

SAP nach dem Ausfall: Best Practices zum Wiederanlauf komplexer Systeme

Franz Diegruber, Libelle AG

Um die Verfügbarkeit heutiger SAP-Systeme sicherstellen zu können, reicht es nicht mehr aus, Datenbanken und Datenbankinhalte verfügbar und sicher zu machen. SAP-Systemumgebungen bestehen mittlerweile aus einem komplexen Konglomerat von Komponenten, die in verschiedenen Abhängigkeiten zueinander stehen.

Insbesondere mit der Nutzung der SAP-NetWeaver-Funktionalität, die aus dem J2EE-Stack resultiert, wird die Komplexität durch eine Zunahme an Abhängigkeiten und Komponenten weiter vergrößert. Ein zusätzlicher Treiber der Komplexität ist, dass mittlerweile die Geschäftsprozesse verteilt über verschiedene Systeme ablaufen und somit insbesondere ein Wiederanlauf der Systeme Fragen über deren Konsistenz und auch deren Wiederanlaufszszenarien aufwirft. Klassische Übernahmeverfahren und Methoden zur Sicherstellung der Betriebsvitalität funktionieren hierbei nicht mehr oder nur noch eingeschränkt.

RPO, RTO und RCO

Klassisch bemisst sich die Qualität von Wiederanlauf- beziehungsweise Disaster-Recovery-Szenarien mit den Größen der Recovery Point Objective (RPO), also wie viel Datenverlust im Fehlerfall tolerierbar ist, sowie mit der Recovery Time Objective (RTO), also wie lange der Wiederanlauf dauert. Diese Kriterien werden dabei allerdings auf der Basis einzelner Systeme und nicht nach dem Verlauf der Geschäftsprozesse definiert.

Die Verteilung der Daten und deren Datenentität über eine Vielzahl von Systemen in Kombination mit einer ganzheitlichen Betrachtung der Daten über die Geschäftsprozesse stellt die IT vor die Herausforderung, Daten systemübergreifend konsistent wiederherstellbar zu machen. Das Prinzip der Logical Units of Work (LUWs) funktioniert originär jedoch

nur innerhalb abgeschlossener Datenumfelder wie der jeweiligen Datenbanken.

Schnittstellendaten sowie Daten in Filesystemen besitzen größtenteils keinen transaktionalen Konsistenz-Algorithmus. Die Erzeugung einer Gesamtsystem-Konsistenz ist somit unter ganzheitlicher Betrachtung von Datenbanken, Filesystemen und Schnittstellen besonders schwierig. Konsistenz muss quantifizierbar und überprüfbar hinterlegt und umgesetzt werden.

Eine Angabe über Recovery Time Objective (RTO) und Recovery Point Objective (RPO), die auf der Basis einzelner Systeme definiert und errechnet werden, ist aus Sicht der Geschäftsprozesse nicht ausreichend. Eine Cross-System-Datenintegrität muss definiert und umgesetzt sein. Hierzu definiert die Recovery Consistency Objective (RCO) über alle Systeme hinweg die benötigten Konsistenz-Anforderungen (siehe *Abbildung 1*).

Im Detail beschreibt die RCO die erlaubte Abweichung wiederhergestellter Datenbestände nach einem Systemvorfall. Sie gibt somit an, wie groß der Unterschied der Geschäftsdaten-Basis verteilt über die beteiligten Systeme sein darf, sowohl qualitativ als auch quantitativ. Dabei sind für alle an einem Geschäftsprozess beteiligten Systeme sowohl die durch LUWs abgesicherten Daten in einer Datenbank als auch die Daten in Filesystemen und Schnittstellen zu berücksichtigen. Insbesondere bei externen Schnittstellen wie etwa in Logistik-Systemen ist eine solche systemübergreifende Datenkonsis-

tenz von entscheidender Bedeutung. Ist die RCO für die Logistikprozesse wesentlich kleiner als 100 Prozent, entspricht die Wahrnehmung im System nicht mehr der Wirklichkeit. Das bedeutet, dass sowohl quantitative (wie aktuelle Lagerbestände) als auch qualitative Informationen (wie aktueller Standort der Ware) aufwändig nachgearbeitet werden müssten. In gleichem Zusammenhang verwendete unkritische Daten (wie Telefonnummer eines Kunden) können mit einem wesentlich geringeren RCO auskommen, da diese in Relation einfacher nachpflegbar sind und der Verlust bzw. eine Inkonsistenz keinen wesentlichen geschäftlichen Schaden verursacht.

Besonders wichtig ist es, im Verlauf der Requirement-Analyse für die Katastrophenvorsorge die richtigen Anforderungen für RPO, RTO und RCO zu definieren. Fragt man den Geschäftsbereich nach der RPO für den Katastrophenfall, bekommt man im ersten Moment immer die Aussage, dass kein Datenverlust tolerierbar sei. Unter diesen Bedingungen ist zu beachten, dass eine solche Anforderung für einen tatsächlichen Katastrophenfall sich zum größten Teil eher im Bereich von 10 bis 30 Minuten bewegt. Bei einer tatsächlichen Katastrophe, wie sprechen hier vom Brand eines Datacenters, Explosionen, Flugzeugunglücken ist ein Datenverlust von 30 Minuten meist das kleinere Problem. Der Wiederanlauf des Geschäfts, Zugang für Mitarbeiter zu den Daten, Arbeitsplätze für Mitarbeiter und insbesondere der Wiederanlauf der Geschäfts- und

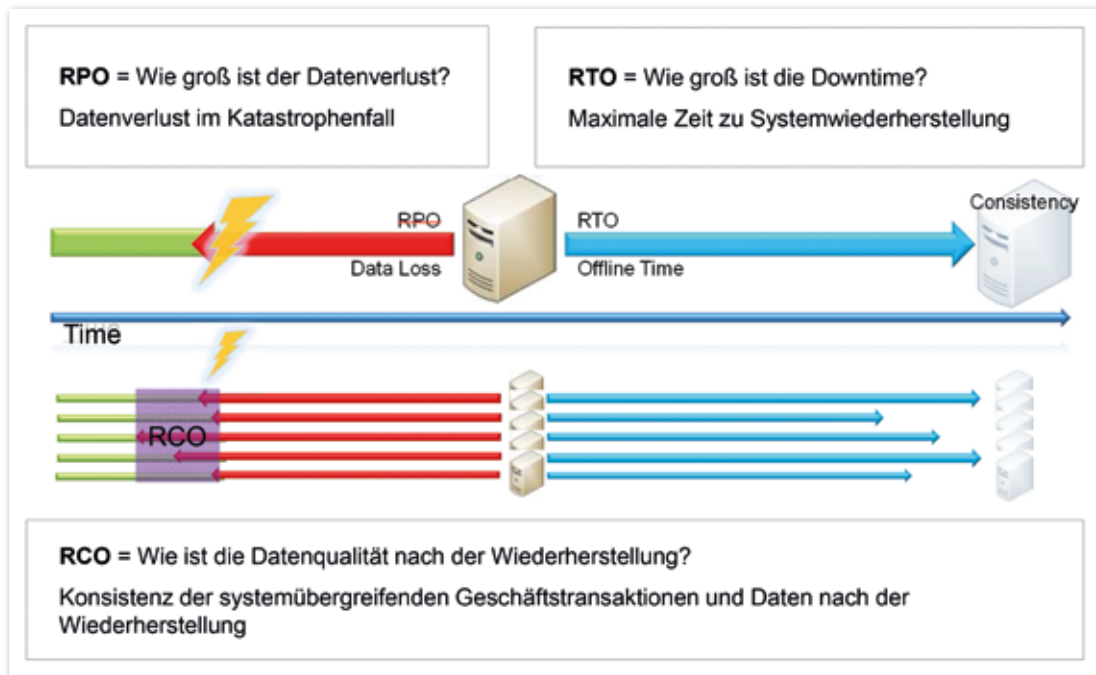


Abbildung 1: RPO, RTO und RCO

IT-Prozesse sind wesentliche und weitere kritische Bestandteile eines Wiederanlaufs von ganzen Rechenzentren.

Evolutionäre Entwicklung einer Katastrophenvorsorge

Um Geschäftsprozesse in der heutigen Zeit richtig abzusichern, ist es notwendig, dass die Vorsorgelösung sich zyklisch an die sich ändernden Bedürfnisse anpasst. Von entscheidender Bedeutung ist schon in der Phase der Identifizierung der kritischen Geschäftsprozesse und Systeme eine Klassifizierung und Gruppierung dieser nach den Verfügbarkeits- und Konsistenzkriterien. Eine typische Klassifizierung in diesem Rahmen kann durch folgende Klassen beschrieben werden:

- (D) Vitale Business-Daten (wie SAP ERP)
- (I) Interfaces und Schnittstellen-Systeme (wie SAP PI)
- (B) Business-Unterstützungs-Systeme (wie BOBJ)
- (S) Support-/Infrastruktur-Systeme (wie SAP SolMan)
- (E) End User (wie SAP EP)

In der Requirement- und Solution-Design-Phase wird die Lösung vom generischen Ansatz bis hin zu Systemen und deren Ab-

hängigkeiten runtergebrochen; auf der Basis wird dann die Infrastruktur-Architektur für die Katastrophenvorsorge erstellt. Die Business-Continuity-Prozesse und die Infrastruktur werden zu einer integrierten Lösung verbunden, um im Wiederherstellungsfall die Qualität der Prozesse, Abläufe und Technologie sicherzustellen (siehe Abbildung 2).

Die Implementierung rollt das Design für die Systeme aus, während in der Run-Phase die Systeme betrieben werden. Ein wichtiger Bestandteil in der Run-Phase sind zyklische Tests der Lösung. Diese sollten mindestens einmal jährlich durchgeführt werden, um die eingesetzte Technologie, Dokumentation und Prozesse zu prüfen.

Die Verifikation der Lösung sollte in einem Zyklus von drei bis fünf Jahren erfolgen. Der Scope dieser Verifikation ist es, die Implementation erneut gegen die Geschäftsprozesse zu mappen. Sobald die Lösung den Anforderungen der Geschäftsprozesse nicht mehr gerecht wird, muss die Lösung den Zyklus erneut durchschreiten.

SAP ganz klassisch

Die klassische Implementierung eines SAP-Systems umfasst die Installation ei-

ner Datenbank, einer SAP Central Instance und optionaler SAP Application Server. Dieses Konzept basiert auf der SAP ABAP Engine, die alle Daten und Funktionen in der korrespondierenden Datenbank ablegt. Soll eine solche SAP-Umgebung verfügbar gemacht werden und in der Lage sein, sowohl logischen als auch physikalischen Fehlern entgegenzuwirken, ist es notwendig, die zentrale Komponente – die SAP-Datenbank – abzusichern. Eine solche Absicherung kann etwa mit der DBShadow-Technologie erfolgen.

Bei einer solchen Absicherung sind alle Single Points of Failure (SPOFs) in Abhängigkeit ihrer Priorität und ihres Einflusses auf die Verfügbarkeit des SAP-Systems mit einer farblichen Markierung versehen, wobei die roten Markierungen die zentrale Vitalität des SAP-Systems im Fall der Übernahme darstellen. Ist nur die Datenbank in einer SAP-Umgebung gespiegelt, werden noch nicht alle wesentlichen SPOFs des SAP-Systems nachhaltig gesichert. Weitere SPOFs wie die SAP Central Instance mit Abhängigkeiten an eine IP-Adresse beziehungsweise an einen Hostnamen und auch das Filesystem, das unter anderem die Profile und Parameter, die Job Logs sowie die Executables des SAP-Systems enthält und typischerweise

durch ein NFS-Share allen untergeordneten Application-Servern zur Verfügung gestellt wird, sind in dieses Übernahme- und Verfügbarkeits-Verfahren nicht integriert. Eine vollständige Absicherung der SAP-Umgebung erfolgt somit ausschließlich dann, wenn die folgenden Elemente im Übernahmeszenario berücksichtigt sind:

- SAP-Datenbank
- SAP ABAP Central Services (Bestandteil der Central Instance wie ENQ, MSG etc.)
- SAP-Filesystem und dazugehöriger NFS-Share
- IP-Adresse für SAP ABAP Central Services

Typischerweise ist die Änderungsrate im klassischen Filesystem der SAP nicht groß, auch das Starten und Stoppen der SAP Central Instance inklusive der Übernahme der IP-Adresse und Hostnamen ist von der Durchführungskomplexität im Prinzip gering. Deswegen erfolgen in einigen Installationen eine Übernahme des Filesystems und der IP-Adresse sowie das Starten der SAP Central Instance manuell. Diese Lösung ist allerdings nicht empfehlenswert, wenn sich mehrere Systeme in einem Ver-

bund befinden oder ein komplexes Übernahmeszenario existiert.

Aus diesem Grund sollte die SAP-Umgebung in ein Szenario überführt werden, das sowohl die Komponenten zur Spiegelung der Datenbank enthält als auch solche, die für die Absicherung des Filesystems und der SAP Central Services sorgen. Eine Absicherung des Filesystems für SAP kann beispielsweise mittels FSShadow erfolgen und hiermit analog zur zeitversetzten Spiegelung der Datenbank. Dabei wird der entsprechend für die Datenbank definierte Zeitversatz auch auf das Filesystem angewendet, um eine zeitliche Konsistenz für Datenbank und Filesystem, insbesondere für die Job-Logs, zu erzeugen.

Eine Umschaltintegration der VIP und des virtuellen Hostnamen zum Beispiel mittels SwitchApplication gewährleistet ein problemfreies Management und Umschaltverfahren sowohl bei Ausfällen als auch bei der Systemwartung auf den entsprechenden Knoten. Dies garantiert eine durchweg hohe Verfügbarkeit des SAP-Systems.

SAP mit J2EE

Eine neue Herausforderung stellt die Implementierung eines SAP-Systems auf Basis der aktuellen SAP-NetWeaver-Pro-

duktlinie dar. Der J2EE-Stack, auch Java-Stack genannt, fügt nicht nur einen neuen Stack an Komponenten in die Umgebung ein. Damit wird auch die bisher bekannte Vorgehensweise, alle für die Anwendung benötigten Programme und Daten innerhalb der Datenbank vorzuhalten, aufgebrochen. Für die Ausführung des SAP-Systems benötigte Logik wird ab sofort mit dem J2EE-Stack auch im Filesystem vorgehalten. Dies weist dem Filesystem von nun an eine zentrale Rolle beim Erhalt der Vitalität der SAP-Systemumgebung zu.

Zusätzlich zu den klassischen Systemkomponenten gilt es somit, die folgenden Bestandteile des J2EE-Stacks entsprechend in die Verfahren der Hochverfügbarkeit und Disaster-Vorsorge zu integrieren:

- SAP J2EE Central Services (Bestandteil der Central Instance wie ENQ, MSG etc.)
- SAP J2EE Filesystem für JAR Files und dazugehöriger NFS-Share
- IP-Adresse für SAP J2EE Central Services

Bei der Absicherung der J2EE-Komponenten kommt das gleiche Verfahren zum Einsatz, das auch schon bei der vollständigen Absicherung des ABAP-Stacks genutzt

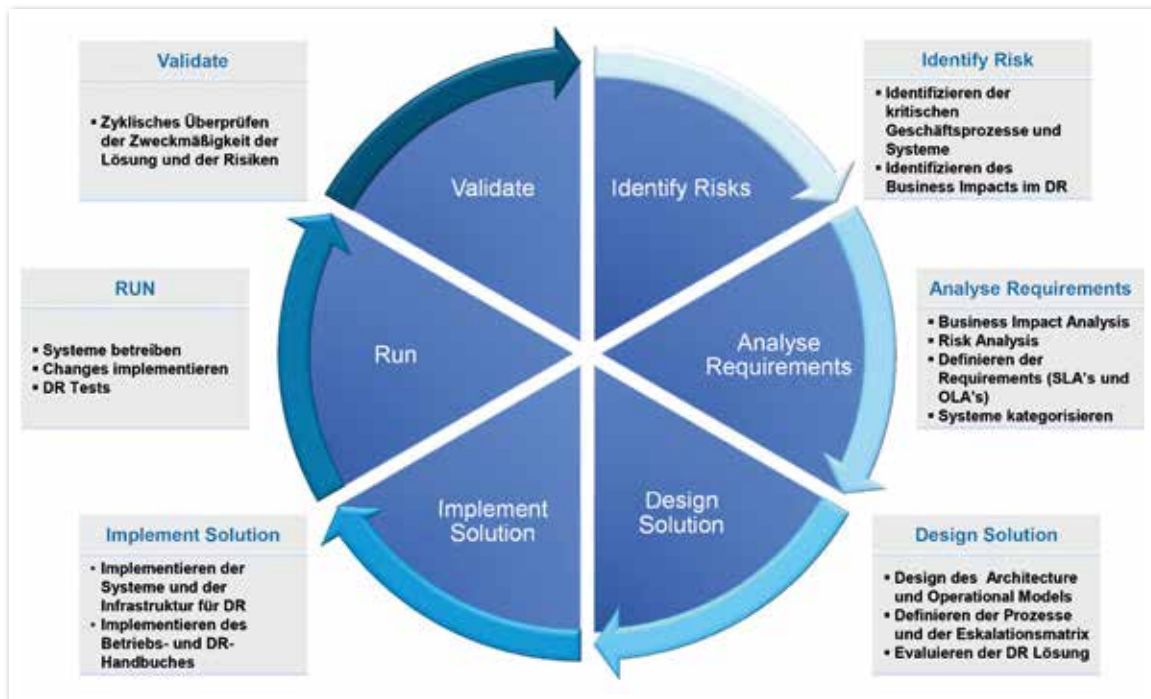


Abbildung 2: Die Projektphasen

wird. Die Architektur einer Verfügbarkeitsumgebung, die sowohl ABAP- als auch J2EE-Stack integriert, sichert das zentrale Filesystem mittels FSShadow analog der Datenbank mit Zeitversatz auf die sekundäre Seite ab. Insbesondere beim J2EE-Stack ist die zeitliche Gleichschaltung der Datenbank und des Filesystems von Bedeutung, da sich vitale und sich ändernde Bestandteile im Filesystem befinden. Manuelles Kopieren oder ein Kopieren ohne Zeitversatz sind hierbei nur die zweitbeste Lösung, die im Fall einer Übernahme zu gravierenden Komplikationen beim Start und Betrieb des Notfallsystems führen kann. Eine integrierte Umschaltung und Übernahme der IP-Adressen und Hostnamen erfolgt analog zur ABAP-Umgebung, etwa mittels SwitchApplication.

Zusätzlich zu den hier gezeigten Komponenten ist eine Absicherung des SAP Web Dispatcher, sofern dieser zum Einsatz kommt, sowie des Software Deployment Manager (SDM) erforderlich. Diese Integration des Web Dispatcher kann ebenfalls durch SwitchApplication erfolgen. Ob eine Absicherung des SDM erfolgen muss oder dieser ausschließlich über „Cold Standby“-Lösungen abgesichert wird, hängt vom Änderungsvolumen und der Dringlichkeit der entsprechenden Änderungen für den Betrieb der SAP-Systeme ab.

Weiterführend gehören gegebenenfalls genutzte zusätzliche Komponenten von SAP-Landschaften in ein ganzheitliches Konzept für Verfügbarkeit und Katastrophenvorsorge. Unter anderem sind in bestimmten Installationen TREX (Suche), das System Landscape Directory (SLD), das Enterprise Portal (EP) und die Process Integration (PI), um nur einige zu nennen, von elementarer Bedeutung. Generell gilt jedoch, dass die gezeigten Verfahren für diese Komponenten mit leichter Abwandlung angewandt werden können.

Best Practices: Übernahme-Workflow eines SAP-Systems

Die Übernahme eines SAP-Systems im Katastrophenfall sollte nach einer streng definierten Methode erfolgen. Als Best Practices hat sich folgender Workflow bewährt:

- Notification
- Preparation

- Failover
- ORA Activation
- SAP Start
- Final Switch
- Verification

Der Ablauf integriert eine Notification für alle betroffenen Beteiligten nach einem Disaster. Die Umschaltung wird durch Pre-Check vorbereitet und stellt sicher, dass alle benötigten Komponenten erreichbar sind. Der Failover erfolgt durch die Libelle für alle Systeme. Ist eine RCO über unterschiedliche Systeme notwendig, muss an dieser Stelle der Konsistenzpunkt über alle Systeme definiert werden. Nach der Umschaltung sollte die Oracle-Datenbank auf deren Vitalität überprüft werden, um anschließend SAP zu starten. IP-Adressen und Hostnamen werden dabei mit übernommen. Anschließend ist das SAP-System zu verifizieren und an die User freizugeben. Falls möglich, sollte ein Backup eingeplant werden, um eine Wiederaufsetzbarkeit der Systeme zu gewährleisten.

Fazit

Eine Absicherung von SAP-Systemen für den Katastrophenfall kann generell durch unterschiedliche Technologien gewährleistet werden. Um die Umsetzung den Anforderungen der SAP-Systeme gerecht werden zu lassen, ist ein entscheidendes Kriterium, dass man einen ganzheitlichen Ansatz für die Katastrophenvorsorge wählt. Die Schritte einer Umschaltung sollten dabei auf einer DIN-A4-Seite zusammenfassbar sein, um die Möglichkeit des Auftretens einer Fehlbedienung in solchen Fällen so klein wie möglich zu halten.

Franz Diegruber
fdiegruber@libelle.com

Libelle SystemCopy



- ✓ Automatisierte und optimierte Vor- und Nacharbeiten
- ✓ Ohne in Ihre SAP-Umgebung einzugreifen bzw. diese zu verändern
- ✓ Ohne aufwändige Vorplanung
- ✓ Mit minimaler Durchlaufzeit
- ✓ Bei gleichbleibender Qualität der Kopie

... mit deutlich reduzierten Prozesskosten



Hans-Joachim Krüger
Chief Technology Officer
Libelle AG

Erfahren Sie mehr:
www.Libelle.com/systemcopy



ORACLE Gold Partner



Libelle

Libelle AG

Gewerbestr. 42 • 70565 Stuttgart, Germany
T +49 711 / 78335-0 • F +49 711 / 78335-148
www.Libelle.com • sales@libelle.com