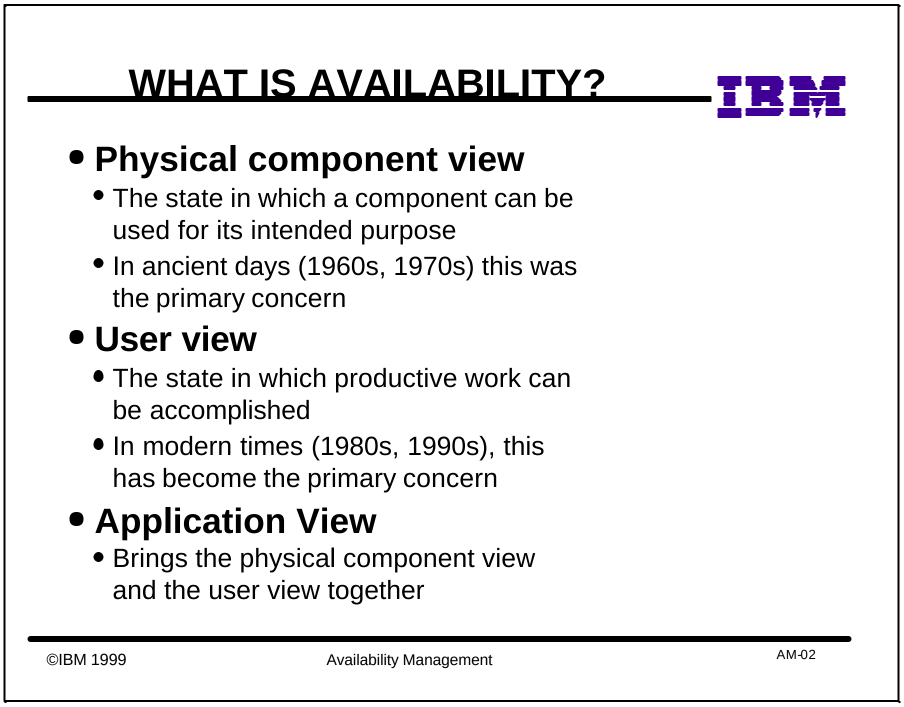
This document provides a basic introduction to Availability Management concepts. It will provide a definition - actually several definitions - of what availability is, to help an organization determine the type of availability that is important to them. It will show evidence of its importance - since the lack of availability can bring a business to its knees. It will review the techniques available for improving availability. All of them will not be applicable in a given situation, but are worth consideration.

Availability Management is a management discipline within Systems Management. Therefore, metrics are needed to assess what is being achieved and where improvements can be made. This document will review information for creating and using those metrics.

Finally, availability doesn’t just “happen”. It takes sound planning, implementation, and evaluation to support a defined availability objective. The major activities to include in an overall planning process will be highlighted.

This paper is not intended to cover these topics in great detail. The purpose is to provide an introductory overview for those new to the world of availability, who now have to manage, measure, or work within the planning and implementation process. The objective her is to provide the “first steps” to understand availability management.

Page AM-02



WHAT IS AVAILABILITY?

- **Physical component view**
  - The state in which a component can be used for its intended purpose
  - In ancient days (1960s, 1970s) this was the primary concern
- **User view**
  - The state in which productive work can be accomplished
  - In modern times (1980s, 1990s), this has become the primary concern
- **Application View**
  - Brings the physical component view and the user view together

©IBM 1999 Availability Management AM-02

Availability can be defined based on the “view” of the environment one takes. There are three major views to consider:

1. **The Physical component view.** Availability means that the physical component - either hardware or software - is in a state where it can be used for its intended purpose. This is usually referred to as the “up” state. If the server is turned on, it is considered “up”. If the operating system reaches a point where it can begin processing work, it is considered “up”. If the network interface is active, it is considered “up”. Conversely, when the “up” state does not exist, the component is “down”, or unavailable.

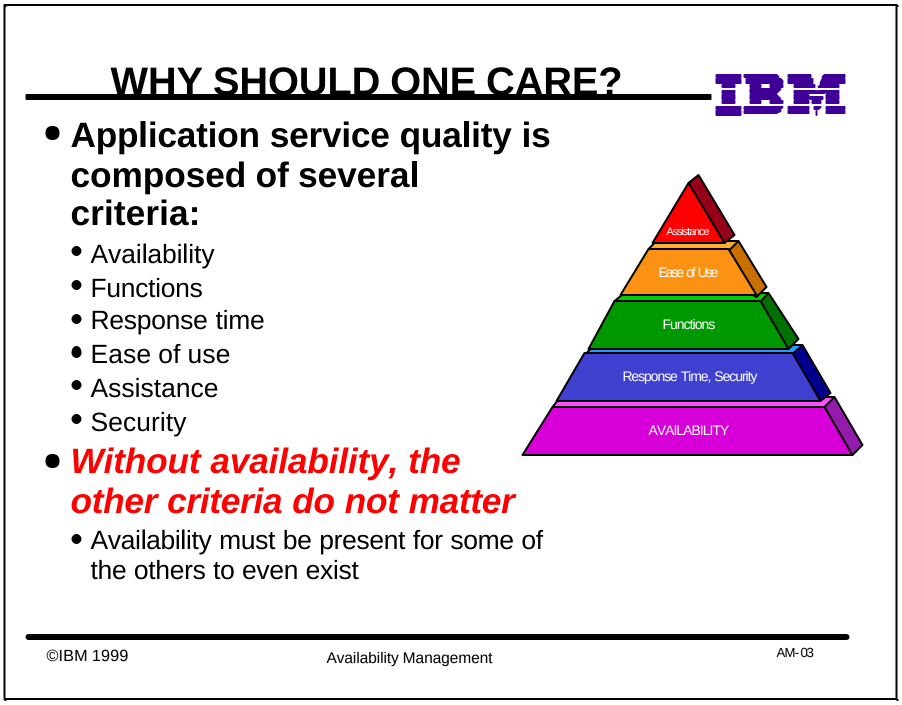
   When data processing first began, there was little separating the components from the work to be processed. The primary concern was if the component was working. It was assumed that, if the component was working, the work was running, and the users could access that work.

2. **The User View**. As information processing grew and became more integrated in the workflow of companies, the definition of availability began to change to take more of a user perspective. The number of components in the infrastructure grew, and workload processing moved from batch to online. No longer could one or two components accurately reveal if online work was actually getting done. Users became more vocal when they were not able to use the systems. Availability in this view means that the

environment is in a state where productive user work can be accomplished. This is commonly referred to as “availability from the user perspective”. The criteria is no longer just “is the component up?”, but moves to “is the user able to do their work?”.

3. **The Application View**. This view brings together the component and user views. Applications today, especially with the prevalence of client server and e-business, consist of a number of components and span multiple technologies. When all of the required application components are working together as designed, the user is able to do productive work. Availability in this view means that all required application components are active, **and** productive user work is able to occur. This type of availability is more complex to manage; however, it gives the most accurate reflection of how availability is helping or hurting an I/T environment.

PAGE AM-03



WHY SHOULD ONE CARE?

- **Application service quality is composed of several criteria:**
  - Availability
  - Functions
  - Response time
  - Ease of use
  - Assistance
  - Security
- ***Without availability, the other criteria do not matter***
  - Availability must be present for some of the others to even exist

©IBM 1999 Availability Management AM-03

IBM
Assistance
Ease of Use
Functions
Response Time, Security
AVAILABILITY

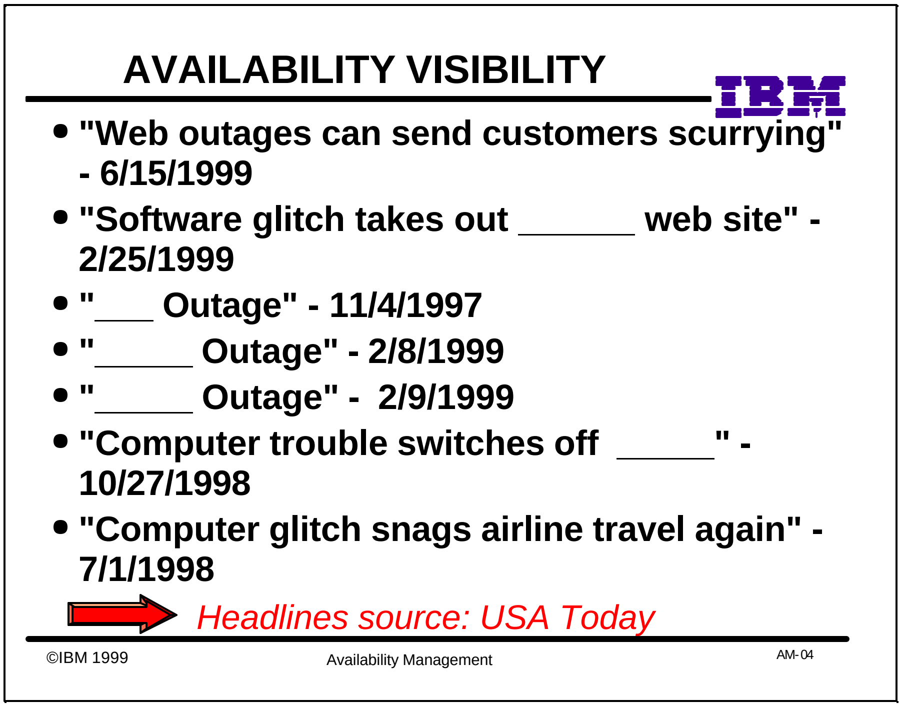
Applications are what users use to get their work done in a productive manner. Applications can also bring in revenue to the company. Application service quality is important, since higher levels of service can result in higher user satisfaction, increased user productivity, and/or increased revenue.

Application service quality is composed of several criteria. They include:

- Availability, as has been discussed on the previous page.
- Application functions to accept and process the user requests
- Response time when using the application functions, or returning requests
- Ease of using the application, including the ability to quickly learn its functions
- Assistance provided when help is needed
- Security of information access

For all service quality criteria, availability is the foundation. Without availability it is impossible to even judge or measure the quality of the other criteria. Without availability many of these criteria cease to exist. Therefore, attention need to be paid to availability.

PAGE AM-04



AVAILABILITY VISIBILITY

- "Web outages can send customers scurrying" - 6/15/1999
- "Software glitch takes out ______ web site" - 2/25/1999
- "___ Outage" - 11/4/1997
- "_____ Outage" - 2/8/1999
- "_____ Outage" - 2/9/1999
- "Computer trouble switches off _____" - 10/27/1998
- "Computer glitch snags airline travel again" - 7/1/1998

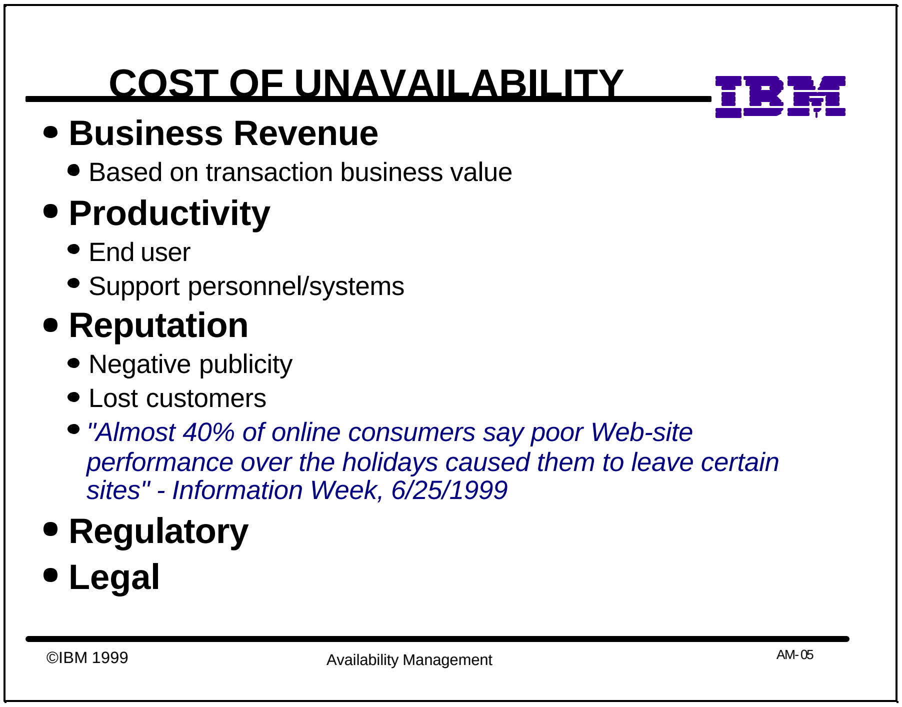
Headlines source: USA Today

©IBM 1999 Availability Management AM-04

Here is another reason to care about availability: the unwanted visibility when system and application outages occur. Application users are not just within the company; they are outside the company, as consumers of or suppliers to the company's offerings. Unavailability in today's world can end up being exposed beyond the company's walls.

For example, a search of the USA Today article archive resulted in many headlines related to availability - all negative. A sample is shown here. The company names have been removed to avoid further embarrassment. This is the inherent risk in not managing availability properly.

PAGE AM-05



COST OF UNAVAILABILITY

- **Business Revenue**
  - Based on transaction business value
- **Productivity**
  - End user
  - Support personnel/systems
- **Reputation**
  - Negative publicity
  - Lost customers
  - *"Almost 40% of online consumers say poor Web-site performance over the holidays caused them to leave certain sites" - Information Week, 6/25/1999*
- **Regulatory**
- **Legal**

©IBM 1999 Availability Management AM-05

The costs associated with unavailability impact the bottom line in several ways:

- Business revenue. This is based on the value of a business transaction for an application, or set of applications. As more revenue-producing activities depend upon applications, this number gets easier to quantify.

- Productivity. This affects two areas. First, the user of the application must wait to get his or her job done, and may not be able to do anything else while their application(s) are unavailable. This will vary by job role. A call center employee will be impacted much more than an administrator. A call center employee cannot do their job if the call center application is unavailable, while the administrator may be able to do other activities not dependent on the applications. However, there will always be some impact to the user.

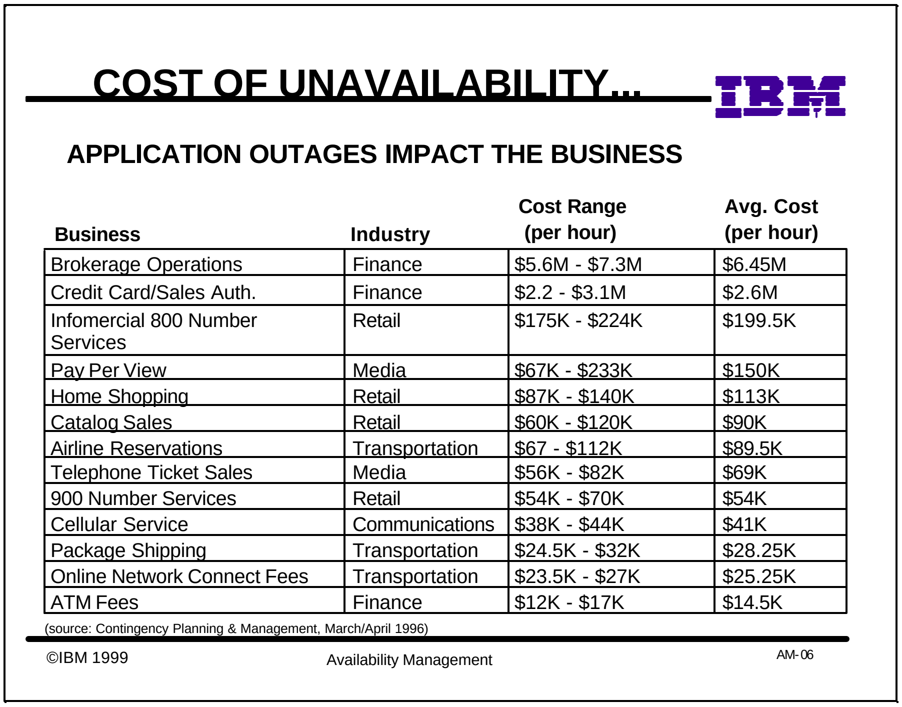
  Second, unavailability has to be resolved, and work that was not run or delayed still has to be processed. This places an added burden on support personnel, who must spend their time fighting unavailability fires, and on the systems, which much try to rerun process delayed work.

w Reputation. Users can vote with their checkbooks when unavailability occurs. The quote from the Information Week article demonstrates this. Negative publicity can result in the loss of customers, which also mean loss of revenue.

w Regulatory. Unavailability may cause missed deadlines in providing required information. In some industries, such as banking, this can result in fines and penalties.

w Legal. The impact of unavailability can expose a company to unwanted legal action from those affected. The Y2K computer “bug” situation is a classic example of a situation where companies and users might consider taking legal recourse if it causes their applications to be unavailable.

PAGE AM-06



COST OF UNAVAILABILITY...

IBM

APPLICATION OUTAGES IMPACT THE BUSINESS

Business	Industry	Cost Range (per hour)	Avg. Cost (per hour)
Brokerage Operations	Finance	$5.6M - $7.3M	$6.45M
Credit Card/Sales Auth.	Finance	$2.2 - $3.1M	$2.6M
Infomercial 800 Number Services	Retail	$175K - $224K	$199.5K
Pay Per View	Media	$67K - $233K	$150K
Home Shopping	Retail	$87K - $140K	$113K
Catalog Sales	Retail	$60K - $120K	$90K
Airline Reservations	Transportation	$67 - $112K	$89.5K
Telephone Ticket Sales	Media	$56K - $82K	$69K
900 Number Services	Retail	$54K - $70K	$54K
Cellular Service	Communications	$38K - $44K	$41K
Package Shipping	Transportation	$24.5K - $32K	$28.25K
Online Network Connect Fees	Transportation	$23.5K - $27K	$25.25K
ATM Fees	Finance	$12K - $17K	$14.5K

(source: Contingency Planning & Management, March/April 1996)

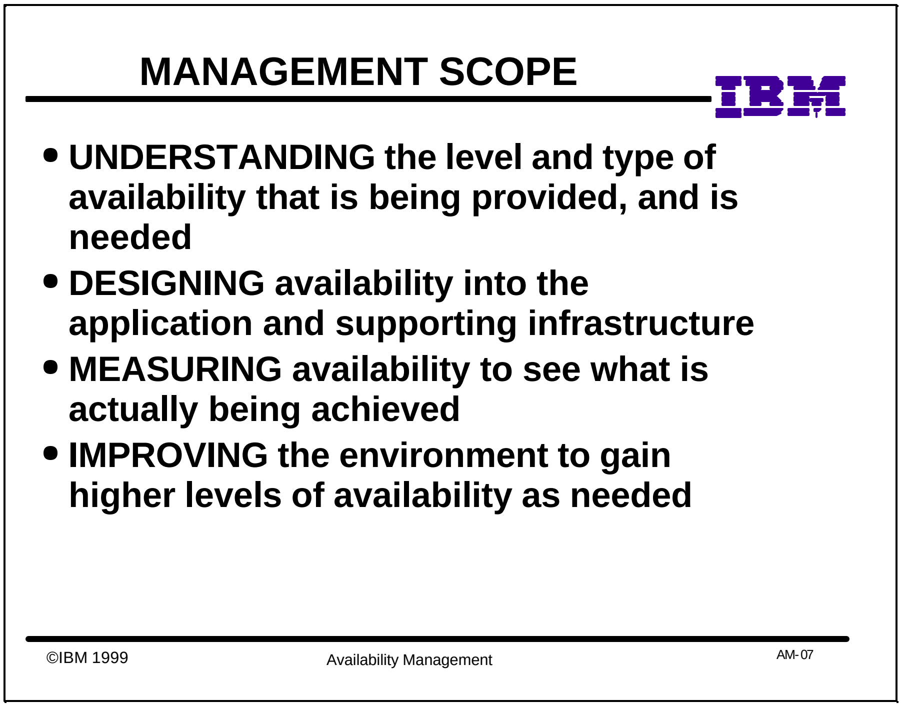
©IBM 1999 Availability Management AM-06

This table. published in the Contingency Planning and Management March/April 1996 issue, summarizes the results of a study that quantified the cost of unavailability.

The “Cost Range (per hour)” column shows, for the companies surveyed within an industry, the minimum and maximum cost of an outage for the identified application reported. The “Average Cost (per hour) column shows the average of the reported hourly costs.

From the information in the table, It is clear that unavailability does not carry a trivial cost. This increases the need for availability management.

PAGE AM-07



MANAGEMENT SCOPE

- **UNDERSTANDING the level and type of availability that is being provided, and is needed**
- **DESIGNING availability into the application and supporting infrastructure**
- **MEASURING availability to see what is actually being achieved**
- **IMPROVING the environment to gain higher levels of availability as needed**

©IBM 1999 Availability Management AM-07

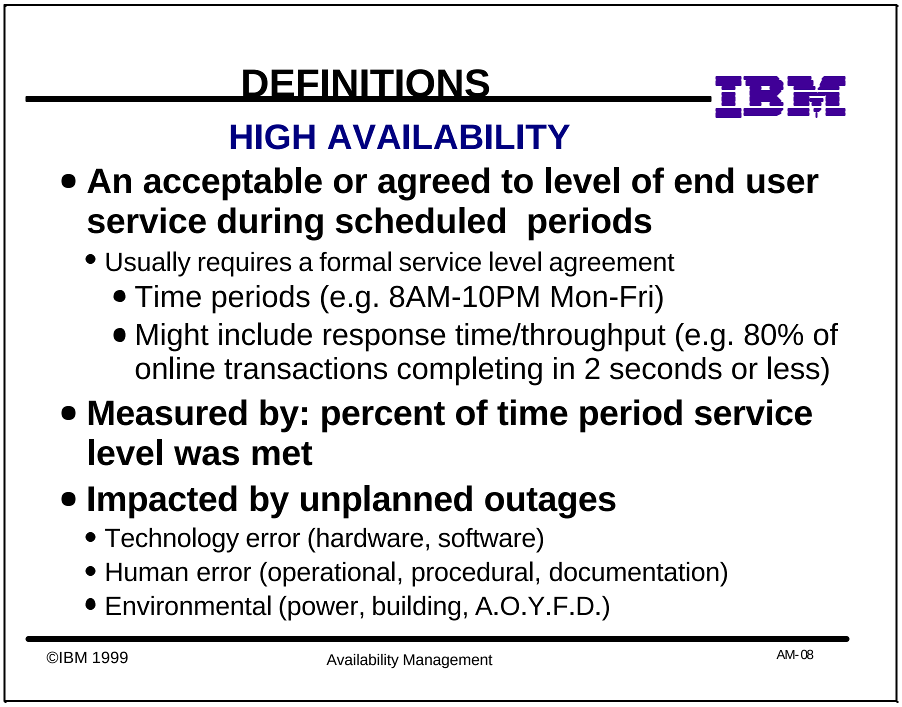
What must the “managing” of availability encompass? The management scope applied must include the following:

w An objective understanding of the existing availability situation. Before any availability techniques can be applied, the current level of availability being achieved, along with the required availability type and level, must be known.

- The use of various techniques to design availability into the application and support infrastructure, including the hardware, software, and support processes.
- Taking quantifiable, objective measurements that can show the true availability of applications.
- Improving the environment, based on information from the measurements, and applying additional techniques, to gain higher levels of availability.

The important thing to remember is that the management scope is not applied as a “straight line”. It will be a circle, or even a spiral, of repeating these activities to meet and maintain the availability objectives.

PAGE AM-08



DEFINITIONS

HIGH AVAILABILITY

- **An acceptable or agreed to level of end user service during scheduled periods**
  - Usually requires a formal service level agreement
    - Time periods (e.g. 8AM-10PM Mon-Fri)
    - Might include response time/throughput (e.g. 80% of online transactions completing in 2 seconds or less)
- **Measured by: percent of time period service level was met**
- **Impacted by unplanned outages**
  - Technology error (hardware, software)
  - Human error (operational, procedural, documentation)
  - Environmental (power, building, A.O.Y.F.D.)

©IBM 1999 Availability Management AM-08

To determine what availability is being currently provided, as well as what is needed, there are three types of availability that have to be defined: High Availability, Continuous Operations, and Continuous Availability.

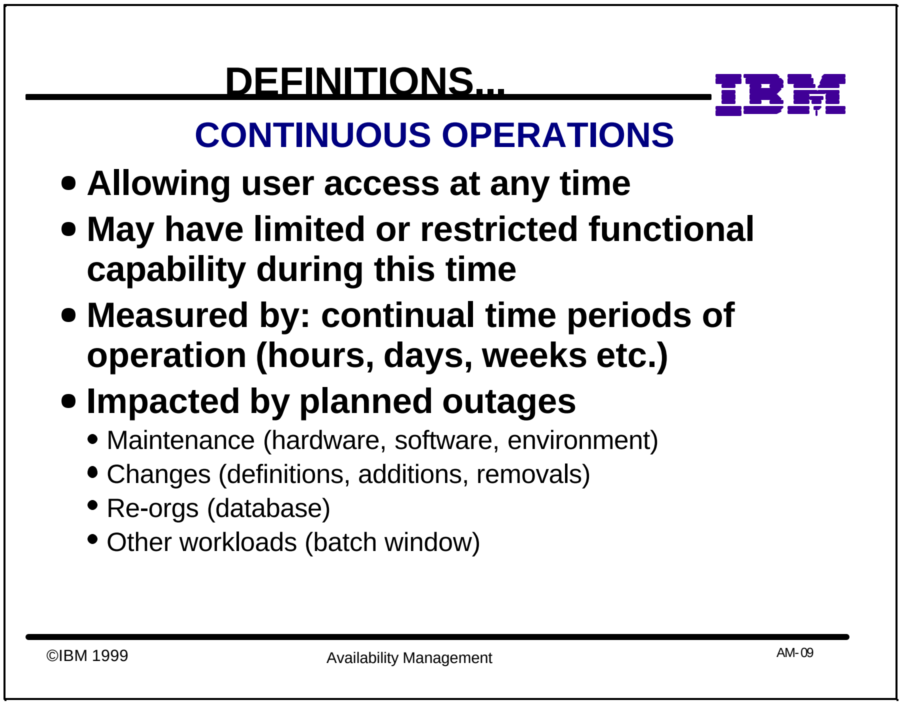
High Availability means **an acceptable or agreed to level of end user service during scheduled periods**. To provide high availability an acceptable or agreed to objective between the providers of service and the end users is needed. This is usually included in a documented **service level agreement**.
Within the service level agreement, two items serve to refine this measurement:

1. The time period for high availability. Anything that happens outside of this time period does not affect the availability measurements. It is not necessary to main high availability outside of the scheduled period.
2. The criteria that indicates high availability is being met. This is increasingly becoming a workload performance or thruput based criteria - for example, the number of transactions, the majority of transactions being below a response time level, etc.

High Availability is normally measured by the percentage of the measured time period that the service level criteria was met. For example, if the time period is 12 hours, and during 11 of those hours the criteria is met, the high availability achievement is 11/12 = 91.67%.

The biggest impact to high availability will be unplanned outages. Planned outages can be scheduled outside of the high availability period. Unplanned outages, however, can occur during the high availability period. Technology, human, and environment errors and actions will be the cause of these unplanned outages.

PAGE AM-09



DEFINITIONS...

CONTINUOUS OPERATIONS

- **Allowing user access at any time**
- **May have limited or restricted functional capability during this time**
- **Measured by: continual time periods of operation (hours, days, weeks etc.)**
- **Impacted by planned outages**
  - Maintenance (hardware, software, environment)
  - Changes (definitions, additions, removals)
  - Re-orgs (database)
  - Other workloads (batch window)

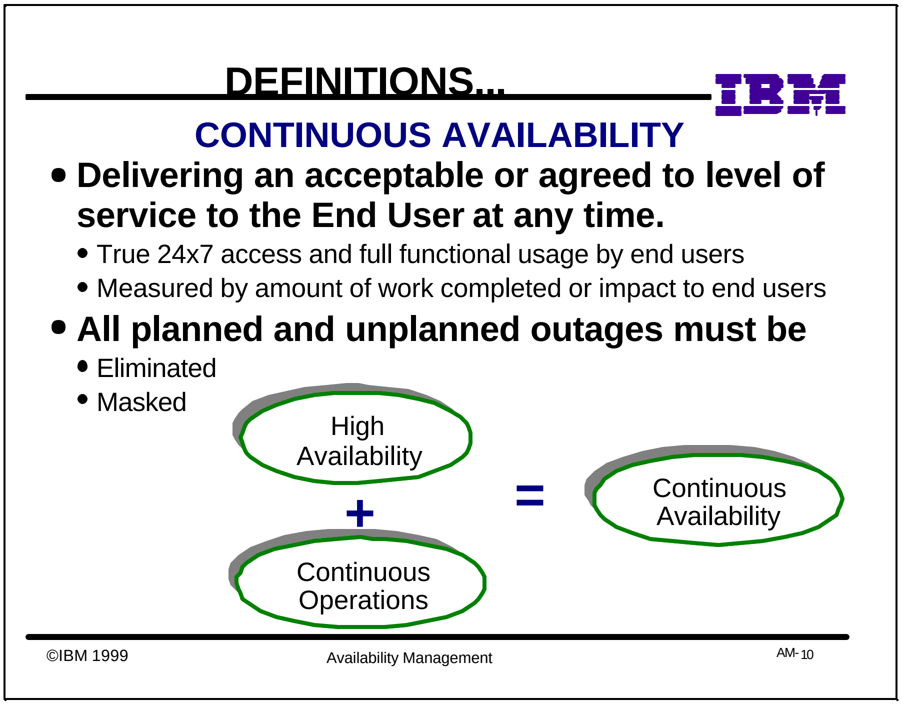
©IBM 1999 Availability Management AM-09

Continuous Operations means that the user (human or workload) has access at any time. However, during periods of this access not all functions may be available. For example, certain data may only be in read-only access, or certain software functions may be disabled. There may or may not be service level criteria associated with a continuous operations environment. If there is, it is usually limited to significant interruptions, such as the number of IPLs permitted during a time period.

Continuous operations are measured by the length of time is has been operating. For example, on many application platforms commands can be issued to determine when the platform was last started. The response indicates the last start time and may also provide how long it has been running - for hours, weeks, days, even months. However, that does not mean that, during this time, all available functions on the platform have always been available. This measurement does accurately reflect what a user or workload is actually experiencing.

Planned outages are the biggest impact to continuous operations. Maintenance of hardware and software, implementing changes and upgrades, and database reorganizations are examples of activities that might require a disruptive outage. Outages to avoid conflicts for other workloads may prevent an application being continually operational; the classic example is having to shut down online applications to allow batch processing to take place. While unplanned outages can affect continuous operations, they are relatively minor compared to the amount of time a planned outage might require.

Page AM-10



DEFINITIONS...

CONTINUOUS AVAILABILITY

- **Delivering an acceptable or agreed to level of service to the End User at any time.**
  - True 24x7 access and full functional usage by end users
  - Measured by amount of work completed or impact to end users
- **All planned and unplanned outages must be**
  - Eliminated
  - Masked

High Availability + Continuous Operations = Continuous Availability

©IBM 1999 Availability Management AM-10

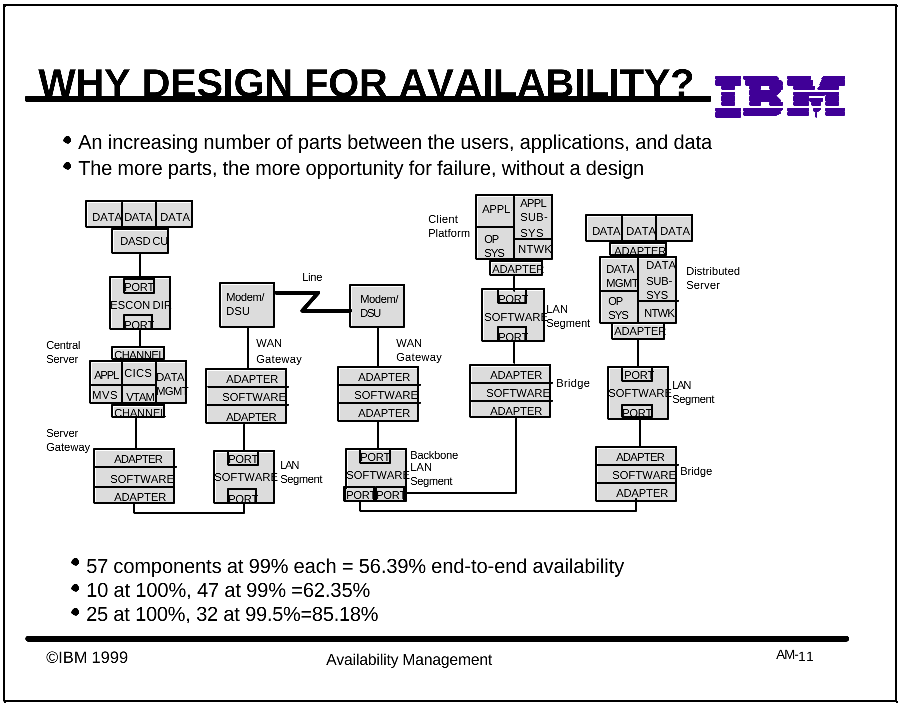
Continuous Availability combines the best of both worlds from Continuous Operations and High Availability. This is sometime written as the formula **CA = HA + CO**. It means that acceptable or agreed to service, as documented in a service level agreement, is being provided to users at all times.

Continuous availability is a **true** ’24 by 7” environment. All functions must be available to users and workloads. And these functions must perform at a level to meet the documented service level criteria. While continuous availability is measured the same as high availability - except that the time period is 24 X 7 - measurement beyond just a percentage of time is needed. For example, the number of users (or user minutes) or the amount of work completed as compared to the “24 by 7” level of users/user minutes and work completed would provide a more accurate measurement for continuous availability.

Since continuous availability combines both high availability and continuous operations, it will be impacted by both planned and unplanned outages. These outages must be either eliminated, or masked, to achieve high levels of availability. Techniques to address this will be discussed later in this presentation.

An important question to answer, when trying to provide availability for an environment, is: which TYPE of availability is needed (or will be selected)? While continuous availability is a worthwhile goal, the investment to achieve it will be far greater than the investment for high availability or continuous operations. When someone says they want “24 by 7” availability, it is important to define their meaning for that term, to determine if they really are asking for continuous availability.

.



WHY DESIGN FOR AVAILABILITY?

- An increasing number of parts between the users, applications, and data
- The more parts, the more opportunity for failure, without a design

- 57 components at 99% each = 56.39% end-to-end availability
- 10 at 100%, 47 at 99% =62.35%
- 25 at 100%, 32 at 99.5%=85.18%

©IBM 1999 Availability Management AM-11

Availability - be it high availability, continuous operations, or continuous availability - does not just “happen”. It must be designed. Today’s application environments are supported by a large number of components - “moving parts” - to connect users, applications, and data. Individual parts may be highly reliable, but more parts means more potential points of failure, leading to reduced availability.

This is a picture of a simple application environment - a user on a local area network accessing an application that resides on a distributed server. The distributed server must have access to functions and data on a central server, to support incoming user requests. For this simple environment there are a minimum of 57 components. If each component achieves a reliability of 99%, the best overall availability expected for this environment is:

.99 * .99 * .99 * .... (57 times) = 56.39%

By using design techniques, some components can be made to provide 100% reliability for the application. If this were done to 10 components, the overall availability increases to:

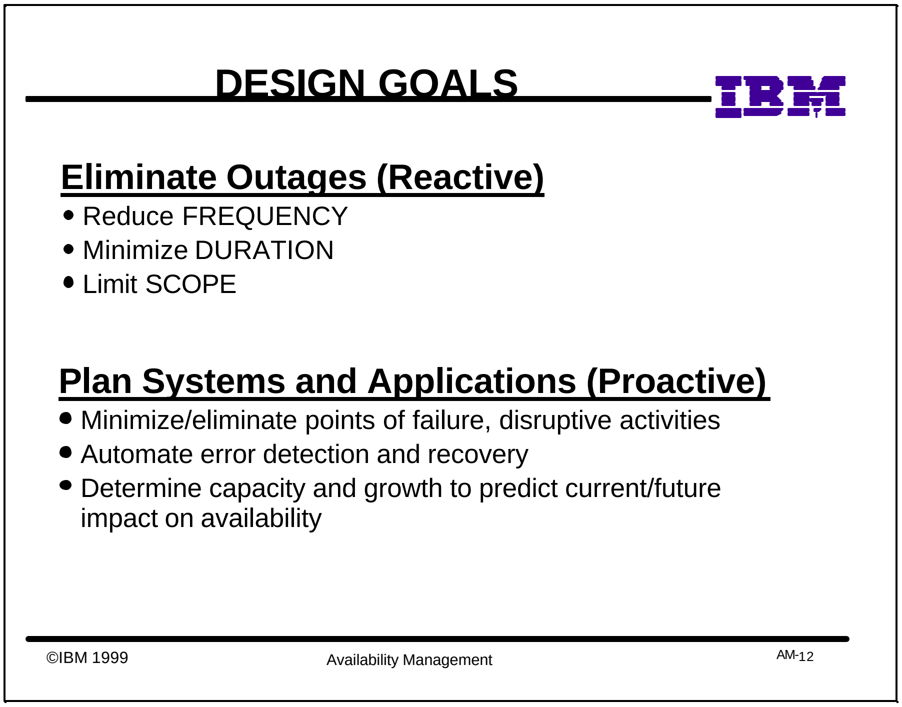
1* 1 * 1 * ... (10 times) * .99 *.99 *.99 * ... (47 times) = 62.35%

If improvements were made to 25 components to make them appear 100% reliable, and the other 32 were improved to 99.5% reliability, the overall availability increases again to:

1* 1 * 1 * ... (25 times) * .995 *.995 *.995 * ... (32 times) = 85.18%

These are estimates, but illustrate the impact that designing for availability can have. Never forget that availability is a system-wide process - it will depend upon more than just a single component.

PAGE AM-12



DESIGN GOALS

Eliminate Outages (Reactive)

- Reduce FREQUENCY
- Minimize DURATION
- Limit SCOPE

Plan Systems and Applications (Proactive)

- Minimize/eliminate points of failure, disruptive activities
- Automate error detection and recovery
- Determine capacity and growth to predict current/future impact on availability

©IBM 1999 Availability Management AM-12

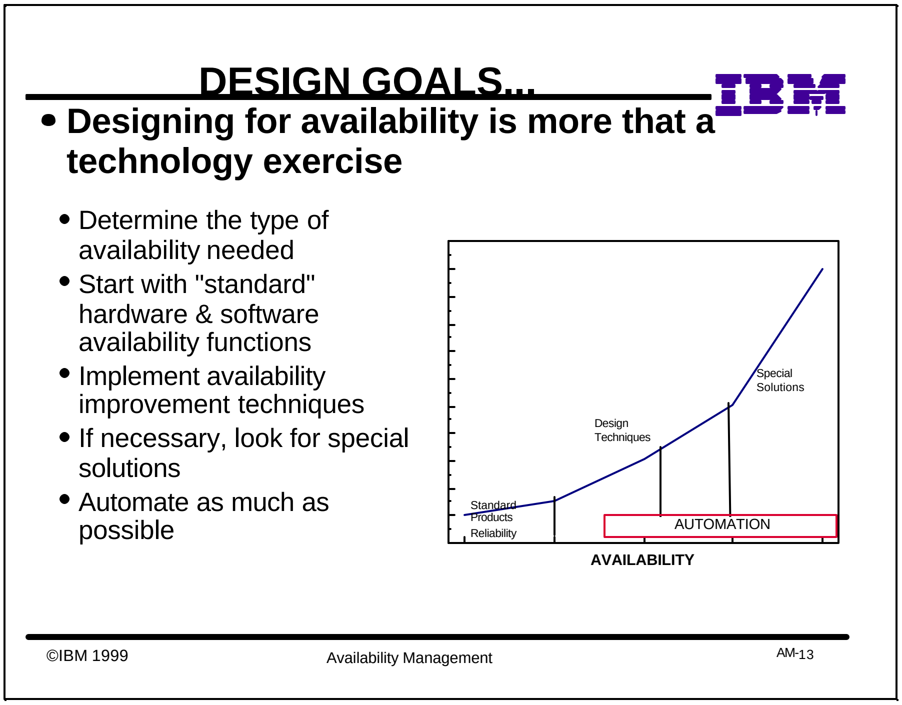
There are two goals to consider when designing for availability. The first goal is the elimination of outages by reacting to and addressing the ones that are currently occurring, both planned and unplanned.
Eliminating existing outages can be done in three ways:

1. Reduce their frequency. Determine the root cause of an outage, and attempt to correct the conditions so that this type of outage does not happen again.
2. Minimize the duration of an outage. If it cannot be totally eliminated, attempt to detect it as quickly as possible, and take actions to reduce its length.
3. Limit the scope of an outage, by reducing the number of users or workload affected by an outage.

The second set is to plan systems and applications to address potential future outages. This is best done before the system or application in placed into production, via planning activities as they are developed, tested and implemented. The planning should anticipate that situations that will lead to outages will occur, and should then have the design include techniques to address them. Something as simple as going through an outage scenario and documenting the required recovery actions can make a big difference. It is easier done in a planning phase than trying develop one in the middle of an outage. The planning should address:

w Minimizing or eliminating single points of failure and disruptive maintenance/change activities.
w Using automation faster error detection and recovery.
w Forecasting capacity and growth to predict the future impact of performance on availability.

PAGE AM-13



DESIGN GOALS...

- **Designing for availability is more that a technology exercise**
  - Determine the type of availability needed
  - Start with "standard" hardware & software availability functions
  - Implement availability improvement techniques
  - If necessary, look for special solutions
  - Automate as much as possible

Standard Products Reliability
Design Techniques
Special Solutions
AUTOMATION
AVAILABILITY

©IBM 1999 Availability Management AM-13

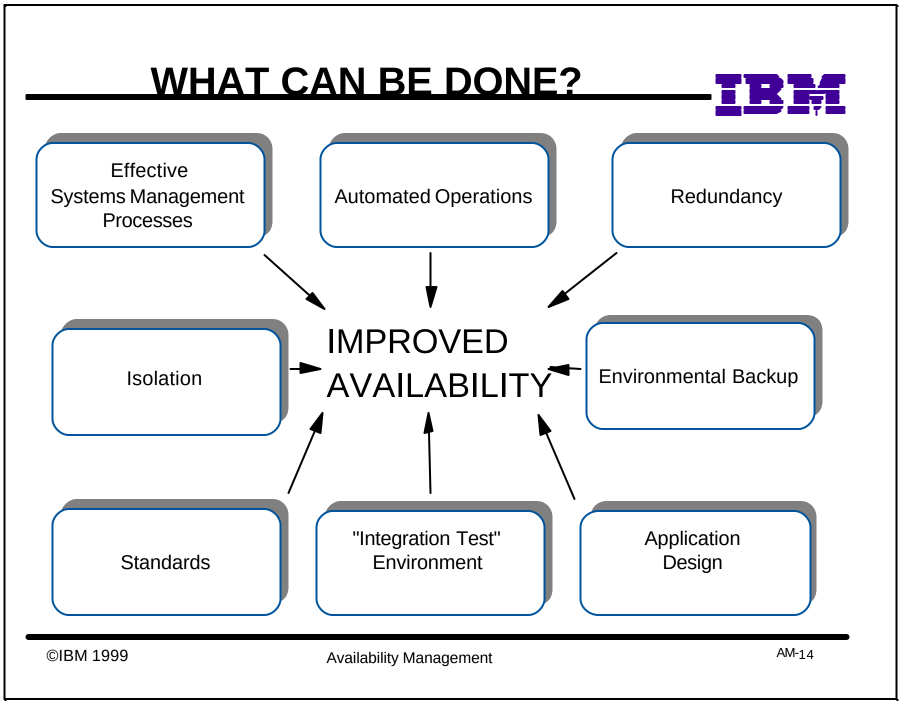
It is easy to fall into the trap that technology is all that is needed to design for availability. For example, “fault-tolerant” is a technology that many assume will provide high or continuous availability on its own. While technology provides the foundation to achieve a “base” level of availability, more work has to done to achieve a specific desired type and level of availability.

Design activities must be done within the context of the type of availability that is to be provided. As mentioned earlier, there are different levels of investment for high availability, continuous operations, and continuous availability.

Components - hardware and software - likely provide some availability functions; these will be the foundation for the initial availability achievement level. Next, look at specific availability improvement techniques to enable higher availability levels. Special solutions - such as fault-tolerant components - may also have to be deployed among the components.

Deploying automation to support the components, improvement techniques, and solutions becomes critical as the complexity and chance for human error increases.

PAGE AM-14



WHAT CAN BE DONE?

Effective Systems Management Processes

Automated Operations

Redundancy

Isolation

IMPROVED AVAILABILITY

Environmental Backup

Standards

"Integration Test" Environment

Application Design

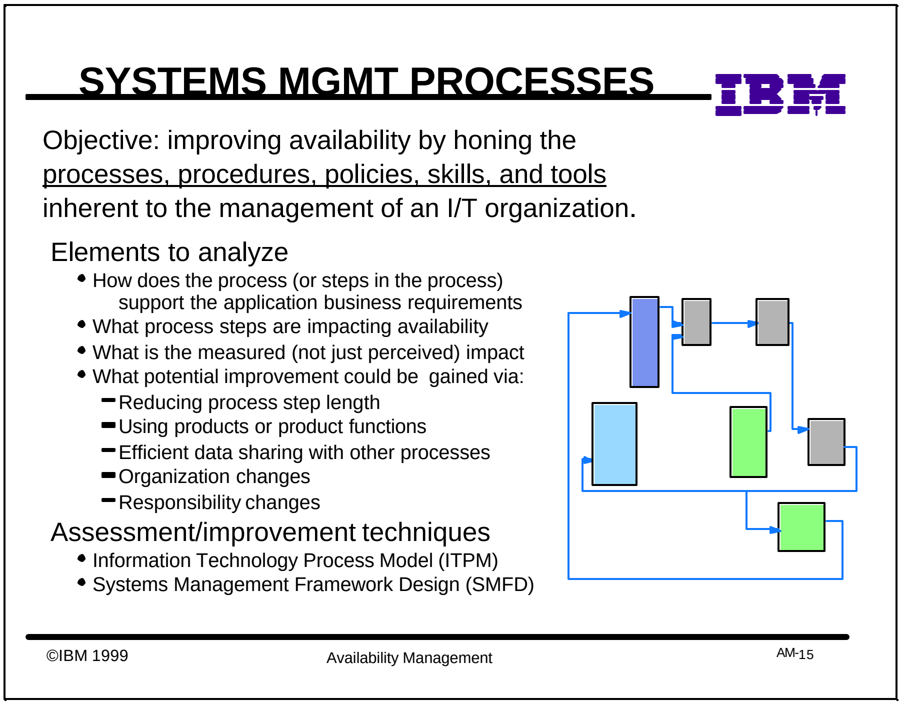
©IBM 1999 Availability Management AM-14

Specific availability improvement techniques can be sued as part of the availability management processed. They fall into one or more of these categories:

- Effective Systems Management Processes
- Automated Operations
- Redundancy
- Isolation
- Environmental Backup
- Standards
- “Integration Test” Environment
- Application Design

Not all of the techniques will apply in every situation. However, each is worth investigating to determine its potential applicability for improving availability. The following pages provide a brief overview of each category, the related techniques, and the potential benefits.

PAGE AM-15



SYSTEMS MGMT PROCESSES

Objective: improving availability by honing the processes, procedures, policies, skills, and tools inherent to the management of an I/T organization.

Elements to analyze

- How does the process (or steps in the process) support the application business requirements
- What process steps are impacting availability
- What is the measured (not just perceived) impact
- What potential improvement could be gained via:
  - Reducing process step length
  - Using products or product functions
  - Efficient data sharing with other processes
  - Organization changes
  - Responsibility changes

Assessment/improvement techniques

- Information Technology Process Model (ITPM)
- Systems Management Framework Design (SMFD)

©IBM 1999 Availability Management AM-15

The road to availability passes through systems management.

Systems management is made up of the processes, procedures, policies, skills, and tools needed to direct the I/T organization to properly support the business. Availability can be the most visible I/T measurement the business outside of I/T sees. Therefore it is very important that systems management supports the business and provides the availability required by the business.

Effective Systems Management Processes are **fundamental** to Availability Management (as well as most other I/T endeavors). Every business has a management structure today with varying degrees of processes. Consequently, just the tuning of these processes may result in sizable availability improvements.

While an exhaustive look at systems management is beyond the scope of this presentation white paper, there are some basic actions that can be taken to determine the impact - good or bad - of systems management on availability, and then identify the necessary actions for improvement.

Systems management processes should always be evaluated to insure they are supporting the business requirements. It is easy for business requirements to change and the processes to remain the same, thus causing conflict. This can even cause steps within a process to have a negative impact on availability. Measuring outages, and seeing how a process was performed during an outage, will identify these steps.

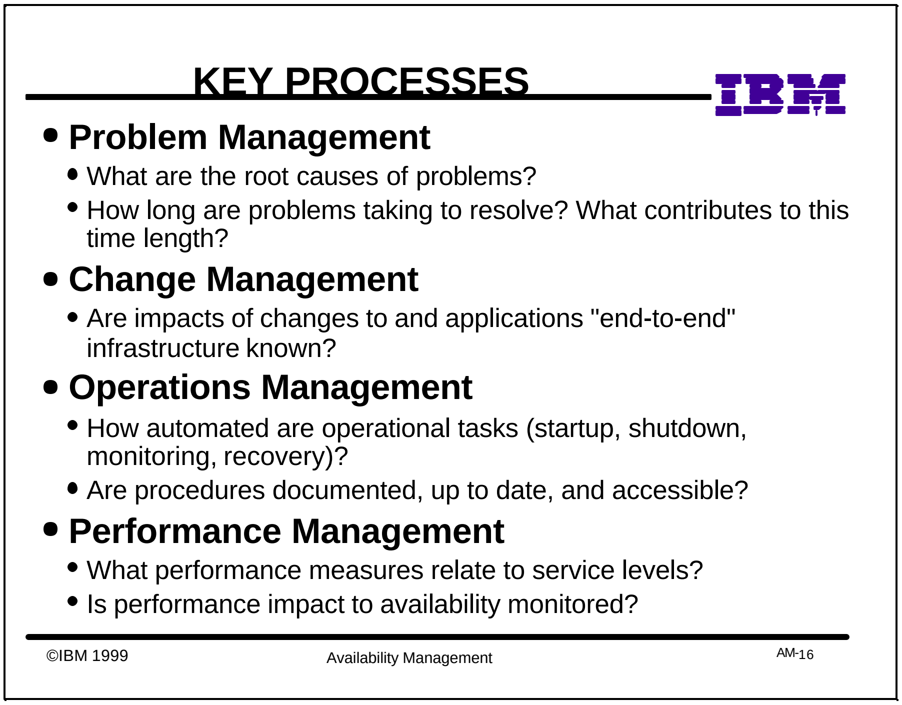
Once identified, potential improvements can be investigated. Benefits can be gained from doing such things as:

- Reducing the length of the process, which can have a corresponding reduction in the related outage time.
- Using products or product functions to increase process efficiency.
- Sharing information with other processes in a more timely and efficient manner (for example, electronically instead of manually).
- Reviewing and changing organization roles and responsibilities. For example, as applications span multiple application platforms and networks, the processes to support the application must also span the organizations supporting the platforms and networks, with clearly defined responsibilities.

A key process that is either overlooked or not kept up to date is that of measuring availability. Many environments still use host-based measurements, collected manually, as the main indicator of availability. However, a truer picture of availability is gained by measuring what the end user actually sees. How to have measurements reflect more what the user sees will be covered later in this white paper.

Formal assessment and improvement techniques for systems management include the Information Technology Process Model (ITPM) and the Systems Management Framework Design (SMFD) methodology. ITPM identifies a set of systems management processes and evaluation criteria to help identify what processes or process steps may need improvement. SMFD carries the assessment further into a design methodology for re-engineering or creating systems management processes that includes the necessary organization and technology elements.

Page AM-16



KEY PROCESSES

- **Problem Management**
  - What are the root causes of problems?
  - How long are problems taking to resolve? What contributes to this time length?
- **Change Management**
  - Are impacts of changes to and applications "end-to-end" infrastructure known?
- **Operations Management**
  - How automated are operational tasks (startup, shutdown, monitoring, recovery)?
  - Are procedures documented, up to date, and accessible?
- **Performance Management**
  - What performance measures relate to service levels?
  - Is performance impact to availability monitored?

©IBM 1999 Availability Management AM-16

Certain systems management processes have an important relationship to availability, and are highlighted here, with examples:

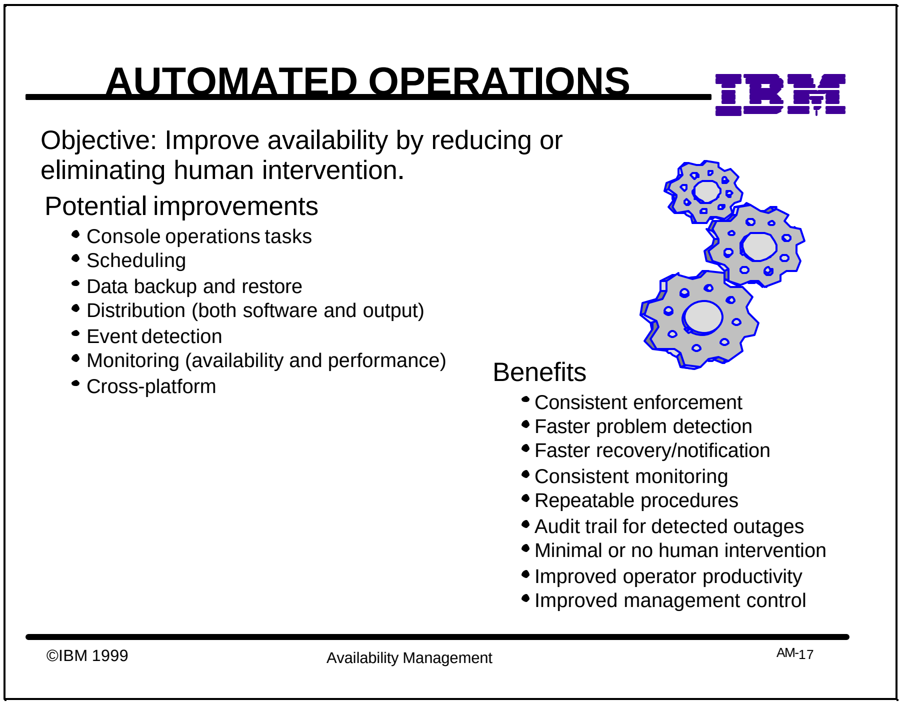
w Problem Management: Beyond reacting to and resolving problems, this process must include determining the root cause of a problem (known as root cause analysis). It must also to break down the problem resolution time to identify any steps that can be improved to reduced the time length.

w Change Management: The wider the “span” of an application across technologies, the more important it is to understand the potential impact of a change in one technology to a change in another. Traditionally different organizations have been responsible for changes to a particular technology. Today these changes must be coordinated across all the environments an application requires.

w Operations Management: Automating of routine tasks and initial detection and recovery of resources removes human error from impacting availability, or extending and outage. Where human interaction is necessary, the supporting documentation to address exception situations must be maintained and kept current.

w Performance Management: Performance indicators can sometimes foreshadow outages. Or, they can identify performance problems that appear to users as outages. It is important to establish a relationship between this information and the targeted service levels.

PAGE AM-17



AUTOMATED OPERATIONS

IBM

Objective: Improve availability by reducing or eliminating human intervention.

Potential improvements

- Console operations tasks
- Scheduling
- Data backup and restore
- Distribution (both software and output)
- Event detection
- Monitoring (availability and performance)
- Cross-platform

Benefits

- Consistent enforcement
- Faster problem detection
- Faster recovery/notification
- Consistent monitoring
- Repeatable procedures
- Audit trail for detected outages
- Minimal or no human intervention
- Improved operator productivity
- Improved management control

©IBM 1999 Availability Management AM-17

Closely related to making systems management processes efficient, and improving availability, is the use of automation. The objective is to reduce or eliminate human intervention but having automation handle routine situations, detect anomalies, and take the initial recovery steps. Automation can detect and react faster than humans, and can supply pre-programmed responses, to avoid specific situation errors.

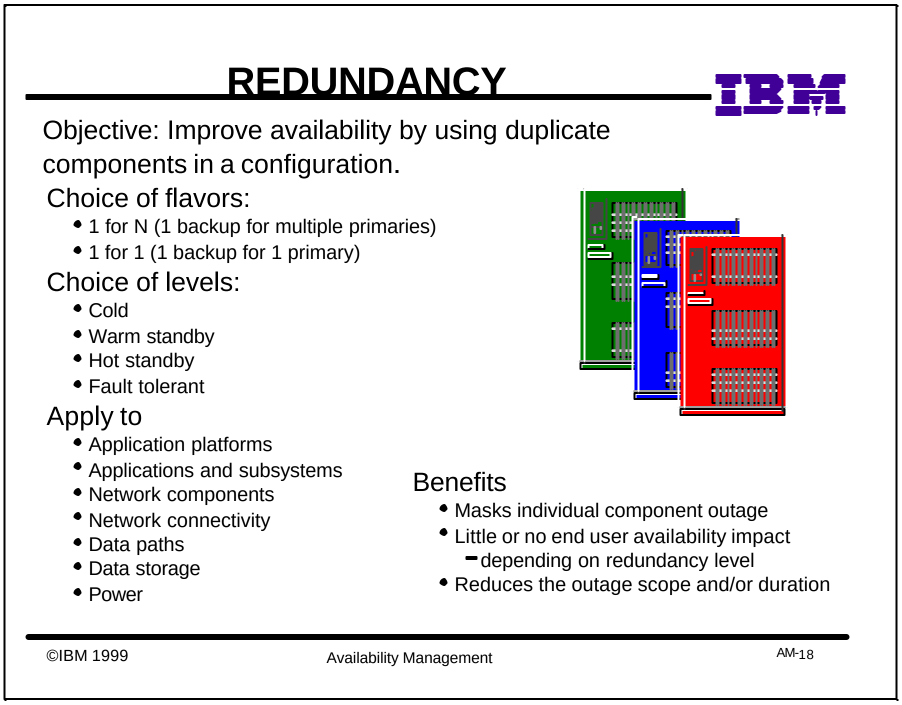
There are many areas where automation can be applied. A few include:

- Console operations tasks, to handle the routine console message monitoring, responses, and initial recovery actions normally done by operators.
- Scheduling of workloads, including adjustments for workload and resource dependencies.
- Data backup and restoration.
- Distribution of software changes and upgrades, and of workload output.
- Monitoring and detecting events that indicate an availability problem or potential availability problem
- Monitoring the performance and availability of critical resources
- Integrating management activities that go across multiple platforms. For example, coordinating workloads in a distributed environment.

Implementing automation can provide many benefits:

- More consistent, repeatable enforcement of defined policies, and monitoring of critical elements
- Minimal human intervention, which will reduce or eliminate human error or response speed from impacting availability.
- Faster detection of exceptions and faster responses for recovery and notification
- Providing an audit trail of an outage. This information can be helpful when analyzing an outage to determine its root cause or potential improvements to address the outage.
- Improved operator productivity, by freeing operators from routine tasks and allowing them to take more an analyst role.
- Improved management control, by using automation to carry out the desired operational and process policies.

PAGE AM-18



REDUNDANCY

IBM

Objective: Improve availability by using duplicate components in a configuration.

Choice of flavors:
- 1 for N (1 backup for multiple primaries)
- 1 for 1 (1 backup for 1 primary)

Choice of levels:
- Cold
- Warm standby
- Hot standby
- Fault tolerant

Apply to
- Application platforms
- Applications and subsystems
- Network components
- Network connectivity
- Data paths
- Data storage
- Power

Benefits
- Masks individual component outage
- Little or no end user availability impact
  - depending on redundancy level
- Reduces the outage scope and/or duration

©IBM 1999 Availability Management AM-18

Redundancy techniques use duplicate components in a configuration to improve availability. Implementing these duplicates requires close attention to the configuration and operation of the environment in which they will be used. Because additional components are being added, complexity increases; more monitoring and better defined exception and recovery actions will be needed.

Redundant components can be implemented as 1/some for N (one or some number component backs up a larger number of other components) or 1 for 1 (one component backs up one other component). The former requires fewer components, but a more complex design to ensure that each of the N components can be backed up. The latter requires more components, but can be more straightforward to design backup procedures around.

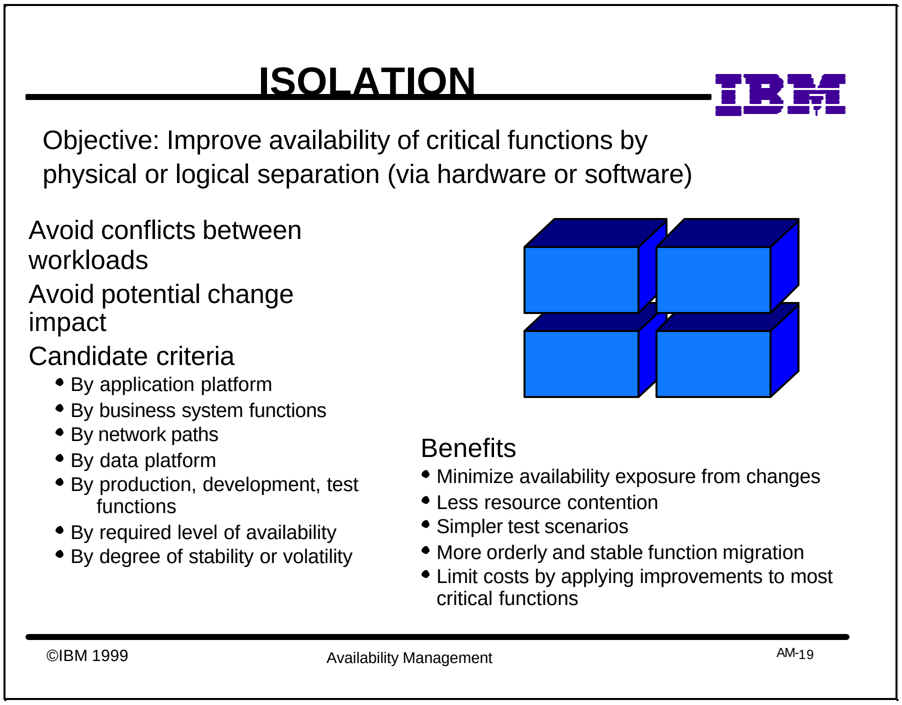
The level of redundancy can also vary. Each has different configuration, operational, and recovery considerations related to availability.:

- COLD: the backup component is normally not operational, and only becomes active when replacing a primary component. This is the easiest level to implement, but can have a higher impact on availability. The time it takes to start, activate, and configure the backup component to replace the primary component can still result in outage time. But it is better than not having any backup at all.

- WARM STANDBY: the backup component is active but not operating as part of the environment. It might be “off-line” or not connected to the network. The time for activation to take over for the primary will be less than that of a cold backup; most of this will be actions to connect it into the environment and to direct workloads to start using it. An outage of a primary component would be almost totally masked by a warm standby.
- HOT STANDBY: the backup component is active and integrated into the environment. It is processing work, or providing workload resources. Less takeover time is needed as compared to a warm standby because it is already part of the configuration; usually all that has to be done is to move work to it, and this can sometimes be done using automation. An outage of a primary component could be masked by having a hot standby as a backup.
- FAULT TOLERANT: Although many equate fault tolerant with availability, it really a redundancy technique. In fault tolerant the backup component is working side by side with the primary component to support the workload or to provide workload resources. Loss of the primary results in the backup continuing to work, and the outage would be completely masked.

Redundancy can be applied to any resource in the environment. Be aware that certain types of resources may more easily support particular backup implementation numbers or levels. For example, fault tolerance is more commonly found as part of hardware components than software components. In addition, additional hardware and software configuration and customization will usually be required to take advantage of redundancy. However, the benefits it can provide - little or no end user availability impact, reduction in an outage scope or duration - is a worthwhile tradeoff.

PAGE AM-19



ISOLATION

Objective: Improve availability of critical functions by physical or logical separation (via hardware or software)

Avoid conflicts between workloads

Avoid potential change impact

Candidate criteria

- By application platform
- By business system functions
- By network paths
- By data platform
- By production, development, test functions
- By required level of availability
- By degree of stability or volatility

Benefits

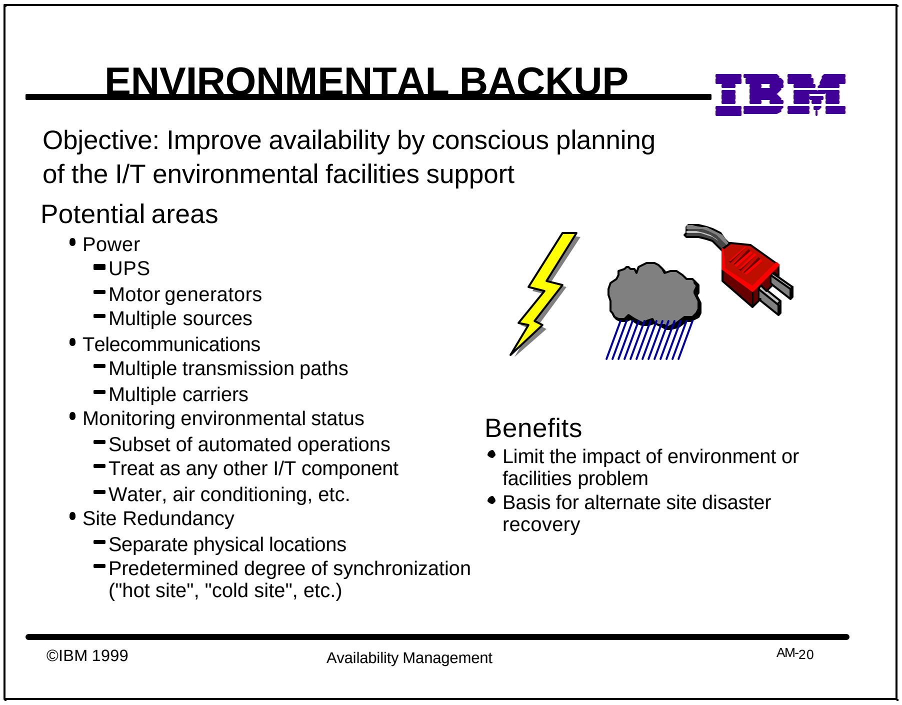
- Minimize availability exposure from changes
- Less resource contention
- Simpler test scenarios
- More orderly and stable function migration
- Limit costs by applying improvements to most critical functions

©IBM 1999 Availability Management AM-19

Isolation is the technique of “protecting your loved ones”. Not all workloads in an environment or equal. Isolation allows separation of those critical workloads to reduce the potential for conflicts from other workloads. This can be implemented physically, using separate application platforms or network infrastructure. It can also be done logically, using performance tuning, workload and bandwidth priorities (as available), etc. To ensure that the critical workloads always have priority access to resources.

Isolation is most commonly done between production and test environments. However, it can also be implemented based on business systems, infrastructure configuration, or application platforms. Total isolation may not be possible using these boundaries for isolation. However, even partial isolation can reduce the impact of outages; it will depend, of course, on the types of outages and environment is experiencing. The benefits of keeping less important work, or changes, from impacting critical work can make isolation a worthwhile technique.

PAGE AM-20



ENVIRONMENTAL BACKUP

Objective: Improve availability by conscious planning of the I/T environmental facilities support

Potential areas

- Power
  - UPS
  - Motor generators
  - Multiple sources
- Telecommunications
  - Multiple transmission paths
  - Multiple carriers
- Monitoring environmental status
  - Subset of automated operations
  - Treat as any other I/T component
  - Water, air conditioning, etc.
- Site Redundancy
  - Separate physical locations
  - Predetermined degree of synchronization ("hot site", "cold site", etc.)

Benefits

- Limit the impact of environment or facilities problem
- Basis for alternate site disaster recovery

©IBM 1999 Availability Management AM-20

Focusing on availability must always include the environment infrastructure that supports the I/T infrastructure. It can have just as much, or even a greater, impact on availability as the I/T components. Planning for the availability of this infrastructure must be done to support all other availability improvements.

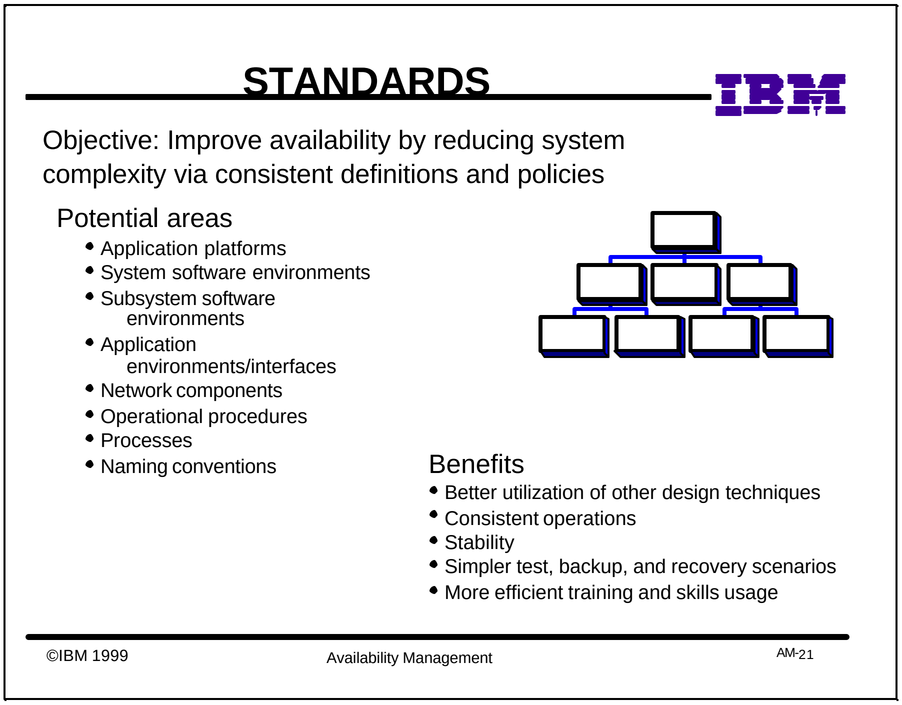
Potential areas for availability planning include:

- Power. Having the proper number and sizes of uninterrruptable power supplies UPS), generators, etc. Also ensuring that there is redundancy so that there are multiple ways to get power.
- Telecommunications. Ensuring that there are multiple physical transmission paths into a site. This will require working with one or more carriers, depending on the types of transmission paths that are selected.
- Environmental status monitoring. There are products that can monitor environmental conditions (power, water, air) and allow the incorporation of the monitoring as part of the overall I/T monitoring. These products can be software running on standard application platforms, or complete standalone systems. In either case they can be connected to the I/T infrastructure to report environmental status and exceptions to I/T systems management products. Some support automated actions.
- Site redundancy. This is simply applying the redundancy technique covered earlier to the entire site. A separate physical location is needed. Most businesses already have, or are planning to have, an alternate site for disaster recovery. The level of site redundancy depends on the required amount of synchronization

between applications and data at the primary and backup sites. The more synchronization required, the higher the cost to implement site redundancy; however, closer synchronization means a site wide outage will have a smaller impact on availability.

Including environmental backup and monitoring as part of an availability design address a major impact to availability and can lead to higher overall availability.

PAGE AM-21



STANDARDS

Objective: Improve availability by reducing system complexity via consistent definitions and policies

Potential areas

- Application platforms
- System software environments
- Subsystem software environments
- Application environments/interfaces
- Network components
- Operational procedures
- Processes
- Naming conventions

Benefits

- Better utilization of other design techniques
- Consistent operations
- Stability
- Simpler test, backup, and recovery scenarios
- More efficient training and skills usage

©IBM 1999 Availability Management AM-21

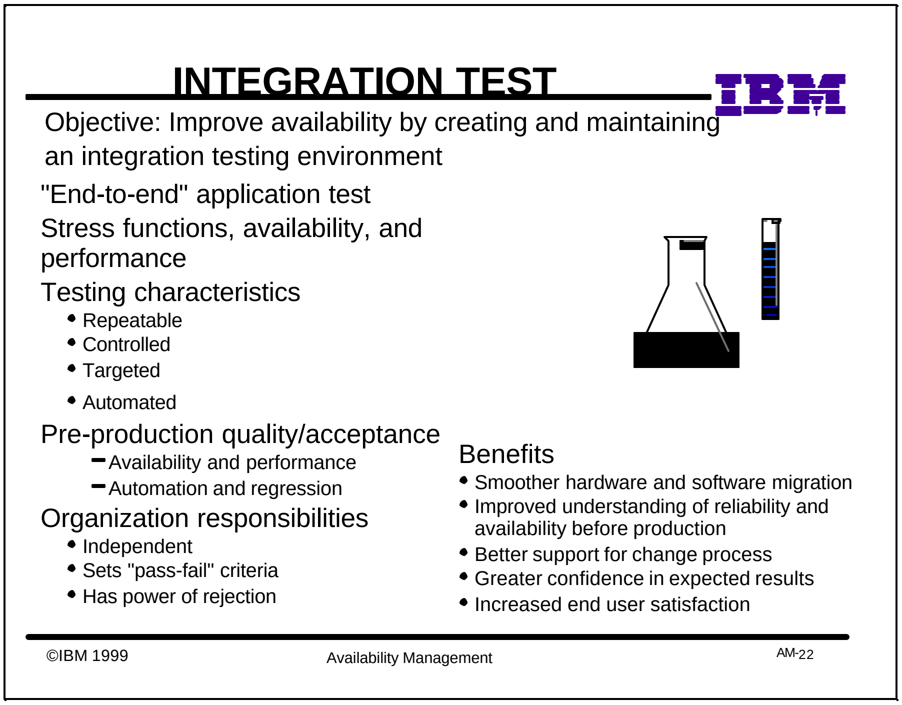
At first glance, standards do not appear to have much of an impact on availability. However, careful deployment of standards can reduce complexity to some degree, and allow systems management processes - including operational, recovery, and support processes - to be streamlined through efficiency.

For example, using standard definitions to customize a set of servers supporting particular types of applications (such as web content servers) can make for easier troubleshooting when a problem does occur. It also makes it easier to design and implement monitoring policies and automation, since, with a standard configuration, the policies or automation can be applied in the same fashion to each server.

Something as simple as standard naming conventions for components can convey a lot of information about a component without having to delve deeply into configuration information.

Standards are part of systems management and can be applied to any component or resource that is supported using systems management processes. It can indirectly lead to availability improvements by allowing other design techniques to be better utilized. Reducing complexity through standardization also aids training and skills usage, which can also have a impact on availability.

PAGE AM-22



INTEGRATION TEST

Objective: Improve availability by creating and maintaining an integration testing environment

"End-to-end" application test

Stress functions, availability, and performance

Testing characteristics

- Repeatable
- Controlled
- Targeted
- Automated

Pre-production quality/acceptance

- Availability and performance
- Automation and regression

Organization responsibilities

- Independent
- Sets "pass-fail" criteria
- Has power of rejection

Benefits

- Smoother hardware and software migration
- Improved understanding of reliability and availability before production
- Better support for change process
- Greater confidence in expected results
- Increased end user satisfaction

©IBM 1999 Availability Management AM-22

Test environments are nothing new; every business has some test platform or system where components are tested to ensure that they function properly. The traditional test environment has concentrated on function testing; that is, testing components to determine if their functions match the design requirements.

Availability improvements requires a higher level of testing. It goes beyond “do functions work?” into “do functions work together, across the end-to-end application, using the same components that are in the production environment?”. This testing almost purposely tries to break an application, and then tests the recovery actions to restore the application to its normal state. This includes:

- Stress testing, to send high volumes of workload activity to determine where the “breaking” or degradation” point is.
- Availability testing, to stop one or more of the application components and determine the impact to the users.
- Performance testing, to determine how changes to the components affect the performance of the application (or the indirect impact to other applications).

Within an integrated test environment, these testing characteristics must be repeatable, controlled, and targeted, so that each result can clearly be matched to a cause. Automation of this testing can provide this.

Testing this fashion will provide a very accurate picture of potential problems and potential resolutions before going into production.

Before moving an application or a set of components into production, there is usually a quality/acceptance criteria that has to be met. From the availability perspective, there should be availability. Performance, and automation acceptance criteria that are tested and validated in the integration test environment. For example, automation of application startup, shutdown, and recovery should be tested and validated for acceptance. Doing it at this stage provides more time to ensure it works properly before migrating to production.

From an organization standpoint, ad integrated testing environment is only as good as the degree of clout it has. Some ideas to consider:

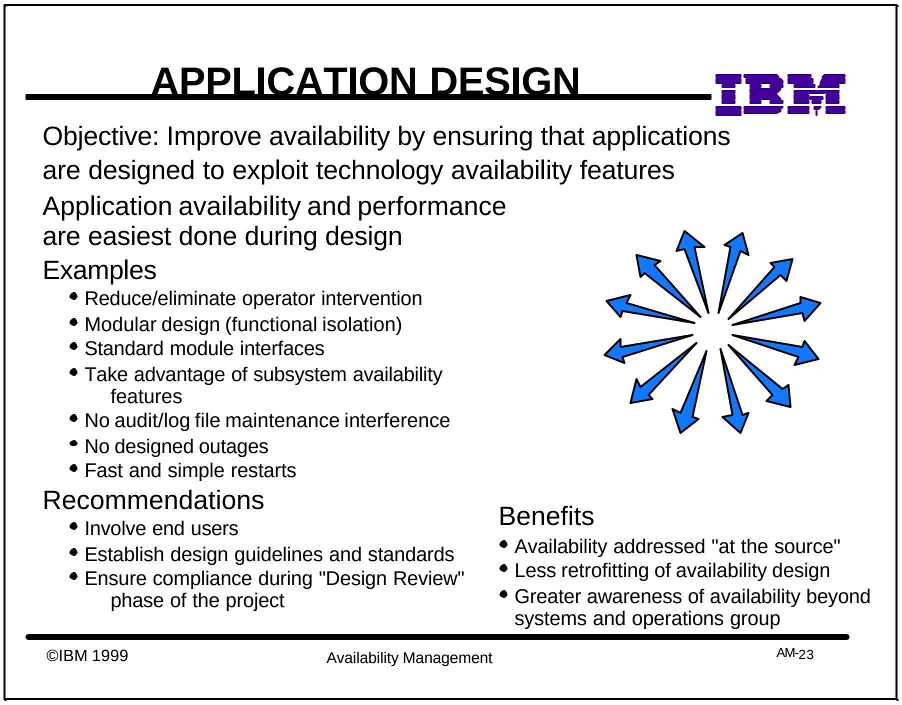
w It is under control of an independent, or cross-organization, entity?

w Can it set “pass-fail” acceptance criteria that everyone must adhere to?

w Does it have to power to “reject” (block from production) applications that do not meet the acceptance criteria?

These type of conditions are essential for an integrated test environment to provide quality availability improvements in a proactive manner. The overall benefit is to discover availability exposures and to address them now, instead of discovering them in production.

PAGE AM-23



APPLICATION DESIGN

Objective: Improve availability by ensuring that applications are designed to exploit technology availability features

Application availability and performance are easiest done during design

Examples
- Reduce/eliminate operator intervention
- Modular design (functional isolation)
- Standard module interfaces
- Take advantage of subsystem availability features
- No audit/log file maintenance interference
- No designed outages
- Fast and simple restarts

Recommendations
- Involve end users
- Establish design guidelines and standards
- Ensure compliance during "Design Review" phase of the project

Benefits
- Availability addressed "at the source"
- Less retrofitting of availability design
- Greater awareness of availability beyond systems and operations group

©IBM 1999 Availability Management AM-23

The final technique, for application design, goes back to the source: are application being designed from the start to exploit available technology features in hardware and software?

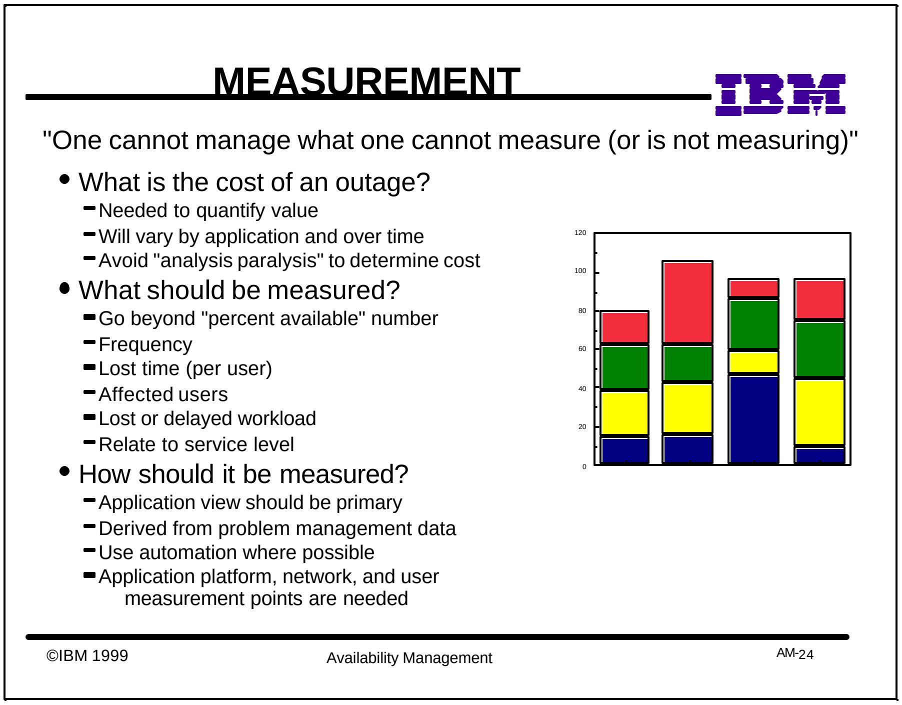
While most of the other techniques concentrate on the “external” view of the application, this technique focuses on the “internal” view. The best time to address availability and performance issues are during the design phase of an application. It is still not uncommon to see applications go into production, and at that point attempt to apply availability techniques. It is more efficient to apply them as part of the design.

For example, suppose an application, at startup time, required some information to be enter manually before it would continue. Failure to do this, or to do it correctly, could result in an outage due to a start up delay or crash. Automation could be designed to input that information, so that it would be consistent and would prevent an input error from occurring. But the best approach would have been to design the application so that manual input was not needed in the first place.

The presentation page gives just a few examples of design actions that should be considered. In general, the application should not require manual input, should anticipate file or log problems, and should not have to be shutdown for routine maintenance. The best way to determine design actions are to involve the end users, who have a direct vested interest in application availability, and to establish design guidelines and standards that include availability considerations. These considerations should be evaluated during testing and design reviews for compliance.

This technique is not of much use to applications already in production, or from third parties. But in today's environment, businesses are bringing new applications forward at a faster rate to reach new markets and to use as a competitive edge. In those cases design time should not be sacrificed for expediency, because of the exposure unavailability will bring, and the benefits of a highly available application.

PAGE AM-24



MEASUREMENT

"One cannot manage what one cannot measure (or is not measuring)"

- What is the cost of an outage?
  - Needed to quantify value
  - Will vary by application and over time
  - Avoid "analysis paralysis" to determine cost
- What should be measured?
  - Go beyond "percent available" number
  - Frequency
  - Lost time (per user)
  - Affected users
  - Lost or delayed workload
  - Relate to service level
- How should it be measured?
  - Application view should be primary
  - Derived from problem management data
  - Use automation where possible
  - Application platform, network, and user measurement points are needed

©IBM 1999 Availability Management AM-24

A discussion of availability is not complete without covering availability measurement. There is an old systems management adage: “one cannot manage who one cannot measure (or is not measuring)”.

Improving availability means little if it is not being measured. Measurement provides a foundation to establish where things stand today, and the amount of improvement that is required. It will also reveal which of the implemented availability techniques are making a difference.

To quantify these measurements, the cost of an outage must be determined. This will be used to determine the return an expenditure for improving availability will provide. The cost of an outage will vary by application, and will vary over time. However, do not spend a lot of time coming up with the “perfect” number; it is possible to fall into “analysis paralysis” and never determine an outage cost. This is an art, not a science; a ballpark estimate is a very good starting point. The survey on page **AM-06** can provide input to help come up with an estimate. There are also many web sites with “outage cost calculators” that, once provided with some basic input information, can return an outage cost estimate.

Traditional availability measurements tend to be hardware specific, and only a percentage. To be useful, the measurements must provide more information. The useful information includes:

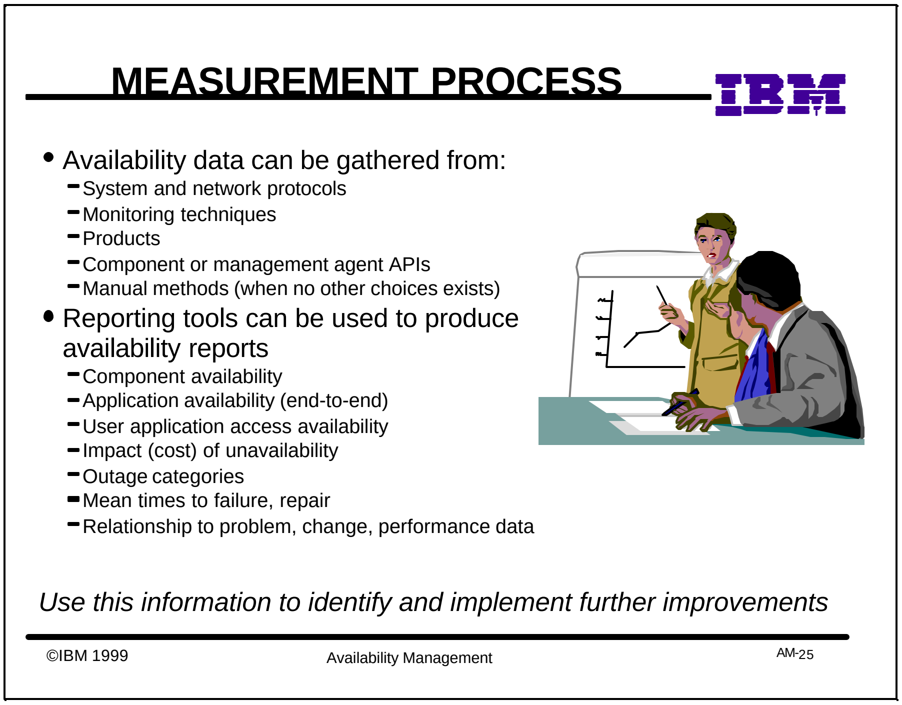
- Frequency - how many times did outages occur

- Amount of lost time (per application or, if possible, per user)
- Number of users affected by an outage
- Volume of workload (transactions, business units, etc.) Lost or delayed

Of course, any measurements should also support the documented service level measurement requirements.

It can be an overwhelming task to take manual availability measurements. These measurements will not accurately reflect what the users are experiencing. The following recommendations will help provide accurate measurements:

- Take an “application view” for the measurement. In other words, the hardware or operating system is not the primary availability measure, but the application is. The hardware and operating system measure is important only in light of the impact to the application measurement.
- A good problem management process already has a wealth of data that can be used. This can provide quantifiable data on root causes, outage categories, and outage scope, to name a few.
- Automation should be used to collect the “raw” data. It can also be used to transform the data and place it into any desired repository for reporting purposes.
- Multiple measurement points are needed. Measuring a single component will not provide an accurate view of an application, especially one that spans multiple platforms and network components. Measurement points - where data is collected from - will be needed at the user, application platform, and in the network.



MEASUREMENT PROCESS

- Availability data can be gathered from:
  - System and network protocols
  - Monitoring techniques
  - Products
  - Component or management agent APIs
  - Manual methods (when no other choices exists)
- Reporting tools can be used to produce availability reports
  - Component availability
  - Application availability (end-to-end)
  - User application access availability
  - Impact (cost) of unavailability
  - Outage categories
  - Mean times to failure, repair
  - Relationship to problem, change, performance data

Use this information to identify and implement further improvements

©IBM 1999 Availability Management AM-25

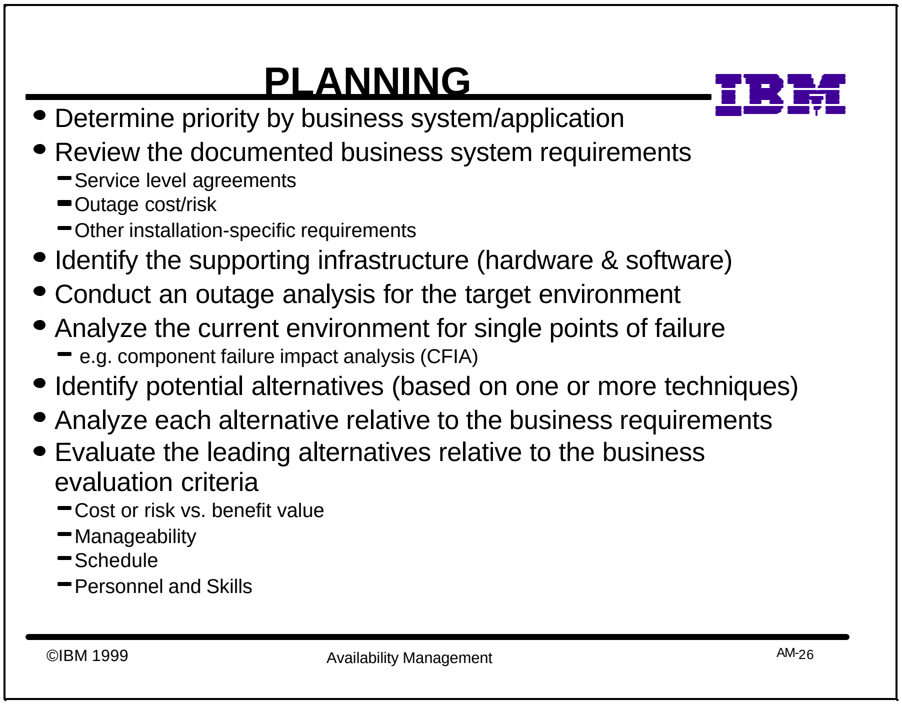
Measurement requires the collecting and analysis of data, and reporting it as useful information. Points to consider:

1. There is already a lot of data being produced by system and network protocols, monitoring techniques, products, and APIs that can be used to feed availability measurements. This is less “does the data exist” and more “of all the data that does exist, which should I use?”. Particular sources will vary based on the type of component being measured, the degree of automation used to obtain the data, and if the data is captured in real time vs. Being extracted from log or file.
2. Reporting tools from simple spreadsheets to full blown enterprise reporting products can be used to create availability reports. Some products even provide canned availability report templates. Varies types of reports should be used. They should provide data on application, component, and user availability (these are not all equal); Outage root cause categories and cost of specific outages; component mean times to failure and repair; and the relationship to problems, changes, and performance situation or data that impacted availability.

It is important to measure, but it is even more important to use these measurements. While there are many measurements that can be taken, the best ones to take will be the used to identify and implement additional availability improvements.

The white paper, “Finding an Collecting Availability Measurement Data”, goes into more detail on measurement process recommendations and is available from the author.

PAGE AM-26



PLANNING

- Determine priority by business system/application
- Review the documented business system requirements
  - Service level agreements
  - Outage cost/risk
  - Other installation-specific requirements
- Identify the supporting infrastructure (hardware & software)
- Conduct an outage analysis for the target environment
- Analyze the current environment for single points of failure
  - e.g. component failure impact analysis (CFIA)
- Identify potential alternatives (based on one or more techniques)
- Analyze each alternative relative to the business requirements
- Evaluate the leading alternatives relative to the business evaluation criteria
  - Cost or risk vs. benefit value
  - Manageability
  - Schedule
  - Personnel and Skills

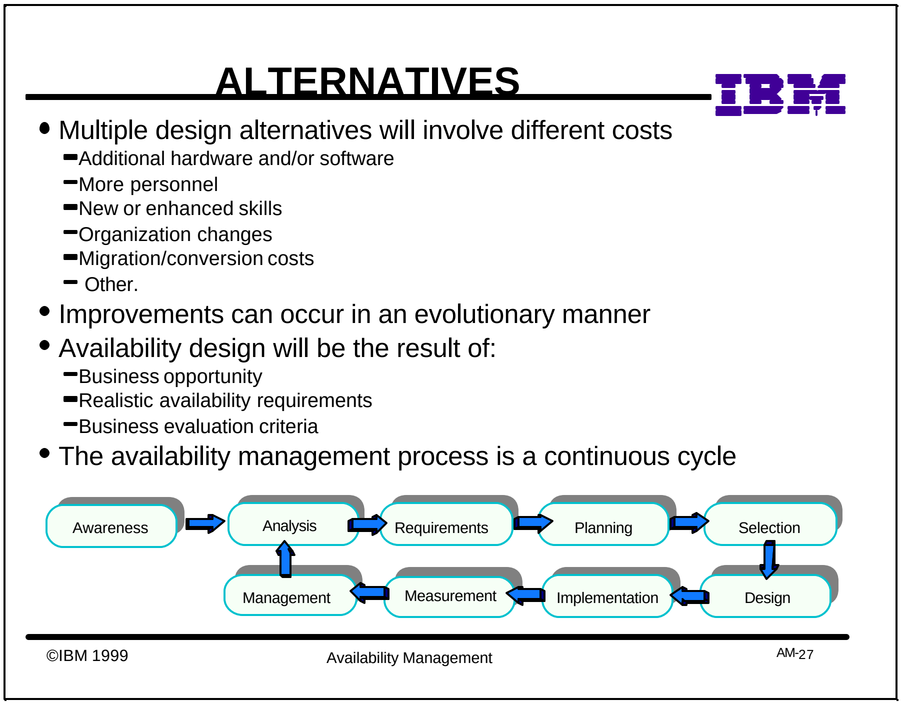
©IBM 1999 Availability Management AM-26

Now that the major techniques for availability improvement and measurement have been reviewed, a framework for planning within availability management will be useful. This is an overview of the major planning activities to be conducted. More formal availability planning methodologies will include these steps:

- It is easier to plan for a specific business system (group of applications) or a specific application. The ones that are deemed most important to the business are the ones that will have priority. These will benefit the most from improved availability.
- The current information and requirements for the business system or application must be understood. At a minimum, service level agreements, outage costs, and other requirements must be documented. If these do not exist, it is recommended that they be put in place before detailed planning continues. Without this information it is difficult to determine the availability objectives.
- The current I/T environment - both hardware and software - that supports the business system or application must be identified, at a detailed level. This means down to the specific components needed for the business system to operate.
- Conduct an outage analysis for this environment . This is a detailed review of the outages that have been experienced to determine how much outage minutes have occurred, what outage categories have the highest amount of incidents and minutes, the root cause of outages, and identification of the things that contribute to the length of an outage.

- Conduct a component failure impact analysis (CFIA) for the components that support the business system or application. CFIA analysis will identify single points of failure and component relationships that can impact their availability and recovery.
- Use the availability improvement techniques, along with the requirements, outage analysis, and CFIA information, to develop potential alternatives to address unavailability or for improving availability. Multiple alternatives will always be found; they must be evaluated against business criteria such as cost vs. benefit, manageability (does the alternative fit within the systems management structure of the environment), schedule (amount of time it will take to implement), personnel (how many are required to implement and/or support), and skills (what skills are need for this alternative).

PAGE AM-27



ALTERNATIVES

- Multiple design alternatives will involve different costs
  - Additional hardware and/or software
  - More personnel
  - New or enhanced skills
  - Organization changes
  - Migration/conversion costs
  - Other.
- Improvements can occur in an evolutionary manner
- Availability design will be the result of:
  - Business opportunity
  - Realistic availability requirements
  - Business evaluation criteria
- The availability management process is a continuous cycle

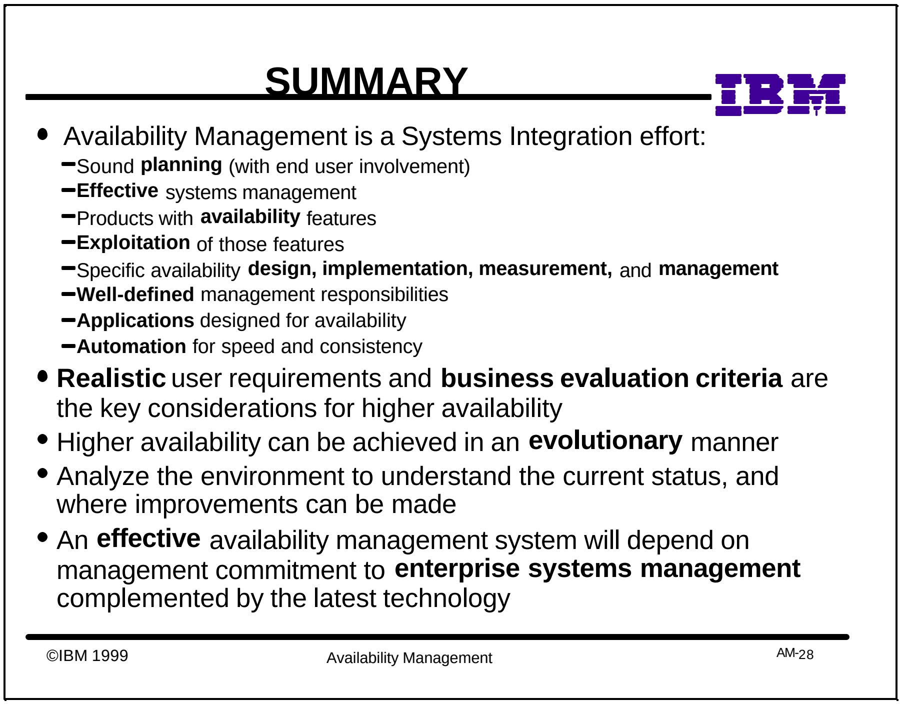
©IBM 1999 Availability Management AM-27

Most of the time there will not be a “perfect” choices among the alternatives. Each will involve different costs in different areas. Some will require additional hardware, software, personnel, or skills needed. Others will involve organization changes - roles, responsibilities, even management philosophies. There can be migration or conversion costs for the selected alternative to work. There may not be a “perfect answer”; but always taking the view of “what is the investment compared to the return” becomes the bottom line. A $50,000 investment to prevent $1,000,000 worth of outages occurring is worthwhile; a $50,000 investment to prevent $10,000 worth of outages may warrant reconsideration.

With whatever alternative solution is selected, always remember that the job is not finished once the solution is implemented. The I/T environment is dynamic, and a selected solution will have to be assessed in light of this. The implemented solutions, and the resulting improvements, can both evolve as the I/T environment changes. The solution must contain a degree of flexibility.

Finally, availability design, as stated earlier, is not purely a technical exercise. It is applying process and technology based on the business opportunity, availability requirements, and business evaluation criteria. Overall, availability management is part of a continuous cycle. After a design is implemented, the results must be measured and fed back to determine if any fine tuning of the design is needed, or even to address the next availability requirement.

PAGE AM-28



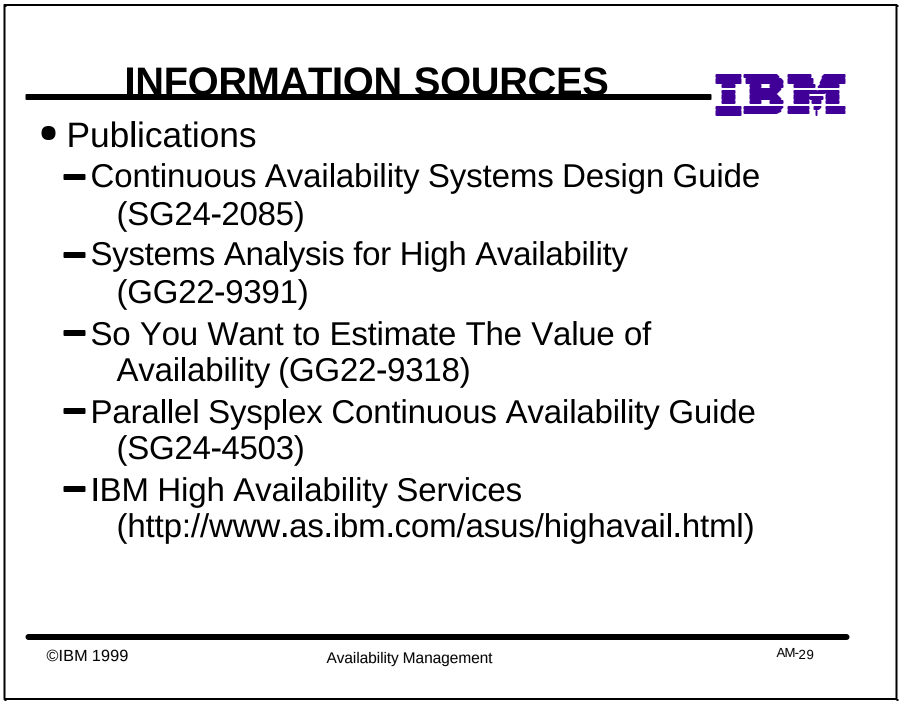
SUMMARY

- Availability Management is a Systems Integration effort:
  - Sound **planning** (with end user involvement)
  - **Effective** systems management
  - Products with **availability** features
  - **Exploitation** of those features
  - Specific availability **design, implementation, measurement,** and **management**
  - **Well-defined** management responsibilities
  - **Applications** designed for availability
  - **Automation** for speed and consistency
- **Realistic** user requirements and **business evaluation criteria** are the key considerations for higher availability
- Higher availability can be achieved in an **evolutionary** manner
- Analyze the environment to understand the current status, and where improvements can be made
- An **effective** availability management system will depend on management commitment to **enterprise systems management** complemented by the latest technology

©IBM 1999 Availability Management AM-28

In Summary, Availability Management is a true systems integration effort. It spans both systems management processes and technology. It requires participation from multiple organizations, both user and I/T. Planning design, implementation, and measurement are all required. Improved availability does not just "happen" but is the result of proper availability management. An understanding of the current environment is necessary to determine the actions needed, in Availability Management, for improvement.

PAGE AM-29



INFORMATION SOURCES

- Publications
  - Continuous Availability Systems Design Guide (SG24-2085)
  - Systems Analysis for High Availability (GG22-9391)
  - So You Want to Estimate The Value of Availability (GG22-9318)
  - Parallel Sysplex Continuous Availability Guide (SG24-4503)
  - IBM High Availability Services (http://www.as.ibm.com/asus/highavail.html)

©IBM 1999 Availability Management AM-29

IBM has several publications that will assist in understanding and implementing the steps required for availability management:

w Continuous Availability Systems Design Guide (SG24-2085)
w Systems Analysis For High Availability (GG22-9391)
w So You Want to Estimate the Value of Availability (GG22-9318)
w Parallel Sysplex Continuous Availability Guide (SG24-4503), for OS/390 Parallel Sysplex Environments

IBM Education and Training offers courses on Availability Management to help in the understanding of the overall process and the planning considerations that are necessary. In addition, IBM Global Services provides IBM High Availability Services offerings to address customer availability requirements within various technology environments. IBM Global Services also conducts customer engagements assess the Availability Management process that is being used or considered, and to identify what steps should be implemented and/or where improvements can be made.

END OF DOCUMENT