

Daraus folgt, dass übergreifende Prozesse, wie beispielsweise Konfigurations- oder Versionsmanagement, je ODS-Instanz oder je Kunde ausgeführt werden können.

Diese Art der Integration wirkt sich auch auf die in ITIL definierten Service-Delivery-Prozesse wie beispielsweise das IT-Finanzmanagement aus. Auch diese Prozesse müssen von dem statischen, klassischen „Apportionment“-Ansatz, der lange Planungszyklen bedingt, in einen Ansatz geändert werden, der die tatsächliche Konfiguration und Ressourcennutzung der ODS-Instanzen widerspiegelt. Dadurch kann das IT-Prozessmanagement-System dann letztendlich das von den Kunden geforderte nutzungsabhängige Gebührenmodell unterstützen.

3.3.9 Zusammenfassung

Zusammenfassend lässt sich sagen, dass Utility-Computing-Modelle aufgrund ihrer hohen Dynamik technologisch äußerst anspruchsvoller Managementsysteme bedürfen. Die technologische Entwicklung der letzten zehn Jahre, der Reifegrad der XML – und speziell Web-Services-Technologien und Standards – ermöglicht es heute, solche Managementsysteme auf der Grundlage der service-orientierten Architektur zu entwickeln und sie noch dazu mit einem hohen Grad von autonomem selbststeuerndem Verhalten zu versehen. Durch den konsequenten Einsatz solcher Systeme wird damit ein Wandel des Rechenzentrums vom Cost- zum Profitcenter ermöglicht. Damit wird das Rechenzentrum zur wichtigsten Keimzelle fortlaufender Innovationen des Unternehmens.

3.4 Virtualisierung

Boas Betzler

3.4.1 Die Auswirkungen der Virtualisierung

Es wurden bereits viele Versuche unternommen, den Boom im Bereich der Informationstechnologie mit dem Silicon Valley in Verbindung zu bringen und die Ursprünge dieser Entwicklung bis zu Haight-Ashbury als Zentrum der amerikanischen Hippiekultur in San Francisco zurückzuverfolgen. Die Suche nach dem Aufkommen der Virtualisierung in der Popkultur führt uns nach Yorkville, dem kanadischen Hippiezentrum der späten 60-er Jahre. Hier erläuterte der 19 Jahre alte William Gibson die Bewegung wie folgt: „Anstatt die Gesellschaft zu attackieren ... zum Zusammenbruch zu bringen ... könnte auch eine Subkultur entstehen. Eine Subkultur mit der entsprechenden Größe könnte wiederum die vorhandene Kultur verändern.“ [1]

Der Film „Matrix“ spielte nicht nur an den Kinokassen einen unerwarteten Gewinn von 460 Mio. Dollar weltweit ein [2], sondern verursachte auch einen Moderrummel um schwarze Sonnenbrillen und lange Trenchcoats. Die Geschichte beschreibt eine virtuelle Welt, die parallel zur realen, physischen Welt existiert. Personen aus der realen Welt können mithilfe von cybernetischen Implantaten mit der virtuellen Welt interagieren. Möglicherweise wurde bisher der gedankliche Einfluss von William Gibson zu wenig beachtet. Gibson begann seine Karriere als Autor und prägte den Begriff „Cyberspace“ als Bezeichnung für eine virtuelle Welt oder Realität. In seinem ersten Roman „Neuromancer“ [3] beschreibt er eine Umgebung, in der eine Gemeinschaft in einem Computernetzwerk namens Matrix gedeiht, das über Techniken der virtuellen Realität zugänglich ist.

3.4.2 Die Definition von Virtualisierung

Während diese Beispiele aus der Pop-Art unterhaltsame Aspekte der Virtualisierung mit sich brachten, kann der Begriff „virtuell“ bis zum mittelalterlichen Ausdruck „virtualis“ („Leistung“, „Potenz“, „Effizienz“) zurückverfolgt werden. Die Verbindung zum Computer im Sinne von „nicht physisch existent, sondern durch Software dargestellt“ stammt aus dem Jahre 1959 [4]. Das Konzept der „Virtualität“ lässt sich auch auf die Physik (virtuelle Partikel) und auf die Philosophie übertragen. Wir konzentrieren uns hier jedoch auf den Zusammenhang dieses Begriffs mit Computern und mit der Geschäftswelt. Heute sind viele verschiedene Definitionen für den viel zitierten Begriff „Virtualisierung“ eingeführt. Daher empfiehlt es sich, einige Beispiele anzuführen, um Merkmale zu ermitteln und die verschiedenen Aspekte der Virtualisierung verständlich zu machen. Bestimmte Konzepte kehren immer wieder und kristallisieren sich eindeutig heraus:

- Abstraktion
- Simulation und Emulation
- Sharing
- Partitionierung
- Isolation
- Aggregation

3.4.3 Beispiele für Virtualisierung

Das Konzept der Virtualisierung wird in vielen Bereichen der Informationstechnologie angewendet. Wir werden zwei Beispiele – virtueller Speicher und virtuelle Maschinen – ausführlicher behandeln. Auch in anderen Bereichen, beispielsweise bei Netzwerken, wird die Virtualisierung als Mittel der Abstraktion und

Isolation eingesetzt. Virtual Private Networks (VPN) nutzen ein öffentliches Netzwerk und stellen durch Tunnelprotokolle verschlüsselte Kommunikation bereit, um Authentifizierung und Datensicherheit zu erzielen. Die Einführung einer Verbindungsüberlagerung führt zu einer virtuellen Abstraktion von Netzwerkressourcen wie beispielsweise Router, Switches, physische Verbindungen und Firewalls, sodass die Kommunikation zwischen Endpunkten durch Vereinbarungen zur Servicequalität (Service Level Agreements, kurz SLA) für Antwortzeit, Bandbreite, Sicherheit und Kosten definiert werden kann.

Virtueller Speicher

Am Beispiel des virtuellen Speichers lassen sich grundlegende Konzepte erläutern, die auf andere Nutzungsarten der Virtualisierung übertragen werden können. Dieses Beispiel mag an einen Computergrundkurs erinnern, aber wir werden die grundlegenden Eigenschaften auf neue Technologien übertragen können. Das hier erläuterte System basiert auf dem IBM System/360 Modell 67 [5]. Ein virtuelles Speichersystem kann generell als Informationssystem definiert werden, in dem zwischen den virtuellen (logischen), von einem Programm erzeugten Adressen und den physischen Adressen eines realen Speichersystems (von dem die Informationen eingelesen werden) unterschieden wird.

Der Begriff „virtueller Speicher“ bezieht sich allgemein auf die Adressierung einzelner Speicherworte durch einen zentralen Prozessor und im Besonderen auf Systeme, bei denen Speicheradressen übersetzt oder durch Adressübersetzung dynamisch neu zugewiesen werden. Ein einfaches System dieser Art findet sich in Computern, bei denen den virtuellen (effektiven, logischen) Adressen eine einfache Offsetkonstante hinzugefügt wird.

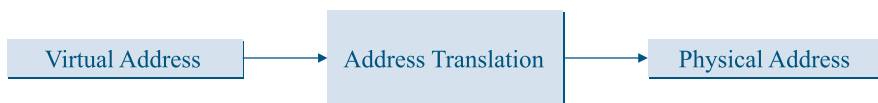


Abb. 3.11.

Aus diesem Modell lassen sich verschiedene grundlegende Merkmale ableiten. Zunächst dient die Einführung der Adressübersetzung als Mittel zur Kapselung der Implementierung der physischen Adressauflösung gegenüber dem Beobachter. Diese Abkopplung ermöglicht außerdem die Standardisierung der Beobachterschnittstelle, während die zugrunde liegende Implementierung modifiziert werden kann, ohne die Anwendung in der Nutzung virtueller Adressen zu beeinträchtigen. Der virtuelle Adressraum kann beispielsweise wesentlich größer als der physische Adressraum sein. Man kann verschiedene Schnittstellen simulieren und alle Arten von nicht funktionellem Verhalten emulieren. Der Nutzen dieser Abstraktion lässt sich mit der folgenden Abbildung veranschaulichen.

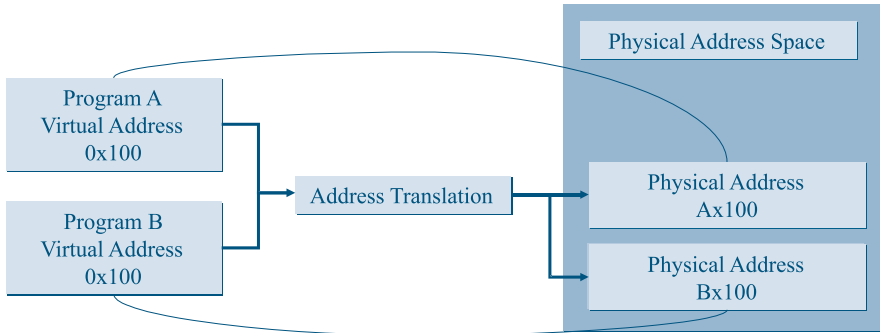


Abb. 3.12.

In diesem Beispiel verwenden zwei Programme den gleichen physischen Speicher, aber das Konzept der Virtualisierung ermöglicht den Programmen A und B die Nutzung der gleichen logischen Adresse. Die Virtualisierungsebene löst die von A und B verwendeten Adressen in verschiedene physische Speicherpositionen auf. Auf diese Weise können A und B ohne Kenntnisse weiterer Teilnehmer im jeweiligen Programm gemeinsam auf den gleichen Speicherraum zugreifen. In diesem Fall partitioniert die Speichervirtualisierung die physischen Ressourcen und projiziert ein normalisiertes Adressierungsschema über die Programmpartitionen. Auf diese Weise werden auch die beiden Programme isoliert, die gemeinsam auf die gleichen Ressourcen zugreifen.

Bis zu diesem Punkt haben wir uns damit befasset, wie Abstraktion für Simulation und gemeinsame Nutzung eingesetzt werden kann. Virtueller Speicher ist jedoch auch ein Beispiel für das Konzept der Aggregation.

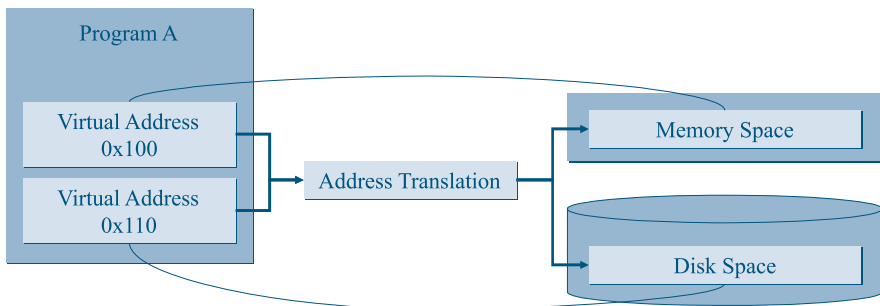


Abb. 3.13.

In diesem Fall verwendet Programm A zwei Adressen, in einem aus seiner Perspektive linearen Adressraum. Die Virtualisierung durch Adressübersetzung ermöglicht die Auflösung einer Adresse zu einem Ort im physischen Speicher und die Auflösung einer anderen Adresse zu einem Ort auf einer Festplatte. Mit

anderen Worten: Aus Sicht eines Beobachters werden die physischen Ressourcen Speicher und Festplatte zu einer einzigen Ressource, dem virtuellen Adressraum, zusammengefasst.

Virtuelle Maschinen

Ähnlich wie ein virtueller Speicher lässt sich auch eine virtuelle Maschine als Datenverarbeitungssystem definieren, bei dem sich die von einem Programm übergebenen Anweisungen von jenen Anweisungen unterscheiden können, die von der Hardware zur Erfüllung der Aufgabe ausgeführt werden. Da Anweisungen in der Regel sowohl Speicheradressen als auch Betriebscodes enthalten, kann ein virtuelles Maschinensystem sowohl virtuellen Speicher als auch andere virtuelle Hardwaremerkmale beinhalten. In dieser Hinsicht ist das Konzept einer virtuellen Maschine eine Verallgemeinerung des Konzepts des virtuellen Speichers. Typischerweise stellt in einem solchen System ein Computer (das Hostsystem) die funktionelle Simulation einer oder mehrerer Computer, also virtueller Maschinen, bereit. Das virtuelle Maschinensystem muss die funktionale Simulation der grundlegenden Komponenten eines realen Systems zur Verfügung stellen.

In dem Ausmaß, in dem die Komponenten einer virtuellen Maschine direkte oder identische Gegenstücke im physischen Hostsystem aufweisen, kann die funktionelle Simulation durch reale Komponenten des Hostcomputers übernommen werden – ansonsten ist eine detaillierte, schrittweise Simulation erforderlich. Die Verwendung von Komponenten im Hostcomputer zur Beeinflussung einer funktionellen Simulation der virtuellen Maschinen hängt vor allem von den gegebenen Architekturen von Host und virtuellem Computer ab, also von der Möglichkeit zur Abstraktion dieser Komponenten. Sind beispielsweise die Anweisungen der virtuellen Maschine und des Hostcomputers identisch, können viele der von der virtuellen Maschine auszuführenden Anweisungen von der Hosthardware direkt umgesetzt werden. Das ist dann der Fall, wenn eine Zwischenstufe (eine Virtualisierungsebene) eingefügt werden kann. Ein Beispiel hierfür ist die Möglichkeit, die virtuelle Maschine daran zu hindern, ihren Status zu verändern (beispielsweise durch Initiieren eines E/A-Vorgangs).

Ein frühes Beispiel einer virtuellen Maschine wurde im Jahre 1967 unter dem Namen CP-67 implementiert. Hierbei handelte es sich um ein virtuelles Maschinensystem für mehrere Anwender zum IBM System/360 Modell 67, das die funktionelle Simulation der Computerreihe /360, einschließlich des Modells 67 selbst, zur Verfügung stellte. Darüber hinaus konnte das System CP-67 eine interaktive Umgebung bereitstellen, die auf dem Konzept des „time sharing“ basierte.

Im Laufe der Zeit wurden zahlreiche Ansätze zur Implementierung virtueller Maschinen mit verschiedenen Abstraktionsgraden vorgestellt. Bei der vollständigen Systememulation (etwa bei Bochs [6]) wird eine Prozessorarchitektur simuliert, sodass Programme und Betriebssysteme, die für andere Computer entwickelt wurden, ausgeführt werden können. Die Paravirtualisierung bricht

mit dem Dogma der Kapselung und setzt voraus, dass das Gastbetriebssystem zum Zwecke der gemeinsamen Nutzung an der Virtualisierung kooperiert. Die Virtualisierung auf Betriebssystemebene abstrahiert Partitionen in einem einzigen Betriebssystem, führt jedoch zur weiteren Isolation von Anwendungen, die in verschiedenen Partitionen ausgeführt werden.

Die möglicherweise gängigste Form der Implementierung virtueller Maschinen auf einer den Betriebssystemen übergeordneten Ebene ist die Java Virtual Machine. Als Beispiel für eine virtuelle Maschine auf der Anwendungsebene abstrahiert diese Lösung die Laufzeitumgebung für Anwendungen durch Definition des Java-Bytecode-Standards. Interpreter oder Just-In-Time-Compiler im Laufzeitsystem dienen dann als Simulation der auf dem Zielsystem implementierten Java Virtual Machine. Mehrere Java-Anwendungen profitieren von dieser Isolation, wenn sie in verschiedenen Java Virtual Machines ausgeführt werden.

3.4.4 Ein breiteres Anwendungsfeld

Die Konzepte der Abstraktion, Teilung, Isolation und Aggregation lassen sich ganz allgemein auch auf die IT-Infrastruktur anwenden. Das führt zu einer Umgebung, in der Ressourcen dynamisch bereitgestellt und mithilfe geschäftlicher Regeln verwaltet werden. In dieser Umgebung werden Arbeiten geplant und überwacht, während sie die Ebenen des Systems durchlaufen (vom Netzwerk durch Firewalls, Webserver, Anwendungsserver, Datenbankserver und zurück). Unabhängig davon, ob diese Server auf physischen oder virtuellen Plattformen existieren, ob ein Ablauf Netzwerkzugriff und Datenspeicher beinhaltet, gilt, dass die Systeme so zugewiesen werden können, dass deren Nutzung maximiert wird. Gleichzeitig bleiben diese Systeme in aktuellen und zukünftigen Bereitstellungen flexibel und reaktionsfähig.

Virtualisierung wird in der Regel auf physischen Hardwareressourcen angewendet, kann jedoch auch auf nicht physische Ressourcen (zum Beispiel Software) und sogar für virtuelle Ressourcen eingesetzt werden. Virtualisierung kombiniert häufig mehrere physische Ressourcen zu gemeinsam genutzten Pools, aus denen den Anwendern virtuelle Ressourcen zugewiesen werden.

3.4.5 Warum ist Virtualisierung sinnvoll?

Virtualisierung trennt die Darstellung von Ressourcen für Anwender von den eigentlichen physischen Ressourcen. Dies schafft die Voraussetzung für die folgenden nützlichen Eigenschaften:

Gemeinsame Nutzung. Mehrere virtuelle Ressourcen können von einer physischen Ressource aus implementiert werden – entweder durch Zuweisung von Teilen der physischen Ressource oder durch deren gemeinsame Nutzung. Die

Anwender einer virtuellen Ressource haben den Eindruck, eine vollständige physische Ressource zu nutzen. Die gemeinsame Verwendung physischer Ressourcen kann deren Nutzung erhöhen und das Auslastungsmanagement vereinfachen, da sich die Aufteilung physischer in virtuelle Ressourcen kontrollieren lässt. Virtuelle Ressourcen können den Anwendern außerdem einen Grad der Isolation bieten, der bei direkter gemeinsamer Nutzung einer physischen Ressource nicht möglich wäre. Auf diese Weise werden die Sicherheit und die Verfügbarkeit verbessert. Zu den Beispielen für Virtualisierung zur gemeinsamen Nutzung zählen LPARs (Logical Partition), virtuelle Maschinen, virtuelle Festplatten und VLANs (Virtual Local Area Network).

Aggregation. Virtuelle Ressourcen können mehrere physische Ressourcen umfassen, sodass sich deren wahrgenommene Kapazität erhöht beziehungsweise deren Nutzung und Verwaltung vereinfacht. Zu den Beispielen für dieses Konzept zählen virtuelle Festplatten, die größer als jede verfügbare physische Festplatte sein können. Diese virtuellen Festplatten können sogar größer sein als alle verfügbaren physischen Festplatten zusammen (auf diese Weise entstehen erweiterbare, robuste virtuelle Festplatten).

Emulation. Virtuelle Ressourcen können Funktionen oder Merkmale aufweisen, die in den zugrunde liegenden physischen Ressourcen nicht zur Verfügung stehen. Zu den Beispielen hierfür zählen Architektur-Emulationssoftware, die mithilfe einer Prozessorarchitektur eine andere implementiert, sowie Internet Small Computer System Interface (iSCSI) zur Implementierung eines virtuellen SCSI-Bus in einem IP-Netzwerk. Auch virtueller Bandspeicher, der auf physischem Festplattenspeicher implementiert wird, lässt sich hier anführen.

Transparente Änderungen. Die Zuweisung von virtuellen zu physischen Ressourcen kann dazu führen, dass die physischen Ressourcen verändert werden können, ohne die virtuellen Ressourcen der Anwender zu beeinflussen. Beispiele hierfür finden sich bei der logischen Partitionierungsfunktion in Mainframes, mit der ein virtueller Prozessor automatisch von einem physischen Prozessor mit ersten Fehlersymptomen auf einen anderen physischen Prozessor verlagert werden kann. Ein weiteres Beispiel ist der RAID-Speicher, der mithilfe redundanter Festplatten Gerätefehler automatisch vor den Anwendern verbirgt.

3.4.6 Welche Vorteile ergeben sich?

Höhere Ressourcennutzung. Virtualisierung ermöglicht die gemeinsame dynamische Nutzung physischer Ressourcen und Ressourcenpools. Dies führt zu höherer Ressourcennutzung, vor allem bei variabler Auslastung, bei der die durchschnittlichen Anforderungen geringer sind als bei einer vollständig dedizierten Ressource.

Geringere Managementkosten. Virtualisierung kann die Mitarbeiterproduktivität verbessern, da die Anzahl zu verwaltender physischer Ressourcen verringert wird. Auf diese Weise wird ein Teil der Komplexität maskiert, was zur Vereinfachung allgemeiner Managementaufgaben und zu Möglichkeiten für die Automatisierung des Auslastungsmanagements führt.

Flexible Nutzung. Mit Virtualisierung können Ressourcen dynamisch bereitgestellt und neu konfiguriert werden, um wechselhafte Anwenderanforderungen zu erfüllen.

Mehr Sicherheit. Mit Virtualisierung lässt sich der kontrollierte, sichere Zugriff auf Daten und Services sicherstellen.

Höhere Verfügbarkeit. Mit Virtualisierung können physische Ressourcen ohne Beeinträchtigung der Anwender entfernt, aktualisiert oder geändert werden.

Mehr Skalierbarkeit. Durch Ressourcenpartitionierung und Aggregation kann eine virtuelle Ressource viel kleiner/größer sein als seine einzelne physische Ressource.

Interoperabilität und Investitionsschutz. Virtuelle Ressourcen können die Kompatibilität mit Schnittstellen und Protokollen sicherstellen, die in den zugrunde liegenden physischen Ressourcen nicht zur Verfügung stehen.

Verbesserte Bereitstellung. Virtualisierung kann die Ressourcenzuweisung gegenüber einzelnen physischen Einheiten verfeinern.

Da sich Unternehmen weiterentwickeln und Strategien einbringen, die die Implementierung echter integrierter End-to-End-Abläufe und gemeinsamer Verarbeitung vorantreiben, wird auch die Anzahl der Anwendungssysteme steigen, die diese Abläufe unterstützen. Diese Anwendungen werden immer mehr Server und verschiedene Technologien umfassen. Sofern ein Unternehmen von den hier beschriebenen Vorzügen profitieren möchte, müssen diese heterogenen, plattformübergreifenden Technologien ganzheitlich verwaltet, überwacht und bewertet werden, um sicherzustellen, dass die Anforderungen von Unternehmen und Service Level Agreements (SLA) erfüllt werden.

3.4.7 Virtualisierung und das Management von IT-Ressourcen

Virtualisierung kann als Technologie durch Management-Services ergänzt werden, um Administratoren beim Systemmanagement zu unterstützen. Beispielsweise kann eine Workload, die von der Hardware entkoppelt ist, unterbrechungsfrei von einem physischen System auf ein anderes verlagert werden.

Darüber hinaus können intelligente Funktionen der Virtualisierungsebene erkennen, ob eine Fehlfunktion bevorsteht; sie können die Isolation zwischen virtuellen Domains bereitstellen und sie können die Integrität der Workload sicherstellen, da Manipulationen bei der Installation und dem Starten erkannt werden.

Die Virtualisierung von Ressourcen und geschäftlichen Funktionen erfolgt zunehmend durch die Bereitstellung von Geschäftsprozessen und deren realer und virtueller Ressourcen als separate Services. Diese Services werden in einer serviceorientierten Architektur (SOA) zusammengefasst. In einer solchen Architektur werden logische Geschäftsabläufe in Form einer Reihe verbundener Aktivitäten dargestellt. Jede dieser Aktivitäten steht als Service zur Verfügung. Jeder Service kann unabhängig genutzt und kombiniert werden, wenn sich neue geschäftliche Möglichkeiten ergeben.

Das SOA-Modell der Bereitstellung logischer Abläufe und Ressourcen in Form unabhängiger Services, die dennoch verbundene Aktivitäten umfassen, ist ein idealer Ansatz für das Management von IT-Ressourcen. Ein gutes Beispiel hierfür sind die aktuellen Grid-Systeme.

Die nächste Phase der Virtualisierung besteht häufig in der Virtualisierung heterogener Umgebungen. Die Integration heterogener Systeme kann zur Vereinfachung der Infrastruktur und zur Senkung der Administrationskosten beitragen, da eine Person oder eine Gruppe die Aufgaben mehrerer Gruppen übernehmen kann, die bisher ihre eigenen Server überwacht haben.

Außerdem bietet sich hier die Möglichkeit eines Einblicks in die virtualisierte Umgebung, unabhängig von der Anzahl der verwendeten logischen oder physischen Server. Diese bildet auch die Grundlage für mehr geschäftliche Stabilität durch integrierte Virtualisierung. Fällt eine Anwendung, ein Server oder ein

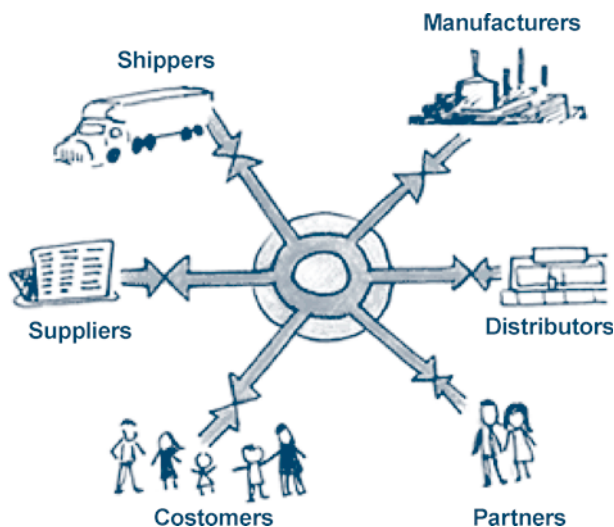


Abb. 3.14.

Netzwerk aus, kommt es nicht zu fehlerhafter Isolation (die dedizierte Notfallsysteme in Bereitschaft erfordert) oder zu Leerlaufzeiten während der Wiederherstellung. Stattdessen bedient sich die dynamische Natur der Virtualisierung vorhandener Ressourcen für andere Aufgaben oder weist vorhandene Server neu zu, um die fehlerhaften Server zu ersetzen.

Collaborative Processing markiert einen weiteren IT-Fortschritt und trägt dazu bei, organisatorische Silos zu überbrücken, sodass Unternehmen schneller Innovationen entwickeln und höheren wirtschaftlichen Nutzen bieten können. Dieser Ansatz nutzt Integration, um mehr interaktive Echtzeitanwendungen zu unterstützen und eine IT-Umgebung zu schaffen, die offener und intensiver gemeinsam genutzt wird und dynamischer ist. Diese Umgebung muss schnell auf wechselhafte Anwenderanforderungen reagieren können und Informationen sowie Ressourcen hinzufügen und neu zuweisen können, um geschäftliche Anforderungen zu erfüllen.

3.4.8 Ein Blick in die nahe Zukunft

Williams Gibson wird das folgende Zitat zugeordnet: „Neue Technologie ist aufgrund ihrer Natur außer Kontrolle und führt zu unvorhersehbaren Ergebnissen.“ Ein Beispiel dient der Validierung. Ein 16-Jähriger wird von seinem Vater zum Abendessen gerufen. Als er endlich aus seinem Zimmer kommt, fragt ihn sein Vater, warum er dafür zehn Minuten gebraucht hat. Der Sohn erklärt, dass er einen Bewerber als mögliches neues Gildemitglied prüfen, das Gespräch jedoch abkürzen musste, da der Mann seine Kinder zur Schule fahren wollte. Der Vater fragt: „Und wie alt war der Mann?“ Darauf der Sohn: „40 oder so.“ Der Vater hatte nie gewusst, dass sein Sohn in einem Online-Rollenspiel der Meister einer Gilde mit über 100 Mitgliedern war [7].

Solche virtuellen Welten ziehen zum Teil (wie bei „World of Warcraft“) bis zu sechs Millionen Spieler in ihren Bann und versinnbildlichen den nächsten Quantensprung der Datenverarbeitung – hin zu einem kollaborativen und intelligenten Arbeiten im Netz, das die reale Welt ergänzt und alle Möglichkeiten bietet, um uns in unserem Leben und bei unserer Arbeit zu unterstützen.

In den vergangenen zehn Jahren haben wir erlebt, dass sich die Art und Weise, wie wir kommunizieren und zusammenarbeiten, auf Inhalte zugreifen und Handel treiben, mit der Weiterentwicklung und dem Erfolg des Internets vollständig verändert hat. In den kommenden zehn Jahren werden wir einen mindestens ebenso tief greifenden Wandel erleben, da alle Arten innovativer neuer Anwendungen auf diesen technologischen Fortschritten aufbauen und neue geschäftliche und soziale Vorzüge schaffen. Diese Entwicklungen fördern einen ganzheitlichen Ansatz für das Management der IT-Infrastruktur im Sinne eines Unternehmens.

Literatur

- [1] Video-Interview mit William Gibson aus dem Jahr 1967
http://archives.cbc.ca/IDC-1-69-1587-10799/life_society/60s/clip11Reference
- [2] Box Office Mojo: The Matrix
<http://www.boxofficemojo.com/movies/?id=matrix.htm>
- [3] Website von William Gibson
<http://www.williamgibsonbooks.com/>
- [4] Online Etymology Dictionary
<http://www.etymonline.com>
- [5] Konzepte für virtuellen Speicher und virtuelle Maschinen
<http://www.research.ibm.com/journal/sj/112/ibmsj1102B.pdf>
- [6] Bochs Emulator
<http://bochs.sourceforge.net/>
- [7] Aus einem Gespräch anlässlich des Directors Guild of America's Digital Day, Los Angeles, 17. Mai, 2003.

3.5 Systemautomatisierung

Jürgen Schneider

3.5.1 Überblick

Der Begriff Automation beschreibt im Allgemeinen die Fähigkeit, den Computer selbst einzusetzen, um einige der täglich anfallenden Aufgaben im IT-Umfeld auf programmatische Weise zu managen und zu optimieren. Die Systemautomatisierung (SA) hat ihren Ursprung in dem sehr verständlichen Bedürfnis der IT-Systembediener, einige der wiederholt anfallenden und vordefinierten Aktionen in kleinen Einheiten von Befehlsscripts zu „programmieren“. Die IT-Systembediener führen in der Regel folgende Aufgaben durch:

- Überwachung der IT-Systeme
- Analyse und Korrelation maßgeblicher Informationen
- Eingabe von Befehlen zur Änderung des Systems, um zu dem gewünschten Verhalten zurückzukehren

Der hier grob beschriebene Aufgabenablauf (Überwachung, Analyse, Planung, Ausführung) ist ein häufig eingesetztes Muster für die Erstellung autonomer Komponenten. Im Rahmen der IBM Autonomic-Computing-Initiativen wurde der MAPE-K-Regelkreis (Monitor, Analyze, Plan, Execute – Knowledge) als Referenzarchitektur entwickelt.