Making Smart Investments to Reduce Unplanned Downtime

Unplanned application downtime causes havoc and great expense. Conventional vendor wisdom focuses on redundancy to improve availability. Redundancy, however, solves just 20 percent of the problem.

Based on extensive feedback from clients, we estimate that, on average, unplanned application downtime is caused (see Figure 1): 20 percent of the time by hardware (e.g., server and network), OSs, environmental factors (e.g., heating, cooling and power failures) and disasters; 40 percent of the time by application failures including “bugs,” performance issues or changes to applications that cause problems (including the application code itself or layered software on which the application is dependent); and 40 percent of the time by operator errors, including not performing a required operations task or performing a task incorrectly (e.g., changes made to infrastructure components that result in problems and incur unexpected downtime).

Thus, approximately 80 percent of unplanned downtime is caused by people and process issues, while the remainder is caused by technology failures and disasters. Improving availability requires a different strategy and set of investment choices for each of the three unplanned downtime categories.

Technology Failures and Disasters: Despite being just 20 percent of unplanned downtime, these types of failures can be very catastrophic and result in a significant amount of downtime per incident (see Note 1). To mitigate this risk, enterprises should take the following steps. Monitor components for availability (since failure identification is the first step toward resolution). This is typically done with agents or sensors. Ideally, monitoring is predictive and warns the operator or vendor of potential failures prior to their occurrence. Buy vendor service contracts to reduce time to repair. Many vendors offer time-to-repair commitments for increased fees. Implement redundancy to ensure alternate processing capabilities in the event of a catastrophic failure. Data mirroring, clustering and diesel generators are examples of redundancies that limit downtime when failures occur. In comparing potential solutions, pay particular consideration to

GartnerGroup

Entire contents ©1999 by Gartner Group, Inc. All rights reserved. Reproduction of this publication in any form without prior written permission is forbidden. The information contained herein has been obtained from sources believed to be reliable. GartnerGroup disclaims all warranties as to the accuracy, completeness or adequacy of such information. GartnerGroup shall have no liability for errors, omissions or inadequacies in the information contained herein or for interpretations thereof. The reader assumes sole responsibility for the selection of these materials to achieve its intended results. The opinions expressed herein are subject to change without notice.
how transparent recovery is to end users and applications.

**Application Failures:** To reduce downtime caused by application failures, enterprises should invest in improving and re-engineering IT processes (see Note 2), including the following:

1. **change management** — reduces unplanned downtime caused by inadequate planning and testing of application changes, enables a more proactive approach toward problem prevention (see Note 3);
2. **problem management** — improves problem identification, isolation and resolution, thereby reducing time to repair;
3. **configuration management** — tracks the relationships between dependent application and infrastructure components, enables better understanding of change impact and quicker fault diagnosis;
4. **application architecture and design** — reduces single points of failure, aides in problem isolation and makes application failures more transparent to users; and
5. **performance management and capacity planning** — proactively identifies current and future resource shortages impacting SLAs.

Re-engineering processes requires investing in people responsible for the process and tools to facilitate the process. By comparison, reducing technology failures through redundancies is a much simpler problem to solve than process re-engineering, since redundancy-related failures do not require significant changes in human behavior. Through 2003, less than one third of enterprises will successfully implement change management for the heterogeneous environment (0.8 probability).

**Operator Error:** Reducing downtime caused by operator error requires: 1) maturing IT operations to a more-process-oriented and documented approach that does not require or mandate specific, knowledgeable people be available to perform tasks, 2) hiring competent people and training operators (and vendors) on IT process and procedures, 3) automating the process wherever possible to reduce the chance for errors (e.g., using job scheduling and event management tools), and 4) improving the change and problem management processes related to IT infrastructure and facilities.

**Bottom Line:** Enterprises should not let infrastructure redundancy provide a false sense of availability assurance. To address the 80 percent of unplanned downtime caused by people and process failures (vs. technology failures or disasters), enterprises should invest in improving IT processes, such as change, configuration and problem management; performance and capacity planning; application architecture and design; and operator hiring and training. Investments should also be made in automation. Other downtime causes should be addressed by eliminating single points of failure through redundancy, with vendor service contracts and component monitoring.