

Continuous Software Integration & Delivery

Erfahren Sie in dieser Serie unserer avato notes, mit welchen marktüblichen Tools Sie eine „Continuous Software Integration & Delivery“ aufbauen können, welchen Herausfor-

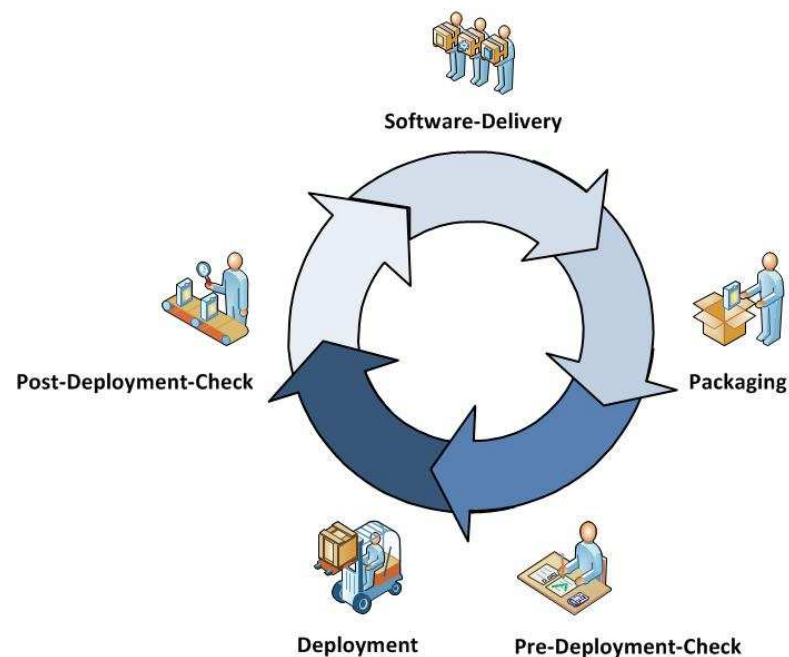
derungen Sie dabei begegnen und wie Sie diese überwinden können. Auf der Basis umfangreicher Projekterfahrung in Enterprise-Umgebungen berichten wir hier über mögliche Ein-

satzszenarien der gängigsten Tools zur Softwareintegration im Markt sowie deren Funktionseinsatz im Softwarestack.

Definition „Continuous Software Integration & Delivery“:

avato versteht unter „Continuous Software Integration & Delivery“ den fortlaufenden toolgestützten Integrations- und Lieferprozess in alle Applikationsumgebungen (Entwicklung, Test sowie Produktion) von der Codeübergabe durch den Softwarelieferanten bzw. der Entwicklung (Softwarelieferung) bis zur Produktivsetzung. Dies umfasst die Schritte Software-Lieferung, Packaging, Pre-Deployment-Check, Deployment und Post-Deployment-Check.

Continuous Software Integration Lifecycle:



Herausforderungen der konventionellen Softwareintegration

Wer kennt die täglichen Herausforderungen nicht? Eine Vielzahl unterschiedlicher Projekte mit diversen Prozess- und Tool-Varianten. Es beginnt meist mit differierenden Anlieferungsprozessen und geht weiter über

fehlende Revisionssicherheit, Konfigurationsprobleme und eingeschränkte Testmöglichkeiten. Hinzu kommen hohe Aufwände und eine große Fehleranfälligkeit durch individuelle und in Teilen manuelle Ver-

fahren. Die Deployment-Laufzeiten sind lang und die Komplexität des Systems führt zu einer geringen Anzahl von Deployments. Nur durch umfangreiches technologisches Know-how, genaue Kenntnis der

Umgebungen (meist an Kopfmonopole gebunden) und eingesetzten Technologien ist eine erfolgreiche Integration zu gewährleisten. Typisch sind im Operations die Fehleranfälligkeit durch nicht standardisierte Übergabeverfahren, manuelles Nacharbeiten und individuelle Softwarelieferungen je Umgebung.

Standardisierung und Automatisierung

Aber wie sieht ein mögliches Lösungsszenario aus? Zentrale Elemente sind die Standardisierung der Softwarelieferungen, Konfigurationen, Paketierungen und Deployment-Verfahren. Erst einheitliche Softwarelieferungen inklusive Konfigurationen in einen zentralen Eingangskanal bieten fehlerresistente, revisions sichere und schnelle Integrationen. Darauf aufbauend lässt sich die Integration (Build und Deployment) weitestgehend automatisieren. Auch die Installation in beliebige Umgebungen kann durch geeignete Verfahren automatisiert und ohne aufwändige manuelle Nacharbeiten auf sämtlichen Plattformen dargestellt werden.

Vorstellung der Werkzeuge

Die Serie bietet eine praxisorientierte Vorstellung von Softwareintegrationswerkzeugen geordnet nach Einsatzgebieten und Funktionalitäten. Die unten genannten Anforderungen sind der Bewertungsmaßstab und dienen am Ende der Serie zum transparenten Vergleich. Die Liste der untersuchten Tools erhebt keinen Anspruch auf Vollständigkeit – deckt aber die am Markt am häufigsten eingesetzten ab. Wir beginnen die Vorstellungsreihe mit dem BitRock InstallBuilder.

Einsatzspektrum der Tools

In einer Zeit, in der im IT-Bereich zunehmend Kosten reduziert werden müssen, ist eine effektive Softwareintegration unerlässlich. Oft werden die Kostentreiber der Softwareintegration in der Betrachtung vernachlässigt, der Business-Case für den Einsatz eines der nachfolgend beschriebenen Werkzeuge nicht vollständig betrachtet. Was ist aber bei der Einführung eines Toolsets zu beachten, was sollte zum Einsatzspektrum gehören? Eine durchdachte Lösung für „Continuous Software Integration & Delivery“ muss durch die schnelle Bereitstellung von Umgebungen, kurze Deploymentzyklen und reduzierte Build- und Deployment-Aufwände glänzen. Zudem kann eine solche Lösung vollständige Kostentransparenz (Managed Service) schaffen, da eine preisliche Ausdifferenzierung bis auf Einzelschritzebene, z.B. für ein einzelnes Deployment, erfolgen kann. Idealerweise sind die Werkzeuge zentral gesteuert und bieten ein Interface zur Prozesssteuerung sowie Übersichten zu Status des Service.



Spezifikation und Bewertung des geforderten Leistungsumfanges

Um das beschriebene Einsatzspektrum abzudecken und Werkzeuge in adäquater Weise vergleichen zu können, ist es zwingend notwendig, den geforderten Leistungsumfang in detail zu spezifizieren.

Die ausgewählten Bewertungskriterien orientieren sich an der

Anforderung, ein komplettes End-to-End-System zur „Continuous Software Integration & Delivery“-Plattform realisieren zu können. Die Betrachtungen konzentrieren sich deshalb auf die Analyse von Key-Features, die zum Aufbau einer solchen End-to-End-Lösung zwingend erforderlich sind.

Die Bewertung der zu testenden Werkzeuge erfolgt anhand einer ungewichteten Bewertungsmatrix nach „Anforderung erfüllt“, „Anforderung teilweise erfüllt“ oder „Anforderung nicht erfüllt“.

Anforderung	Erklärung
Zentrale Lieferverwaltung	Annahme und Verwaltung interner/externer Softwarestände, um Zulieferung von Abteilungen/Firmen zu managen.
Parameterverwaltung	Verwaltung umgebungs- und applikationsabhängiger Parametersätze, um den Einsatz im Multiserverbetrieb zu gewährleisten.
Parameterersetzung	Ersetzungsmechanismen, um Konfigurationsdateien mit umgebungsabhängigen Parametern zu befüllen.
Optionale Konfigurationsdateien	In Enterprise-Umgebungen zwingend erforderlich, um sicherheitsrelevante Parameter bzw. zum Lieferzeitpunkt noch nicht feststehende Parameter erst zum Deploymentzeitpunkt zu ersetzen (Last-Minute-Eingriff, ohne ein neues Paket erzeugen zu müssen).
Komponentenbasiertes Deployment	Erstellung von Filtervorlagen, um nur bestimmte Teile eines Deployments automatisch ablaufen zu lassen.
Zentrale Buildplattform	Zentraler Buildmechanismus, um reproduzierbare Softwarepakete zu gewährleisten.
Snapshot-Buildplattform	Bereitstellung der Build- und Packagingumgebung, um Abteilungen/Firmen eine Validierungsplattform zu bieten.
Management interner Abhängigkeiten	Definition interner Abhängigkeiten beim Deployment, um keine inkonsistenten SW-Stände zu erzeugen.

Plattformunabhängigkeit	Einsatz des erzeugten Packages in Multiserver- / Multibetriebssystemumgebungen.
In sich abgeschlossenes Paket (self-contained)	Einfache Handhabung und geringe Fehleranfälligkeit durch Bereitstellung EINES Lieferpaketes. Es beinhaltet Software, Konfigurationen und Installationslogik.
Clusterfähigkeit	Rollout der Software im Clusterbetrieb, um mehrmaliges Deployment zu vermeiden.
Plug-In-Fähigkeit	Möglichkeit der Spezifikation eigener Plug-In-Module, um einen Multiprojektbetrieb zu ermöglichen.
Integration in übergeordneten Workflow	Bereitstellung einfacher Schnittstellen, um das System in bereits bestehende Tracking- und Ticketingsysteme integrieren zu können.
Automatisierte Pre- / Posttests	Gewährleistung automatisierter Pre- und Postbuildtests, um die Einsatzfähigkeit des Systems / des Servers nicht zu beeinträchtigen.

Impressum

Datum: August 2012

Autoren: Uwe Bloch
Andreas Fränkle

Kontakt:

marketing@avato-consulting.com
www.avato-consulting.com

© 2012 avato consulting

Haben Sie weitere Fragen? Wir beraten Sie gerne:
marketing@avato-consulting.com