

Visual Modeler



Best Practices

Release 9.1

Visual Modeler



Best Practices

Release 9.1

Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen unter „Bemerkungen“ auf Seite 43 gelesen werden.

Copyright

Diese Ausgabe bezieht sich auf Version 9.1 von Visual Modeler und alle nachfolgenden Releases und Modifikationen, bis dieser Hinweis in einer Neuauflage geändert wird.

© Copyright IBM Corporation 2007, 2011.

Inhaltsverzeichnis

Kapitel 1. Best Practices für Sicherung und Wiederherstellung 1

| | |
|---|---|
| Überblick über die Deploymentarchitektur | 1 |
| Deploymentinfrastruktur | 2 |
| Sicherungsstrategien. | 3 |
| Strategie zur Datenbankwiederherstellung | 3 |
| Wiederherstellung der Datenbank | 4 |
| Wiederherstellung von Anwendungsserver und Web-Server | 4 |

Kapitel 2. Symbole in der Benutzerschnittstelle 5

Kapitel 3. Artikelmodellierung 7

| | |
|---|----|
| Absolute und relative Pfade für die Artikelmodellierung verwenden | 7 |
| Artikelmodellierung planen | 7 |
| Betrachtungen zur Entwurfsphase der Artikelmodellierung | 8 |
| Dimensionierung eines Artikelmodells. | 8 |
| Artikelmenge über Popup-Steuerelemente angeben | 10 |
| Modelleigenschaften definieren. | 12 |
| Modelleigenschaftsnamen definieren | 13 |
| Eindeutige Modelleigenschaften definieren | 13 |
| Hierarchie mit Modelleigenschaften definieren. | 14 |
| Mehrere Eigenschaften mit identischem Wert definieren | 14 |
| Arbeitsblätter für die Eigenschaftszuweisung verwenden | 15 |
| Verkettungen in Eigenschaftsformeln vermeiden | 16 |
| Regeln definieren | 17 |
| Regelbedingungen definieren | 17 |
| Reihenfolge von Regelfragmenten festlegen. | 17 |
| Allgemeine Regeln erstellen | 17 |
| Formeln zum Definieren von Regeln verwenden | 18 |
| Pfadangaben in Regeln vermeiden. | 18 |
| Einschränkungstabellen und Regeln im Vergleich | 19 |

| | |
|---|----|
| Leistungsverhalten von Modellen verbessern | 19 |
| Modellierungstools | 20 |
| Traceprotokoll verwenden | 20 |
| Berichtstool für Modelle | 22 |
| Tools für Belastungstests | 22 |
| Tool für Cachestatus | 24 |
| Leistung | 24 |
| Regeln für das Leistungsverhalten eines Modells | 24 |
| Eigenschaften für das Leistungsverhalten von Modellen | 25 |

Kapitel 4. Daten archivieren. 27

Kapitel 5. Statistikdaten aktualisieren 29

| | |
|--|----|
| Statistikdaten für eine Oracle-Datenbank aktualisieren | 29 |
| Statistikdaten für eine SQL Server-Datenbank aktualisieren | 29 |

Kapitel 6. Richtlinien für JVM-Speichereinstellungen und Leistungsoptimierung 31

| | |
|--|----|
| JVM-Speichereinstellungen anpassen | 31 |
| Zusätzliche Leistungsoptimierung | 32 |
| Garbage-Collection-Aktivitäten verfolgen | 32 |

Kapitel 7. Log Analyzer. 35

| | |
|---|----|
| Tägliche Log Analyzer-Berichte konfigurieren | 36 |
| Tägliche Log Analyzer-Berichte automatisieren | 36 |
| Tägliche Log Analyzer-Berichte konfigurieren | 37 |
| Empfohlene Verzeichnisstruktur | 37 |
| Log Analyzer konfigurieren | 38 |

Index 41

Bemerkungen. 43

Kapitel 1. Best Practices für Sicherung und Wiederherstellung

Bei durchdachten Wiederherstellungsplänen steht Vorbeugung im Mittelpunkt. Wenn Sie eine stabile und leistungsfähige Umgebung aufbauen, redundante Systeme installieren, Richtlinien für eine regelmäßige Sicherung und Wiederherstellung festlegen und in regelmäßigen Abständen Tests durchführen, die sich aus der Sicherung wiederherstellen lassen, können Sie die Auswirkungen von Katastrophenfällen begrenzen, indem Sie Sicherheitseinrichtungen für jede Schicht Ihrer Systeminfrastruktur schaffen.

Einige Entscheidungen zu den Richtlinien für Sicherung und Wiederherstellung müssen auf der Basis von geschäftlichen Kriterien getroffen werden. Welchen Nutzen bringt die tägliche Verfügbarkeit der Site für Ihr Unternehmen? Welche Menge an verlorenen Daten können Sie verkraften? Wie lange darf Ihre E-Commerce-Site für die Kunden nicht verfügbar sein? Wie sehen die Kompromisse bei Zeit- und Kostenaufwand für verschiedene Sicherungs- und Wiederherstellungslösungen aus? Anhand der Antworten auf diese Fragen können Sie Ihre Anforderungen bzgl. Sicherung und Wiederherstellung leichter bestimmen.

Mit dem einfachsten Sicherungssystem lassen sich möglicherweise Daten und eine Kopie Ihrer Anwendung auf Band oder anderen fernen Einheiten speichern oder Daten auf einem fernen Rechenzentrum ablegen. Eine bessere Strategie wäre sicherlich, für jeden Teil Ihrer Installation redundante Systeme zu implementieren, sodass beim Ausfall eines Systems bereits ein anderes System verfügbar ist. Die leistungsfähigste Lösung ist eine Lösung, bei der ein Spiegelimage Ihrer Site erstellt und an einem fernen Standort abgelegt und das Image in regelmäßigen Abständen mit den Livedaten synchronisiert wird. Diese Lösung ist zwar am teuersten, ermöglicht aber eine unverzügliche Wiederherstellung. Die anderen Lösungen sind weniger kostenintensiv, erfordern aber mehr Zeit und Aufwand bei der Wiederherstellung.

Überblick über die Deploymentarchitektur

Die Einrichtung einer mit zahlreichen Funktionen ausgestatteten Entwicklungsumgebung erleichtert nicht nur die Aktualisierung und Verwaltung der Site, sondern ermöglicht auch bei Bedarf die schnelle Wiederherstellung einer vollständigen Anwendung in einem Schritt. Die Deploymentarchitektur umfasst die folgenden Elemente:

- Buildumgebung: Eine vorhersehbare, bekannte Buildumgebung, die alle Elemente enthält, die für den Aufbau der Deploymentumgebung benötigt wird. Dies umfasst Folgendes:

- JDKs
- SDKs
- Code-Repository (z. B. CVS)
- Bibliotheken

Die erforderlichen Schritte von der Programmierung bis zur Produktion können variieren und iterativ sein. Die Buildumgebung muss jedoch alle Komponenten enthalten, die bei Bedarf für die Wiederherstellung Ihrer Deploymentumgebung in einer vorhersehbaren Weise benötigt werden.

- Qualitätssicherungsbereich: Eine separate Umgebung für die Durchführung von Aufgaben zur Qualitätssicherung. Die Qualitätssicherung ist die erste

Umgebung, in die die Arbeit (möglicherweise) verschiedener Entwickler integriert und als eine Einheit behandelt wird.

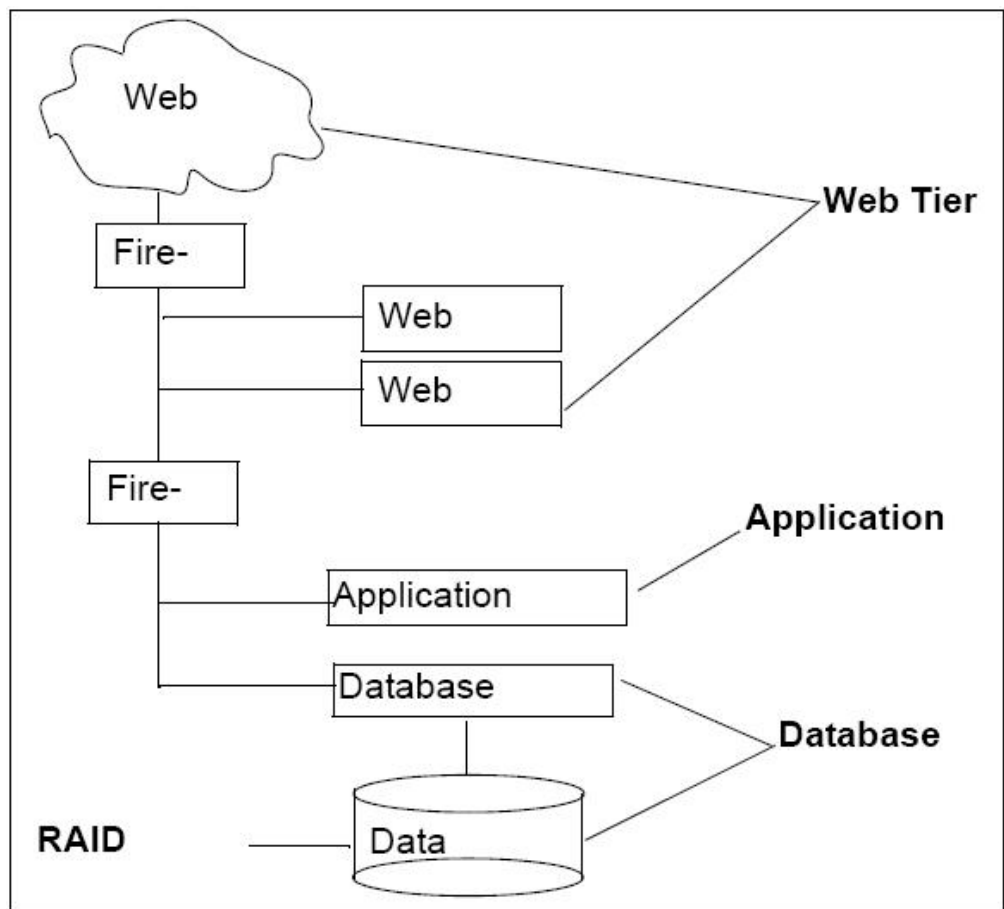
- Bereitstellungsbereich: Eine separate Umgebung, in der die in den Qualitätssicherungsbereich integrierte Arbeit jetzt in einem speziellen Kontext ausgeführt wird, in dem die Produktionsphase emuliert wird.

Deploymentinfrastruktur

Eine generelle Strategie für die Einrichtung einer leistungsfähigen Infrastruktur ist, alles doppelt anzulegen, d. h., redundante Systeme zu implementieren, sodass beim Ausfall eines primären Systems ein sekundäres System so schnell wie möglich online geschaltet wird. Die folgende Abbildung zeigt eine typische Infrastruktur mit drei Schichten:

- Webschicht: Alle Komponenten, die Anforderungen von Web-Browsern bearbeiten und Inhalte für Web-Browser bereitstellen.
- Anwendungsschicht: Alle Komponenten, die Anforderungen von der Webschicht bearbeiten und dynamische Inhalte für die Webschicht bereitstellen, in der Regel Daten aus der Datenbankschicht.
- Datenbankschicht: Alle Komponenten, die Daten für die Anwendungsschicht bereitstellen.

Das folgende Diagramm zeigt eine typische Deploymentinfrastruktur:



Zwei Web-Server stellen sicher, dass beim Ausfall eines Servers der andere die Funktionen des ausgefallenen Servers übernimmt. Durch eine zweite Firewall wird

der Datenschutz verstärkt und zudem sichergestellt, dass bestimmte gesetzliche Bestimmungen zur Datensicherung eingehalten werden.

Eine Strategie für den physischen Schutz Ihrer Daten ist die Speicherung der Produktionsdaten in einer RAID-Einheit. Schlägt ein einzelnes Laufwerk fehl, gehen keine Daten verloren. Bei einer solchen Strategie gibt es jedoch einen mechanischen Schwellenwert: Abhängig von Ihrer Konfiguration kann es vorkommen, dass es beim Überschreiten des Schwellenwerts für die Anzahl der fehlgeschlagenen Laufwerke zu einem Datenverlust kommt. Dies müssen Sie bei der Definition Ihrer Anforderungen berücksichtigen.

Sicherungsstrategien

Es gibt verschiedene Sicherungsstrategien für den Datenschutz und für die Wiederherstellung Ihrer Anwendungsserver und Web-Server. Die meisten Sicherungsstrategien beschränken sich auf das Erstellen von Sicherungskopien der aktuellen Daten in der Anwendung und den Webschichten.

Strategie zur Datenbankwiederherstellung

Ihre Sicherungsstrategie gibt vor, wie schnell Sie Daten nach einer Katastrophe wiederherstellen können. Legen Sie deshalb einen für Sie akzeptablen Zeitplan für die Wiederherstellung Ihrer Datenbank fest. Berücksichtigen Sie dabei die Zeit, die Sie brauchen, um das Betriebssystem wiederherzustellen und die Datenbank bei Bedarf erneut zu laden. Planen Sie auch Ihre Sicherungsrichtlinien entsprechend.

Nachfolgend sind die verschiedenen Datenbanksicherungen aufgeführt, die Sie ausführen sollten:

- **Prüfpunktsicherungen:** Datenbankserver protokollieren Aktivitäten auf Transaktionsebene bei der Festschreibung von Transaktionen. Bei Prüfpunktsicherungen wird ein Protokoll der Transaktionen seit der letzten Prüfpunktsicherung geschrieben. Das Prüfpunktprotokoll sollte auf eine separate physische Einheit geschrieben werden. Dabei wird eine Momentaufnahme (Snapshot) der Datenbankaktivität auf Transaktionsebene erstellt. Schlägt die Datenbank fehl, steht eine Datensatzkette zur Verfügung, über die die Aktivität wiederhergestellt werden kann.
- **Das Intervall, mit dem in Ihrer Deploymentumgebung Prüfpunktsicherungen ausgeführt werden, ist eine Geschäftsentscheidung.** Wenn Ihre Site in jeder Stunde Millionen von US-Dollar Umsatz generiert, empfiehlt es sich, Prüfpunktsicherungen mehrmals pro Stunde vorzunehmen. Werden auf Ihrer Site pro Stunde nur wenige Transaktionen verarbeitet, müssen Prüfpunktsicherungen nicht in dieser Häufigkeit ausgeführt werden. Legen Sie ein Intervall fest, bei dem die Datenwiederherstellung an Ihre Geschäftsaktivitäten angepasst ist.
- **Tägliche inkrementelle Sicherungen:** Bei inkrementellen Sicherungen werden nur die Dateien gespeichert, die sich jeden Tag ändern. Die Durchführung täglicher inkrementeller Sicherungen auf einer anderen Einrichtung statt auf physischen Medien ist eine gute Strategie. Die Sicherung ist dann eine Sicherungskopie von Platte auf Platte.
- **Wöchentliche Gesamtsicherungen:** Bei Gesamtsicherungen wird die gesamte Datenbank und nicht nur die Dateien gespeichert, die sich seit der letzten Sicherung geändert haben. Die Durchführung von Gesamtsicherungen auf einer anderen Einrichtung ist eine gute Strategie.

Wiederherstellung der Datenbank

Informationen zu diesem Vorgang

Eine typische Datenbankwiederherstellung umfasst die folgenden Schritte:

Vorgehensweise

1. Wiederherstellung der ursprünglichen Datenbank seit der letzten Datensicherung.
2. Wiederherstellung der Daten in der Datumsreihenfolge von den täglichen inkrementellen Sicherungen.
3. Wiederherstellung der Aktivitäten der letzten Stunden mithilfe von Prüfpunktsicherungen.

Wiederherstellung von Anwendungsserver und Web-Server

Bei der Planung der Wiederherstellungsrichtlinien für Ihre Anwendung und Ihre Web-Server müssen Sie mit einplanen, genaue Spiegelbilder erstellen zu können, wenn Ihre Anwendung oder Ihre Web-Server fehlschlagen. Dies bezeichnet man auch als das "Build Another One And Go"-Prinzip.


Stellen Sie sicher, dass Sie Kopien von allen Komponenten Ihrer Anwendung und Ihrer Web-Server haben: Konfigurationsdateien, Eigenschaftendateien, JVM, ursprünglicher Quellcode aus Ihrem CVS-Repository usw. Sichern Sie alle statischen Daten auf der Webschicht, z. B. benutzerdefinierte JSP-Seiten. Stellen Sie sicher, dass Ihr Sicherungsprozess auch Web-Server- und Container-Konfigurationsdateien oder andere Dateien umfasst, die Sie für den Betrieb Ihrer Site brauchen, die jedoch nicht als Quellcode betrachtet werden und daher nicht im Quellcode-Repository enthalten sind.

Verwenden Sie Ihre Qualitätssicherungs- und Bereitstellungsumgebung als Ausgangspunkte für die Wiederherstellung Ihrer Server und der Betriebsbereitschaft.

Kapitel 2. Symbole in der Benutzerschnittstelle

In der folgenden Tabelle werden die in Visual Modeler verwendeten Symbole beschrieben.

Tabelle 1. Tabelle mit Symbolen

| Symbol | Beschreibung |
|---|---|
|  | Modell bearbeiten - Über dieses Symbol wird ein Modell geändert. |

Kapitel 3. Artikelmodellierung

Absolute und relative Pfade für die Artikelmodellierung verwenden

In diesem Abschnitt wird die Verwendung von absoluten oder relativen Pfaden beschrieben, um Entitäten wie Eigenschaften und Regeln anzugeben. Die Pfade haben folgendes Format:

`<stammknoten_zur_modellgruppe>.<pfad_zum_optionsartikel_mit_der_eigenschaft_oder_regel>.<eigenschaft_oder_regelname>`

Beispiel für einen absoluten Pfad zur Eigenschaft "memoryProvided":

`MXDS-7500.memory.sim256.memoryProvided`

Der Stammknoten der Modellgruppe lautet "MXDS-7500", der Pfad zum Optionsartikel mit der Eigenschaft "memoryProvided" lautet "memory.sim256" und der Eigenschaftsname lautet "memoryProvided".

Wenn Sie eine Eigenschaft oder Regel in mehreren Modellen verwenden wollen, können Sie über spezielle Symbole relative Pfade angeben. Beispiel: "*" im folgenden Pfad gibt an, dass der Pfad im Stammverzeichnis der Modellgruppenhierarchie beginnt.

`*.memory.sim256.memoryProvided`

Beginnt der Pfad mit einem Punkt (.), gibt dies Folgendes an: "Vom Anschlusspunkt der Regel". Beispiel: "." im folgenden Pfad gibt Folgendes an: ein Optionsartikel namens "sim256" in einer Optionsklasse namens "memory" im aktuellen Modell.

`.memory.sim256`

Artikelmodellierung planen

Ein Modell stellt einen konfigurierbaren Artikel dar. Wenn Sie eine IBM Sterling Web-Implementierung planen, überlegen Sie zunächst, wie ein Modell des Artikels entworfen werden kann. Den einzigen richtigen Weg, einen bestimmten Artikel zu modellieren, gibt es nicht. Andererseits gibt es unzählige Möglichkeiten, Modelle zu erstellen, die, obwohl technisch richtig, ineffizient und schwer zu verwalten sind.

Sie sollten deshalb die folgenden Faktoren beachten.

- **Kostenfaktoren: Erstellung, Verwaltung und Leistung.**
Sie müssen die Kosten für die Erstellung des Modells mit den Kosten für dessen Verwaltung und dessen erwarteter Leistung abgleichen. Die Kosten für die Modellerstellung sind Einmalkosten für die Entwicklung des Modells. Die Kosten für die Modellverwaltung sind die laufenden Kosten für die Verwaltung und Verbesserung des Modells. Die Leistung steht für die Ausführungsgeschwindigkeit des Modells auf einer bestimmten Hardwareplattform. Sie können jeden dieser Faktoren optimieren, sollten aber bedenken, dass Sie bei der Optimierung mehrerer dieser Faktoren in den meisten Fällen konkurrierende Zielsetzungen verfolgen. Ein komplexes Modell

kann in der Ausführung zwar schnell, aber schwer zu verwalten sein. Einen Überblick über die Modelle und die besten Entwurfsprinzipien erhalten Sie unter „Betrachtungen zur Entwurfsphase der Artikelmodellierung“.

- Rollen des Implementierers: reiner Modellersteller, Modellersteller und -verwalter, reiner Modellverwalter.

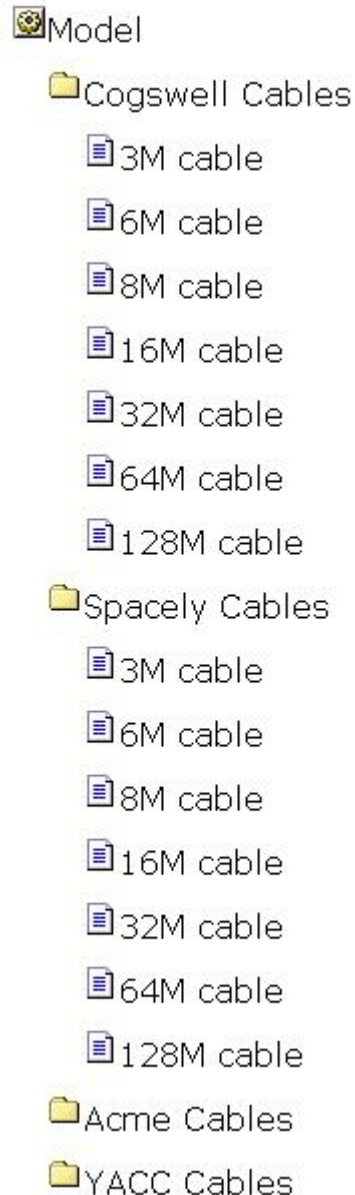
Die Rolle des Implementierers wirkt sich auf dessen Modellvorauswahl aus. Ein Berater, der einen Monat Zeit hat, um ein Modell zu implementieren, das dann an eine andere Gruppe zur Verwaltung weitergeleitet wird, wird sich auf den schnellen Entwurf des Modells konzentrieren und weniger auf den Entwurf eines Modells, das einfach zu verwalten ist. Wenn der Implementierer das Modell auch verwaltet, dauert der Entwurf des Modells wahrscheinlich länger und die Ausführung des Modells ist langsamer. Dafür ist dann aber auf lange Sicht gesehen die Verwaltung einfacher. Unabhängig von der Rolle muss Ihr Ziel sein, ein Modell aufzubauen, das im weiteren Verlauf keine Probleme verursacht.

Betrachtungen zur Entwurfsphase der Artikelmodellierung

Beim Entwurf eines Artikelmodells ist es wichtig, die vorgegebenen Entwurfsprinzipien zu beachten, damit ein komplexes Modell auch effizient in der Ausführung ist.

Dimensionierung eines Artikelmodells

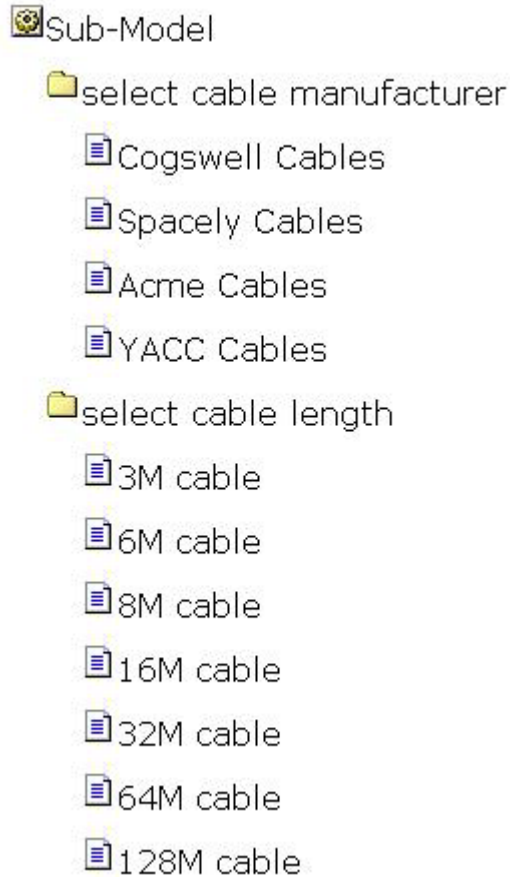
Die Modellgröße spielt beispielsweise eine Rolle. Umfangreiche Modelle dauern länger bei der Wiedergabe im Browser und sind schwieriger zu verwalten. Zudem muss der Konfigurator bei der Konfiguration mehrmals die Modellstruktur durchsuchen, um Preise, Auslöseregeln usw. zu ermitteln. Wird die Modellgröße durch Unterbaugruppen und andere in diesem Abschnitt genannte Verfahren möglichst klein gehalten, wirkt sich dies positiv auf die Leistung und die Verwaltungskosten aus. Angenommen, verschiebene Kabelanbieter haben jeweils verschiedene Kabellängen im Angebot. Sie wollen dem Benutzer die Möglichkeit geben, die Menge, Länge und den Kabelanbieter auszuwählen. Eine Möglichkeit, dies umzusetzen, besteht darin, Optionsartikel in Ihrem Modell zu erstellen, die jede der verfügbaren Optionen darstellen (siehe folgende Abbildung):



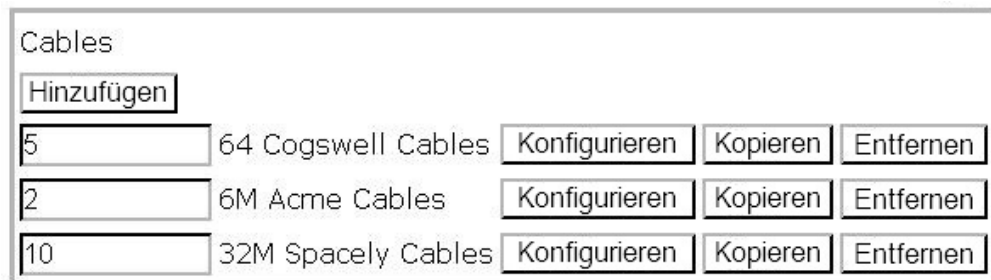
Dieser Ansatz funktioniert, auch wenn es etwas aufwendig ist, ein Modell mit zahlreichen Optionen, die für den Endbenutzer vielleicht nie von Interesse sein werden, zu erstellen, zu implementieren und zu verwalten.

Ein alternativer Ansatz wäre die Implementierung der Optionsartikel mit den verschiedenen Kabellängen als Optionsartikelgruppe und die herstellereinspezifische Einbindung dieser Gruppe. Dies würde die Verwaltung vereinfachen: Der Modellierer muss nur an einem Ort suchen, um die Optionsartikeldaten zu den Kabeln zu aktualisieren. Bei diesem Ansatz erhält der Endbenutzer jedoch eine sehr umfangreiche Kabelliste, wodurch die Leistung nicht verbessert wird, da auch hier das zu durchsuchende Modell sehr komplex ist.

Eine bessere Alternative ist das Erstellen eines Untermodells, über das der Benutzer den Kabelhersteller und die Kabellänge auswählen kann. Mithilfe der dynamischen Instanziierung kann der Benutzer dann so viele unterschiedliche Kabeltypen und -längen hinzufügen, wie er braucht (siehe folgende Abbildung):



Die folgende Abbildung zeigt eine Musterschnittstelle für die Kabelauswahl, bei der der Endbenutzer über die dynamische Instanziierung seine Kabelauswahl konfigurieren kann.



Wie Sie sehen können, wird bei diesem Ansatz die Modellgröße klein gehalten. Die Verwaltung dieses Modells ist einfacher, da der Modellierer keine große Anzahl an doppelten Optionsartikeln mehr handhaben muss. Zudem ist das Leistungsverhalten besser, da das Modell kleiner ist. Auch die Konfiguration gestaltet sich für den Endbenutzer einfacher, da keine langen Kabel- und Herstellerlisten mehr zu durchsuchen sind, um die richtige Auswahl zu treffen.

Artikelmenge über Popup-Steuerelemente angeben

Es gibt Fälle, in denen der Modellierer dem Benutzer ermöglichen will, einen Artikel auszuwählen und dann die Anzahl der gewünschten Artikel einzugeben. Die beste Lösung hierfür ist die Festlegung der Optionsklassenanzeige auf "Popup

für Menge (Mge)". Wenn der Endbenutzer einen Artikel auswählt, wird ein Mengenfeld angezeigt, in das der Endbenutzer die Anzahl der gewünschten Artikel eingeben kann.

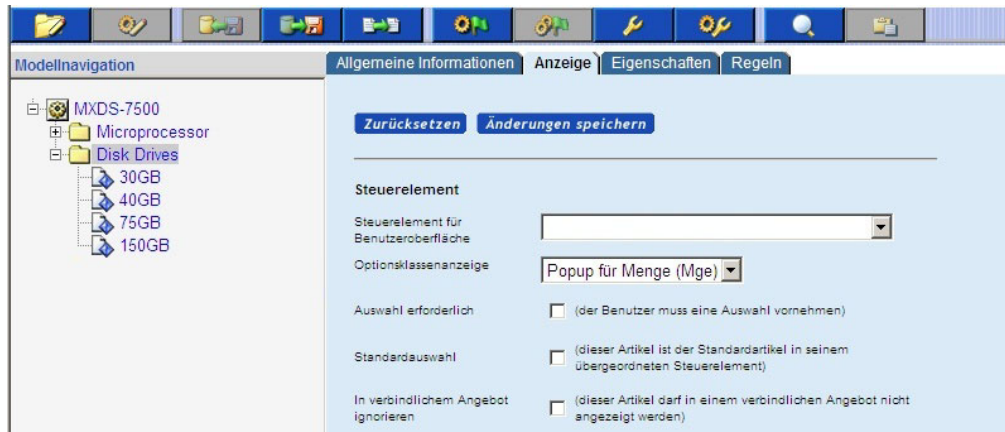
Modellierer, denen solche Mengenfelder nicht gefallen, können stattdessen UEV-Steuerelemente (User Entered Values) verwenden, um ein Bearbeitungsfeld neben dem Artikel anzuzeigen, in das Benutzer eine Menge eingeben können. Das Problem dabei ist, dass sich das Verhalten des Steuerelements "Popup für Menge (Mge)" deutlich vom Verhalten der UEV-Steuerelemente unterscheidet: Das Steuerelement "Popup für Menge (Mge)" verfügt über eine integrierte Mengenverarbeitungsfunktion, wohingegen bei den UEV-Steuerelementen weitere Arbeiten erforderlich sind.

Wenn ein Endbenutzer in das Feld "Popup für Menge (Mge)" eine Menge eingibt, wählt die Anwendung automatisch die Menge des ausgewählten Artikels aus. Alle mit den Artikeln verbundenen Eigenschaften werden im Konfiguratorstatus (Eigenschaftenpool) berücksichtigt und die Werte der numerischen Eigenschaften werden mit der eingegebenen Menge multipliziert.

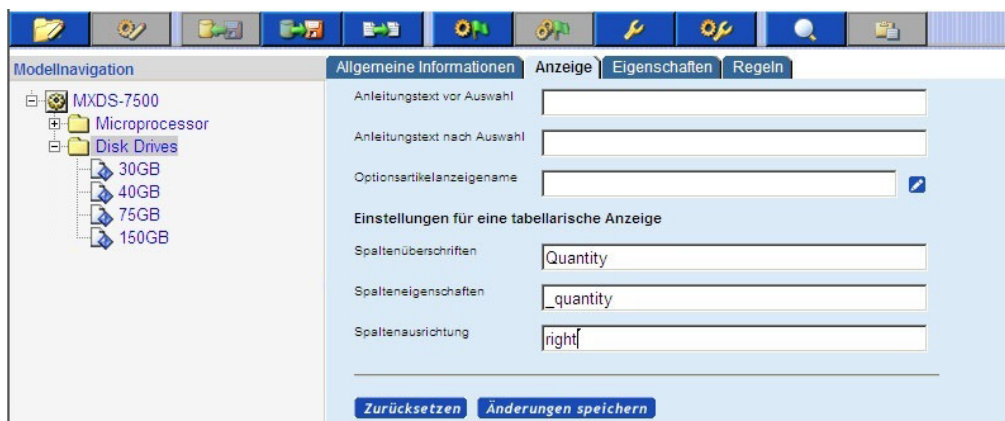
Wenn in das UEV-Steuerelement ein Wert eingegeben wird, passiert zunächst nichts. UEV-Steuerelemente dienen lediglich dazu, zusätzliche Informationen vom Benutzer zu erfassen. Um zu erreichen, dass der vom Benutzer eingegebene Wert als Menge interpretiert wird, muss der Modellierer eine Erweiterungsregel schreiben, durch die auf Basis des in das UEV-Steuerelement eingegebenen Werts die Anzahl des ausgewählten Artikels ausgewählt wird. Wenn Sie versuchen, anhand des in das UEV-Steuerelement eingegebenen Werts die Eigenschaft "_quantity" mithilfe einer Zuordnungsregel festzulegen, wird dies nicht wie erwartet funktionieren, da nicht automatisch Instanzen der Eigenschaften des Artikels im Eigenschaftenpool erstellt werden.

Wenn Sie ein Feld "Popup für Menge (Mge)" neben dem ausgewählten Artikel anzeigen wollen, müssen Sie die Optionsklassenanzeige "Popup für Menge (Mge)" und eine der tabellarischen Anzeigen mit Mengensteuerelementen verwenden. Dadurch wird sichergestellt, dass die richtige Anzahl an Artikeln ausgewählt wird und die richtigen Eigenschaften in den Eigenschaftenpool kopiert werden.

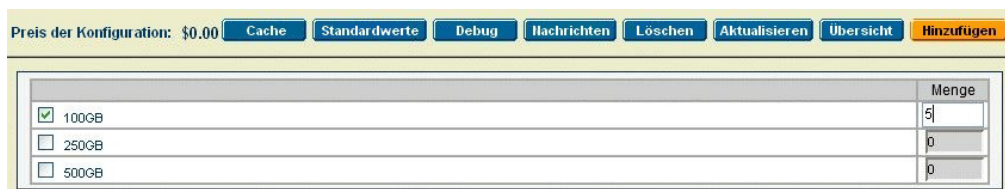
Beispiel: Die folgende Abbildung zeigt, wie ein Steuerelement "Popup für Menge (Mge)" mithilfe von Visual Modeler definiert wird. Wählen Sie in der Anzeige **Modelle und Gruppen** das Modell aus, das Sie ändern wollen, und klicken Sie dann auf das Symbol **Modell bearbeiten**. Die Anzeige "Modellnavigation" erscheint. Klicken Sie auf die Optionsgruppe, die Sie ändern wollen, klicken Sie dann auf die Registerkarte **Anzeige** und wählen Sie "Multi-select Tabular Display" für Tabellenanzeige mit Mehrfachauswahl aus der Dropdown-Liste **Steuerelement für Benutzerschnittstelle** aus (siehe folgende Abbildung).



Blättern Sie zum Ende der Seite und geben Sie die Einstellungen für "Spaltenüberschriften", "Spalteneigenschaften" und "Spaltenausrichtung" ein. Die folgende Abbildung zeigt Beispiele für diese Einstellungen:



Jetzt können Sie das Modell kompilieren und testen. Wie Sie sehen können, wurde ein Steuerelement "Popup für Menge (Mge)" wie von Ihnen in der Produktkonfiguratorseite angegeben platziert.




Modelleigenschaften definieren

Eigenschaften sind im IBM Sterling Configurator allgegenwärtig. Modellierer hängen Eigenschaften an Modelle, Optionsklassen und Optionsartikel an und schreiben dann Regeln für diese Eigenschaften, um Nachrichten anzuzeigen, Artikel auszubelenden oder anzuzeigen und sogar die Werte von anderen Eigenschaften festzulegen. Zieht man die Bedeutung von Eigenschaften bei der Modellierung eines konfigurierbaren Artikels in Betracht, muss besonders darauf geachtet werden, wie Eigenschaften definiert und verwendet werden. In diesem Abschnitt werden einige hilfreiche Tipps und Verfahren genannt, die beim Definieren und Anhängen von Eigenschaften beachtet werden sollten.


Modelleigenschaftsnamen definieren

Bei der Entwicklung eines Modells, insbesondere bei engen Zeitvorgaben, kommt der Modellierer in die Versuchung, einen kürzeren, zeitsparenderen Weg zu wählen, um den Entwicklungsprozess zu beschleunigen. Eine der gängigsten Varianten hierbei ist die Erstellung von Eigenschaften mit kurzen und häufig ungenauen oder kryptischen Namen. Dies mag zwar die Entwicklung des Modells auf kurze Sicht beschleunigen, im Gegensatz dazu aber den erforderlichen Aufwand für die Verwaltung des Modells extrem erhöhen. Der Modellierer muss seine Modelle immer so entwerfen, dass sofort offensichtlich wird, wofür eine bestimmte Eigenschaft steht. Je aussagekräftiger der Name für eine Eigenschaft ist, desto einfacher wird es sein, das Modell zu testen und zu verwalten – jetzt und auch in der Zukunft.

Das folgende Beispiel soll dies verdeutlichen:

| model node | properties |
|---|---|
|  Model | tmr=sum(mr) tmp=sum(mp) amr=value(tmr)-value(tmp) |
| OC1 | oi1 mr=1 oi2 mr=2 |
| OC2 | oi3 mp=2 oi4 mp=1 |

Auf den ersten Blick ist es nicht offensichtlich, welche Aufgabe die diesem Modell zugeordneten Eigenschaften erfüllen sollen. Mit etwas mehr Zeitaufwand bei der Vergabe aussagekräftigerer Namen ist es wesentlich einfacher, den Kern dieser Eigenschaften und deren Beziehung zueinander zu erfassen.

| model node | properties |
|---|---|
|  Model | total_mem_required=sum(mem_required) total_mem_provided=sum(mem_provided) additional_mem_required=value(total_mem_required) - value(total_mem_provided) |
| OC1 | oi1 mem_required=1 oi2 mem_required=2 |
| OC2 | oi3 mem_provided=2 oi4 mem_provided=1 |

Eindeutige Modelleigenschaften definieren

Es kommt nicht selten vor, dass ein Modellierer unter dem Zeitdruck, eine bestimmte Funktion implementieren zu müssen, eine vorhandene Eigenschaft wiederverwendet, statt eine neue, speziell auf das bestehende Problem zugeschnittene Eigenschaft zu erstellen. Dies kann zwei mögliche Auswirkungen haben:

- Das Modell kann schwerer zu verstehen sein, wenn aus dem Namen der vorhandenen Eigenschaft kein Bezug zum bestehenden Problem abgeleitet werden kann.
- Die Wiederverwendung des Eigenschaftsnamens kann Fehler im Modell verursachen, wenn es durch die Wiederverwendung zu Konflikten mit der ursprünglichen Verwendung der Eigenschaft kommt.

Das Beispiel aus dem vorherigen Abschnitt soll dies verdeutlichen. Angenommen, der Modellierer hat eine Eigenschaft für erforderliche Speicherkapazität erstellt und dieser den Namen "memory" gegeben (der vorherige Abschnitt zeigt, warum dies eine unglückliche Auswahl als Einstiegspunkt war). Wenn er nun feststellt, dass er eine Eigenschaft für bereitgestellte Speicherkapazität benötigt, erkennt er, dass er bereits eine Eigenschaft namens "memory" hat, und entscheidet sich, diese Eigenschaft zu verwenden, anstatt eine neue Eigenschaft zu erstellen.

Auf den ersten Blick sieht es so aus, als würden alle Optionsartikel eine bestimmte Speichermenge benötigen. Stattdessen benötigen aber zwei Artikel Speicherkapazität und zwei andere stellen Speicherkapazität bereit. Diese Problematik wird noch komplizierter: Die Eigenschaft "total_memory_required" hat nicht mehr den richtigen Wert, da jetzt eine Summenoperation für erforderliche und bereitgestellte Speicherkapazität durchgeführt wird. Bei einer Modellierung dieser Art muss der Modellierer zusätzliche Zeit aufwenden, um die jeweiligen Instanzen der benötigten Eigenschaften voneinander zu trennen: So muss er z. B. vollständige oder relative Pfade zu den Artikeln verwenden, die die geeigneten Eigenschaftsinstanzen enthalten.

Hierarchie mit Modelleigenschaften definieren

Eigenschaften können auf jeder Ebene in der Modellgruppenhierarchie definiert werden – von der Stammmodellgruppenebene bis zur Ebene eines einzelnen Modells. Basierend auf der Position der definierten Eigenschaft wird festgelegt, welche Modelle die Eigenschaft sehen und nutzen können. Eine etwas gründlichere Überlegung beim Entwurf Ihrer Modelle trägt dazu bei, die Modellentwicklung zu beschleunigen und Unklarheiten im Zusammenhang mit Eigenschaften zu vermeiden. Anhand der folgenden Richtlinien können Sie festlegen, wo eine Eigenschaft definiert werden sollte:

Wenn eine Eigenschaft nur in einem bestimmten Modell verwendet wird, können Sie das Modell auf Modellebene definieren.

Wenn eine Eigenschaft in mehreren Modellen innerhalb einer bestimmten Modellgruppe verwendet wird, müssen Sie die Eigenschaft auf Modellgruppenebene definieren.

Wird eine Eigenschaft in Modellen verwendet, die mehrere Modellgruppen umfassen, müssen Sie die Eigenschaft in der ersten Modellgruppe definieren, die alle Modellgruppen enthält, deren Modelle die Eigenschaft verwenden.

Als generelle Maßnahme können Sie die Eigenschaft auch auf der Ebene des Stammmodellgruppenknotens definieren.

Mehrere Eigenschaften mit identischem Wert definieren

Mehrere Eigenschaften mit demselben Wert können in manchen Fällen dazu beitragen, dass sich ein Modell leichter erstellen und verwalten lässt. Dieses Konzept mag auf den ersten Blick verwirrend erscheinen und lässt sich daher am besten anhand eines Beispiels verdeutlichen. Angenommen, Sie erstellen ein

Modell, mit dem der Benutzer aus verschiedenen Platteneinheiten auswählen kann. Jedem Platteneinheitentyp sind verschiedene Platten zugeordnet. Der Benutzer kann also mehrere Platteneinheiten eines beliebigen Typs auswählen. Eine Information, die Sie berechnen müssen, ist die Gesamtzahl der Platten, die der Benutzer ausgewählt hat (siehe unten).

| model node | properties |
|--------------------|------------------------|
| DiskArray SubModel | total_disks=sum(disks) |
| 100GB disk arrays | |
| blue | disks=4 |
| mauve | disks=6 |
| 250GB disk arrays | |
| blue | disks=8 |
| mauve | disks=12 |

Nehmen wir jetzt an, dass Sie gerade feststellen, dass Sie auch die Anzahl der 100-GB-Platteneinheiten und der 250-GB-Platteneinheiten kennen müssen. Statt diese Werte durch Angabe von Artikelpfaden zu den Eigenschaften, durch Schreiben von Regeln, die an einem bestimmten Punkt im Modell angehängt werden müssen, oder durch Überarbeiten aller "disk"- und "total_disk"-Eigenschaften zu berechnen, kann dies auch durch die einfache Definition einiger neuer Eigenschaften erfolgen, die dieselben Werte aufweisen wie die alte "disk"-Eigenschaft (siehe unten).

| model node | properties |
|--------------------|--|
| DiskArray SubModel | total_disks=sum(disks) total_100GBdisks = sum(100GB_disks) total_250GBdisks = sum(250GB_disks) |
| 100GB disk arrays | |
| blue | disks=4 100GB_disks=4 |
| mauve | disks=6 100GB_disks=6 |
| 250GB disk arrays | |
| blue | disks=8 250GB_disks=8 |
| mauve | disks=12 250GB_disks=12 |

Wenn also die Gesamtzahl der Platten ermittelt werden soll, kann nach wie vor "sum(disks)" verwendet werden. Sollen die einzelnen Werte berechnet werden, kann auch dies problemlos erfolgen – jeweils ohne Pfade oder einzelne Eigenschaften angeben oder bereits durchgeführte Arbeiten erneut ändern zu müssen.

Arbeitsblätter für die Eigenschaftszuweisung verwenden

Bei der Entwicklung eines Modells muss mehreren Optionsklassen oder Optionsartikeln häufig dieselbe Gruppe von Eigenschaften zugewiesen werden.

Arbeitsblätter sind in diesem Fall sehr hilfreich, da sie die schnelle Festlegung der Werte für eine bestimmte Eigenschaft für Optionsklassen oder Optionsartikel erlauben. Dies gilt insbesondere dann, wenn der Wert einer Eigenschaft an mehreren Stellen mit einer Formel festgelegt werden soll. Der Modellierer kann dabei die Formel einfach kopieren und in alle gewünschten Artikel einfügen. Ein Beispiel hierfür sehen Sie unten. Die Anzeigeeigenschaften werden für jeden Artikel in einer tabellarischen Anzeige festgelegt. Mithilfe eines Arbeitsblatts lassen sich die Formeln ohne großen Aufwand für Spalte 1 (col1) und Spalte 2 (col2) ausschneiden und für jeden Artikel in der Optionsklasse einfügen.

| Artikel | col1 | col2 |
|---------|---|---|
| 30GB | \$(expand("min_array_disk"))/\$(expand("max_array_...)) | \$(expand("min_array_disk"))/\$(expand("max_array_...)) |
| 40GB | \$(expand("min_array_disk"))/\$(expand("max_array_...)) | \$(expand("min_array_disk"))/\$(expand("max_array_...)) |
| 75GB | \$(expand("min_array_disk"))/\$(expand("max_array_...)) | \$(expand("min_array_disk"))/\$(expand("max_array_...)) |

Ein weiterer Vorteil von Arbeitsblättern zeigt sich darin, dass ein präzises Bild eines Abschnitts des Modells zur Verfügung steht. Mit etwas durchdachter Planung kann ein Arbeitsblatt einen guten Überblick zu einem bestimmten Abschnitt des Modells oder eine vollständige Darstellung der Lösung zu einem bestimmten Problem bieten. Nachfolgend ist eine andere Ansicht derselben Optionsklasse dargestellt. In diesem Fall sollen alle Eigenschaften für Minimum (min) und Maximum (max) angezeigt werden, die für jeden Optionsartikel festgelegt wurden.

| Artikel | min array disk | max array disk | min cache memory | max cache memory |
|---------|----------------|----------------|------------------|-------------------|
| 30GB | 4 | 252 | 4 | 64 |
| 40GB | 64 | 508 | 6 | =value(exp_cache) |
| 75GB | 128 | 1148 | 6 | 128 |

Verkettungen in Eigenschaftsformeln vermeiden

Eigenschaften, die an einen Artikel angehängt sind, haben keine bestimmte Reihenfolge. Das bedeutet Folgendes: Wenn für die Festlegung von Eigenschaftswerten Formeln verwendet werden, kann man sich nicht auf eine bestimmte Reihenfolge bei der Auswertung der Formeln verlassen. Enthält eine Eigenschaft A eine Formel und Eigenschaft B eine Formel, die auf Eigenschaft A aufbaut, ist nicht sichergestellt, dass die aus Formel B erstellte Regel nach der für Formel A erstellten Regel ausgelöst wird. Um dieses Problem zu umgehen, hat der Modellierer zwei Wahlmöglichkeiten:

- Er wandelt die erste Formel in eine Regel um, die vor der Auswertung der zweiten Formel ausgelöst wird. Alle aus Formeln generierten Regeln haben die Priorität 50. Durch die Erstellung einer Regel für die erste Formel und die Einstellung der Priorität auf einen Wert kleiner als 50 soll sichergestellt werden, dass der Wert für Eigenschaft A festgelegt wird, bevor der Wert für Eigenschaft B berechnet wird.
- Er aktiviert die Funktion zum Auslösen einer Wiederholungsregel. In diesem Fall wird beim ersten Durchgang der Regelauslösung der Wert für Eigenschaft A berechnet. Im zweiten Durchgang der Regelauslösungsschleife wird der Wert für Eigenschaft B auf Basis des im ersten Durchgang berechneten Werts für Eigenschaft A berechnet. Hinweis: Eine große Anzahl an Formelverkettungen dieser Art kann zu einer Leistungsverschlechterung aufgrund der Anzahl der

Durchläufe in der Regelauslösungsschleife führen, die erforderlich sind, um alle Bedingungen zu erfüllen. Aus diesem Grund wird die erste Alternative empfohlen. Zudem sollte darauf geachtet werden, dass Formelverkettungen auf das unvermeidliche Minimum reduziert werden.

Regeln definieren

Regelbedingungen definieren

Regelbedingungen werden erstellt, indem boolesche Operationen auf Vergleichsausdrücke angewendet werden. Ein Vergleichsausdruck ist der Vergleich eines Funktions-/Eigenschaftspaares mit einem anderen Funktions-/Eigenschaftspaar mithilfe von Vergleichsoperationen wie kleiner als, gleich, größer als, in, nicht in usw. Das Ergebnis ist entweder wahr (true) oder falsch (false). Boolesche Operatoren wie AND und OR schließen Gruppen dieser Vergleichsausdrücke ein. Die Vergleichsausdrücke werden *Fragmente* genannt, weil Sie Fragmente einer Regel sind. Die linke Seite des Vergleichsoperators wird häufig mit dem Kürzel LHS abgekürzt, RHS wiederum steht für die rechte Seite.

Reihenfolge von Regelfragmenten festlegen

Durch die Auswertung von Regelfragmenten wird festgelegt, wann eine Regel ausgelöst wird. Daher ist die Reihenfolge, in der Fragmente in einer Regel erscheinen, besonders wichtig. Je schneller das Modell ermitteln kann, ob eine Regel wahr (true) oder falsch (false) ist, desto effizienter kann das Modell sein. Hinzu kommt: Je schneller das Modell ermitteln kann, dass eine Regel nicht ausgelöst werden soll, desto eher kann das Modell mit anderen Verarbeitungsschritten fortfahren. Die Festlegung der Reihenfolge von Regelfragmenten, also von der höchsten Wahrscheinlichkeit der Verhinderung der Regelauslösung bis zur geringsten Wahrscheinlichkeit, kann zu einer Leistungsverbesserung beitragen.

Testen Sie Ihre Regeln immer, um sicherzustellen, dass diese nur bei Bedarf ausgelöst werden. Außerdem ist es wichtig zu wissen, unter welchen Umständen die Ergebnisse einer Regel verwendet bzw. nicht verwendet werden. Beispiel: Eine Erweiterungsregel, die immer ausgelöst wird, bei der aber keine Auswahl im Erweiterungsabschnitt (Expansions) erfolgt, wenn die Mengenformel den Wert 0 ergibt, ist nicht besonders effizient. Ebenso ineffizient ist es, wenn keine Übereinstimmungen für die Formel in den Feldern > und <= im Erweiterungsabschnitt vorhanden sind.

Allgemeine Regeln erstellen

Schreiben Sie wenn möglich Regeln, die so allgemein wie möglich gehalten sind. Die folgende Regel kann z. B. an jeden Artikel angehängt werden, an den die Eigenschaften "productType" und "handsetType" angehängt sind:

```
If propval(productType) != value(selectProductType)
and propval(handsetType) != value(phonePreference)
    set _isVisible=0
```

Dieses Regel wird nur für Artikel ausgelöst, bei denen die Eigenschaft "productType" angehängt ist UND nicht mit den ausgewählten Produkttypen übereinstimmt UND wenn die ausgewählten Telefonvoreinstellungen nicht mit den Voreinstellungen des aktuellen Artikels übereinstimmen. Eine allgemeine Regel wie diese kann Dutzende von anderen speziellen Regeln wie die folgenden ersetzen:

```

If propval(productType) == literal("handset")
and propval(handsetType) != literal("camera")
and value(phonePreference) == literal("camera")
  set _isVisible=0
If propval(productType) == literal("handset")
and propval(handsetType) != literal("flip")
and value(phonePreference) == literal("flip")
  set _isVisible=0
...

```

Formeln zum Definieren von Regeln verwenden

In vielen Situationen können Formeln statt Regeln verwendet werden. Bei der Modellierung werden Formeln als angehängte Eigenschaften gehandhabt, die als Wert einen Ausdruck aufweisen, der bei der Laufzeit ausgewertet wird. Wenn eine der Funktionen, auf die im Ausdruck verwiesen wird, nicht ausgewertet werden kann, fungiert die Formel wie eine Regel, die nicht ausgelöst wurde. Wenn die Funktion zum mehrmaligen Auslösen von Regeln aktiviert ist, wird die Formel bei jedem Auslösevorgang erneut ausgewertet, bis die Regelauslösung beendet ist oder bis die Formel zu einem Ergebnis führt.

Verwenden Sie eine Formel statt einer Regel, wenn die einzige Bedingung für die Berechnung eines Ergebnisses ist, dass die in der Formel verwendeten Funktionen/Eigenschaften Werte aufweisen.

Angenommen, Sie wollen beispielsweise den Kurvenradius von Lkw-Teilen wie Achse und Achsabstand berechnen, um sicherzustellen, dass die Auswahl eines Lkw-Teils durch den Benutzer sinnvoll ist. Sie können in diesem Fall an die jeweiligen Lkw-Teile eine Formel anhängen, um "turningRadius" wie folgt zu berechnen:

```

turningRadius = value(axelTurnFactor) * value(wheelBaseTurnFactor)*
sum(turningElements)

```

Diese Formel wird ausgelöst, wenn jeder der Ausdrücke "value(axelTurnFactor)", "value(wheelBaseTurnFactor)" und "sum(turningElements)" numerische Ergebnisse ausgibt.

Die entsprechende Regel sieht wie folgt aus:

```

if (value(axelTurnFactor) >= 0 or value(axelTurnFactor) < 0)
and (value(wheelBaseTurnFactor)>= 0 or value(wheelBaseTurnFactor)< 0)
and (sum(turningElements) >=0 or sum(turningElements) <0)
  turningRadius = value(axelTurnFactor) * value(wheelBaseTurnFactor)*
sum(turningElements)

```

Der Bedingungsteil der Regel ist recht lang und scheint immer auf den Auswertungswert "true" hinauszulaufen. Es besteht jedoch die Möglichkeit, dass Funktionen den Wert null zurückgeben, wenn eine Eigenschaft, auf die sie verweisen, nicht vorhanden ist. Die Regel überprüft also genau, ob das Ergebnis ungleich null ist, indem es prüft, ob ein Rückgabewert ≥ 0 oder < 0 lautet.

Pfadangaben in Regeln vermeiden

Die LHS- und die RHS-Seite eines Regelfragments bestehen aus einer Funktion und einem Eigenschaftsnamen. Der Eigenschaftsname darf sowohl relative als auch absolute Pfadangaben enthalten. Wenn Sie jedoch die Pfadangaben einer Eigenschaft in einem Regelfragment angeben, kann dies dazu führen, dass die Regel nicht mehr funktioniert, wenn sich die Pfadangabe oder die Optionsklassen ändern.

Die folgende Regel verweist beispielsweise durch vollständig angegebene Pfadangaben auf "wheelSize" und "wellSize". Wenn der Modellierer die Optionsklassen "Räder" oder "Radabdeckung" ändern muss oder die Regel für ein anderes Modell verwenden will, funktioniert die Regel möglicherweise nicht mehr richtig.

```
If value(*.wheels.wheelSize) == literal("17in")
and value(*.fender.wellSize) < literal(17)
    set _isVisible=0
```

Verwenden Sie Pfadangaben nur, wenn Sie auf eine bestimmte Instanz einer Eigenschaft zugreifen wollen, und dann auch nur, wenn es nicht möglich ist, einen neuen Eigenschaftstyp zu erstellen, der diesen Wert aufnehmen soll. Wenn Sie auf den Pfadnamen einer Eigenschaft verweisen müssen, ist es häufig besser, relative statt absolute Pfadnamen zu verwenden.

Einschränkungstabellen und Regeln im Vergleich

In diesem Abschnitt wird ein Vergleich zwischen der Verwendung von Einschränkungstabellen und Regeln angestellt, um die Auswahlmöglichkeiten für den Kunden einzuschränken. Einschränkungstabellen schränken die Auswahlmöglichkeiten des Kunden auf einen oder mehrere Optionsartikel basierend auf der Auswahlmöglichkeit des Kunden für einen anderen Optionsartikel ein. Die Auswahl einer Außenfarbe für ein Fahrzeug kann beispielsweise die Wahlmöglichkeiten für die Innenfarbe einschränken.

Einschränkungstabellen eignen sich am besten für einfache Auswertungen, z. B. ob ein Optionsartikel mit einem anderen Optionsartikel funktioniert. Einfache Einschränkungstabellen sind einfacher zu verwalten als Regeln. Große, komplexe Einschränkungstabellen hingegen sind schwieriger zu verwalten und können zu Leistungsproblemen führen.

Einschränkungstabellen werden intern in Regeln umgesetzt.

Regeln eignen sich am besten für komplexe Auswertungsprobleme und sind vielseitiger als Einschränkungstabellen. Während sowohl Einschränkungstabellen als auch Regeln Fehlernachrichten ausgeben können, können Sie auch Regeln erstellen, um Eigenschaften festzulegen oder Auswahlen zu treffen.

Leistungsverhalten von Modellen verbessern

In diesem Abschnitt werden einige der Verfahren beschrieben, die die Modellerstellung und -verwaltung vereinfachen. Die Auswahl des richtigen Verfahrens kann signifikante Auswirkungen auf das Leistungsverhalten eines Modells haben.

- Optionsklassengruppen, Optionsartikelgruppe und Unterbaugruppen:

Dieses Verfahren eignet sich sehr gut, wenn eine Gruppe von Optionen in vielen verschiedenen Modellen wiederholt wird.

Nehmen Sie beispielsweise an, dass jeder Computer, den Sie verkaufen, eine Liste mit Festplattenlaufwerken enthält, aus denen der Benutzer auswählen kann. Durch das Erstellen von Optionsklassengruppen, Optionsartikelgruppen und Unterbaugruppen kann der Modellierer allgemeine Informationen an einer Position erstellen und diese dann an verschiedenen Positionen wiederverwenden.

Ein Nachteil dabei ist jedoch, dass dieses Verfahren zu Überlagerungen bei großen Modellen führen kann, wenn eine Unterbaugruppe mehrmals im selben Modell enthalten ist.

- Einfügen/Entfernen von Untermodellen:
Dieses Verfahren ist hilfreich, wenn eine Konfiguration eine Auswahl enthält, die auch konfigurierbar ist. Sie können das Einfügen/Entfernen von Untermodellen verwenden, um komplexe Konfigurationen innerhalb eines größeren Verkaufsmodells zu verschachteln.
Ein Nachteil hierbei ist, dass alle Kopien des konfigurierten Artikels dieselbe Konfiguration aufweisen.
- Dynamische Instanziierung:
Dieses Verfahren ermöglicht mehrere Instanzen eines konfigurierten Artikels in einem einzelnen Modell. Jede Instanz kann eine andere Konfiguration haben.

Modellierungstools

Die Modellierung kann eine zeitaufwendige und mühsame Arbeit sein. Letztendlich sind die Richtigkeit der Modellierung und die Skalierbarkeit der erstellten Lösung zentrale Faktoren für den Erfolg des Projekts. Als Unterstützung bei der Erstellung skalierbarer und richtiger Modelle stehen eine Reihe von Tools zur Verfügung, die in verschiedenen Phasen der Entwicklung dem Modellierer weiterhelfen. Während der Entwicklung kann das Tool für Traceprotokoll- und Modellberichterstellung dem Modellierer helfen festzustellen, welche Modelle mit dem Debugger getestet werden müssen. Bevor Modelle in den Produktionsprozess einfließen, kann deren Skalierbarkeit und Leistungsfähigkeit mithilfe der Belastungstestplattform überprüft werden. In der Ausführungsphase kann die Seite zum Modellcachestatus Einblicke in die Systemnutzung des Modells geben. Das Protokollanalysetool wiederum kann verwendet werden, um sinnvolle Informationen aus der enormen Menge an Protokolldaten herauszuziehen.

Traceprotokoll verwenden

Im Traceprotokoll wird die Ausführung der Regelengine protokolliert. Dies ist häufig der zeitaufwendigste Teil der Anfragen des Konfigurators an den Server. Das Traceprotokoll stellt die Informationen bereit, die zum Beheben (Debuggen) von Fehlern in Regeln benötigt werden, die ein Fehlverhalten aufweisen, und um die Ausführungszeit von Regeln aufzuzeichnen. Ihr Debuggingprozess muss also immer mit der Prüfung des Traceprotokolls beginnen.

Traceprotokolle werden mit Visual Modeler erstellt. Gehen Sie wie folgt vor:

1. Wechseln Sie in die Navigation "Modellgruppe" und navigieren Sie dort zu dem Modell, das Sie debuggen wollen.
2. Wählen Sie das Modell aus der Anzeige "Modelle und Gruppen" aus.
3. Das Modell wird auf der Registerkarte "Modellvorschau" angezeigt.
4. Klicken Sie auf das Testsymbol.
5. Das Modell wird in einem separaten Fenster ausgeführt.
6. Klicken Sie auf "Fehler beheben".

Das Traceprotokoll wird in einem separaten Fenster angezeigt.

Das Protokoll besteht aus zwei Abschnitten. Im ersten Abschnitt befindet sich der Trace für Regelauslösung und im zweiten Abschnitt der Eigenschaftspool, wie er sich am Ende der Regelauslösung darstellt.

Die folgende Abbildung zeigt einen Abschnitt mit einem Beispieltrace für eine Regelauslösung.

Trace für Regelauslösung

| # | (ms) | Ergebnis |
|----|------|--|
| 0 | 0 | Applying picks |
| 1 | 0 | Firing phase [0]:begin |
| 2 | 0 | Firing rules on MXDS-7500.Disk Drives |
| 3 | 0 | MSG_E_Available_HDD_Slots ==> fires on TRUE - priority = 50 |
| 4 | 0 | Property not found [MX75_HDD_Ordered or MX75_Bays_Available], taking null action |
| 5 | 0 | took 0ms. |
| 6 | 0 | Firing rules on MXDS-7500.Memory |
| 7 | 0 | MSG_E_MX75_Memory_Software_Check ==> fires on TRUE - priority = 50 |
| 8 | 0 | TESTING:sum(MX75_Mem_Ordered) <sum(MX75_Mem_Required) [nullreturn=false] |
| 9 | 0 | FALSE: 0.0 < 0.0 |
| 10 | 0 | FALSE: sum(MX75_Mem_Ordered) <sum(MX75_Mem_Required) [nullreturn=false] |
| 11 | 0 | took 0ms. |
| 12 | 0 | Firing rules on MXDS-7500.Placeholder for auto memory selection |
| 13 | 0 | ASG make placeholder invisible ==> fires on TRUE - priority = 50 |
| 14 | 0 | Left side property [MX75_Mem_Auto_Select] not found, taking null action |
| 15 | 0 | took 0ms. |
| 16 | 0 | Firing rules on MXDS-7500.AutoMemory |
| 17 | 0 | EXP_MX75_Automatic_Memory_Selection ==> fires on TRUE - priority = 50 |
| 18 | 0 | Left side property [MX75_Mem_Auto_Select] not found, taking null action |
| 19 | 0 | took 0ms. |
| 20 | 0 | Firing rules on MXDS-7500.Software.Application |
| 21 | 15 | EXP_MX75_Fire_Wire ==> fires on TRUE - priority = 50 |
| 22 | 15 | Left side property [MX75_Video_Editing] not found, taking null action |
| 23 | 15 | took 15ms. |

Der Trace für Regelauslösung weist drei Spalten auf:

- Eine Folgennummer für die Kommunikation mit anderen Benutzern zu Regelproblemen. Dies vereinfacht die Beschreibung eines Problems in der Art "Sehen Sie Zeile 42 mit folgendem Wortlaut?".
- Antwortzeit. Hier wird der Zeitraum zwischen der Vornahme des Protokolleintrags und dem Beginn der Regelauslösung protokolliert.
- Hauptteil des Traceprotokolls. Im Hauptteil werden die einzelnen Punkte der Regelauslösung angezeigt. Dies umfasst beispielsweise eine auszuwertende Bedingung, eine auftretende Zuweisung, den Beginn oder den Abschluss einer Regel usw.

Das Protokoll zeigt zudem die Anzahl der Millisekunden an, die das Auslösen einer Regel nach jedem Regelauslösungseintrag dauert. Die Gesamtzahl der Millisekunden für das Ausführen des Modells wird am Ende des Trace für die Regelauslösung protokolliert.

Der Eigenschaftspool umfasst ebenfalls drei Spalten:

- Der Name steht für den vollständigen Pfadnamen zum Artikel sowie zur Eigenschaft für diesen Artikel.
- Typ steht für den Eigenschaftstyp für die benannte Eigenschaft (z. B. Numerisch, Liste oder Zeichenfolge).
- Wert gibt den Wert der Eigenschaft nach dem Auslösen der Regel an.

Die folgende Abbildung zeigt einen Abschnitt mit einem Beispieltrace für einen Eigenschaftspool.

| Eigenschaftspool | | |
|---|--------------|---|
| Name | Typ | Wert |
| MXDS-7500.CONFIG: FIRST FIRE | Numerisch | 1.0 |
| MXDS-7500.MX75_Bays_Available | Numerisch | 2.0 |
| MXDS-7500.MX75_Card_Slot_Available | Numerisch | 4.0 |
| MXDS-7500.MX75_Mem_Ordered | Numerisch | 0.0 |
| MXDS-7500.MX75_Mem_Required | Numerisch | 0.0 |
| MXDS-7500.Service Options.View Service.UI: COLUMN SPAN | Numerisch | 2.0 |
| MXDS-7500.Placeholder for auto memory selection.UI: CONFIG CELL HTML CLASS | Zeichenfolge | configsubcell_plain |
| MXDS-7500.Accessory Cards Message.Accessory Card Image.UI: CONFIG CELL HTML CLASS | Zeichenfolge | configsubcell_plain |
| MXDS-7500.Microprocessor.UI: CONSTANT GUIDING TEXT | Zeichenfolge | Dual processor capable motherboard, supporting Intel processors |
| MXDS-7500.Microprocessor.UI: CONTROL | Zeichenfolge | RADIO |
| MXDS-7500.Disk Drives.UI: CONTROL | Zeichenfolge | RADIO |
| MXDS-7500.Placeholder for auto memory selection.UI: CONTROL | Zeichenfolge | controls/allpicked.jsp |
| MXDS-7500.Software.UI: CONTROL | Zeichenfolge | CHECKBOX |
| MXDS-7500.Accessory Cards Message.UI: CONTROL | Zeichenfolge | controls/allpicked.jsp |
| MXDS-7500.Accessory Cards Message.Accessory Card Image.UI: CONTROL | Zeichenfolge | controls/allpicked.jsp |
| MXDS-7500.Accessory.Graphic Cards.UI: CONTROL | Zeichenfolge | RADIO |
| MXDS-7500.Accessory.Cards.UI: CONTROL | Zeichenfolge | CHECKBOX |
| MXDS-7500.Accessory.Network Cards.UI: CONTROL | Zeichenfolge | RADIO |
| MXDS-7500.Service Options.View Service.UI: CONTROL | Zeichenfolge | CHECKBOX |
| MXDS-7500.Placeholder for auto memory selection.UI: DEFAULT SELECTION | Zeichenfolge | no |
| MXDS-7500.Accessory Cards Message.UI: DEFAULT SELECTION | Zeichenfolge | no |
| MXDS-7500.Accessory Cards Message.Accessory Card Image.UI: DEFAULT SELECTION | Zeichenfolge | no |

Verwenden Sie dieses Protokoll "single user", um ein Gefühl dafür zu bekommen, wie umfassend die Regeln pro Klick sind. Überprüfen Sie, wie lange es bis zum Auslösen der Regeln dauert. Dauert die Antwort länger als 100-200 ms, liegt möglicherweise ein Skalierbarkeitsproblem vor. Ist dies der Fall, finden Sie anhand des Traceprotokolls heraus, ob das Leistungsverhalten bestimmter Regeln unzureichend ist.

Berichtstool für Modelle

Das Berichtstool für Modelle bietet einen Überblick über die Größe eines Modells in Relation zu anderen Modellen. Verwenden Sie dieses Tool beim Treffen von Entscheidungen, welche Modelle getestet werden sollen. Sie können die erzielten Testergebnisse verfolgen, sodass Sie die Anzahl der Änderungen am Modell ermitteln können.

Tools für Belastungstests

Tools für Belastungstests helfen Ihnen bei der Bestimmung des Leistungsverhaltens Ihres Modells nach dessen Implementierung. Folgende Voraussetzungen müssen erfüllt sein, bevor Sie solche Tools verwenden können:

- Sie müssen verstehen, was zu testen ist.
- Sie müssen Ihre Testfälle eingrenzen, sodass Sie wissen, was die Auswirkungen bedeuten (lokale/ferne LAN-Tests, Test mit/ohne Clustering, Tests mit/ohne Web-Fronting usw.).

- Sie müssen verstehen, dass bei Modelländerungen auch die Scripts geändert werden müssen, die Sie zum Ausführen der Tests und zur Wiedergabe von Testszenarios verwenden.
- Sind diese globaler als die aktuelle Modellgruppe, müssen Sie diese am niedrigsten Punkt in der Baumstruktur der Modellgruppe definieren, die ein Vorgänger eines Modells ist, für das die Eigenschaft verwendet werden soll.

Trace für Regelauslösung

| # | (ms) | Ergebnis |
|----|------|--|
| 0 | 0 | Applying picks |
| 1 | 0 | Firing phase [0]:begin |
| 2 | 0 | Firing rules on MXDS-7500.Disk Drives |
| 3 | 0 | MSG_E_Available_HDD_Slots ==> fires on TRUE - priority = 50 |
| 4 | 0 | Property not found [MX75_HDD_Ordered or MX75_Bays_Available], taking null action |
| 5 | 0 | took 0ms. |
| 6 | 0 | Firing rules on MXDS-7500.Memory |
| 7 | 0 | MSG_E_MX75_Memory_Software_Check ==> fires on TRUE - priority = 50 |
| 8 | 0 | TESTING:sum(MX75_Mem_Ordered) <sum(MX75_Mem_Required) [nullreturn=false] |
| 9 | 0 | FALSE: 0.0 < 0.0 |
| 10 | 0 | FALSE: sum(MX75_Mem_Ordered) <sum(MX75_Mem_Required) [nullreturn=false] |
| 11 | 0 | took 0ms. |
| 12 | 0 | Firing rules on MXDS-7500.Placeholder for auto memory selection |
| 13 | 0 | ASG make placeholder invisible ==> fires on TRUE - priority = 50 |
| 14 | 0 | Left side property [MX75_Mem_Auto_Select] not found, taking null action |
| 15 | 0 | took 0ms. |
| 16 | 0 | Firing rules on MXDS-7500.AutoMemory |
| 17 | 0 | EXP_MX75_Automatic_Memory_Selection ==> fires on TRUE - priority = 50 |
| 18 | 0 | Left side property [MX75_Mem_Auto_Select] not found, taking null action |
| 19 | 0 | took 0ms. |
| 20 | 0 | Firing rules on MXDS-7500.Software.Application |
| 21 | 15 | EXP_MX75_Fire_Wire ==> fires on TRUE - priority = 50 |
| 22 | 15 | Left side property [MX75_Video_Editing] not found, taking null action |
| 23 | 15 | took 15ms. |

Der Trace für Regelauslösung weist drei Spalten auf:

- Eine Folgenummer für die Kommunikation mit anderen Benutzern zu Regelproblemen. Dies vereinfacht die Beschreibung eines Problems in der Art "Sehen Sie Zeile 42 mit folgendem Wortlaut?" .
- Antwortzeit. Hier wird der Zeitraum zwischen der Vornahme des Protokolleintrags und dem Beginn der Regelauslösung protokolliert.
- Hauptteil des Traceprotokolls. Im Hauptteil werden die einzelnen Punkte der Regelauslösung angezeigt. Dies umfasst beispielsweise eine auszuwertende Bedingung, eine auftretende Zuweisung, den Beginn oder den Abschluss einer Regel usw.

Das Protokoll zeigt zudem die Anzahl der Millisekunden an, die das Auslösen einer Regel nach jedem Regelauslösungseintrag dauert. Die Gesamtzahl der Millisekunden für das Ausführen des Modells wird am Ende des Trace für die Regelauslösung protokolliert.

Der Eigenschaftspool umfasst ebenfalls drei Spalten:

- Der Name steht für den vollständigen Pfadnamen zum Artikel sowie zur Eigenschaft für diesen Artikel.
- Typ steht für den Eigenschaftstyp für die benannte Eigenschaft (z. B. Numerisch, Liste oder Zeichenfolge).
- Wert gibt den Wert der Eigenschaft nach dem Auslösen der Regel an.

Die folgende Abbildung zeigt einen Abschnitt mit einem Beispieltrace für einen Eigenschaftspool:

| Eigenschaftspool | | |
|---|--------------|---|
| Name | Typ | Wert |
| MXDS-7500.CONFIG: FIRST FIRE | Numerisch | 1.0 |
| MXDS-7500.MX75_Bays_Available | Numerisch | 2.0 |
| MXDS-7500.MX75_Card_Slot_Available | Numerisch | 4.0 |
| MXDS-7500.MX75_Mem_Ordered | Numerisch | 0.0 |
| MXDS-7500.MX75_Mem_Required | Numerisch | 0.0 |
| MXDS-7500.Service Options.View Service.UI: COLUMN SPAN | Numerisch | 2.0 |
| MXDS-7500.Placeholder for auto memory selection.UI: CONFIG CELL HTML CLASS | Zeichenfolge | configsubcell_plain |
| MXDS-7500.Accessory Cards Message.Accessory Card Image.UI: CONFIG CELL HTML CLASS | Zeichenfolge | configsubcell_plain |
| MXDS-7500.Microprocessor.UI: CONSTANT GUIDING TEXT | Zeichenfolge | Dual processor capable motherboard, supporting Intel processors |
| MXDS-7500.Microprocessor.UI: CONTROL | Zeichenfolge | RADIO |
| MXDS-7500.Disk Drives.UI: CONTROL | Zeichenfolge | RADIO |
| MXDS-7500.Placeholder for auto memory selection.UI: CONTROL | Zeichenfolge | controls/allpicked.jsp |
| MXDS-7500.Software.UI: CONTROL | Zeichenfolge | CHECKBOX |
| MXDS-7500.Accessory Cards Message.UI: CONTROL | Zeichenfolge | controls/allpicked.jsp |
| MXDS-7500.Accessory Cards Message.Accessory Card Image.UI: CONTROL | Zeichenfolge | controls/allpicked.jsp |
| MXDS-7500.Accessory.Graphic Cards.UI: CONTROL | Zeichenfolge | RADIO |
| MXDS-7500.Accessory.Cards.UI: CONTROL | Zeichenfolge | CHECKBOX |
| MXDS-7500.Accessory.Network Cards.UI: CONTROL | Zeichenfolge | RADIO |
| MXDS-7500.Service Options.View Service.UI: CONTROL | Zeichenfolge | CHECKBOX |
| MXDS-7500.Placeholder for auto memory selection.UI: DEFAULT SELECTION | Zeichenfolge | no |
| MXDS-7500.Accessory Cards Message.UI: DEFAULT SELECTION | Zeichenfolge | no |
| MXDS-7500.Accessory Cards Message.Accessory Card Image.UI: DEFAULT SELECTION | Zeichenfolge | no |

Verwenden Sie dieses Protokoll "single user", um ein Gefühl dafür zu bekommen, wie umfassend die Regeln pro Klick sind. Überprüfen Sie, wie lange es bis zum Auslösen der Regeln dauert. Dauert die Antwort länger als 100-200 ms, liegt möglicherweise ein Skalierbarkeitsproblem vor. Ist dies der Fall, finden Sie anhand des Traceprotokolls heraus, ob das Leistungsverhalten bestimmter Regeln unzureichend ist.

Tool für Cachestatus

Sie können den aktuellen Inhalt des Cache mithilfe von "cmd=configstatus" anzeigen.

Leistung

Regeln für das Leistungsverhalten eines Modells

- Komplexe Pfade zu Artikeln.
- Bei einer Regel, die wie folgt die Speicherkapazität erweitert:

$$\text{totalMem} = \text{value}(*.adapter.1.memory) + \text{value}(*.adapter.2.memory) + \text{value}(*.adapter.3.memory) + \text{value}(*.adapter.4.memory)$$
- ist das Leistungsverhalten deutlich schlechter als bei:

```
totalMem = sum(memory)
```

- Wenn die Speichereigenschaft auch an anderen Positionen für andere Zwecke verwendet wird, sodass "sum(memory)" einen falschen Wert ergibt, müssen Sie weitere Eigenschaften namens "adapterMemory" für die Adapterartikel 1-4 erstellen und die folgende Gleichung verwenden:

```
totalMem = sum(adapterMemory).
```

- Dies erfordert deutlich weniger Verwaltungsaufwand als für den folgenden Codeblock:

```
totalMem = value(*.adapter.1.memory) + value(*.adapter.2.memory) +  
value(*.adapter.3.memory) + value(*.adapter.4.memory)
```

- Schreiben Sie die Regeln so, dass sie nur ausgelöst werden, wenn sie wirklich gebraucht werden:

Eine Regel, die "totalMem = sum(mem)" zuweist, muss nur ausgelöst werden, wenn "count(mem) > 0" zutrifft.

Eigenschaften für das Leistungsverhalten von Modellen

- Definieren von Eigenschaften an der richtigen Position in der Modellgruppenhierarchie
 - Treffen die Eigenschaft nur lokal für dieses Modell zu, definieren Sie diese im Modell.
 - Können die Eigenschaften auch von anderen Modellen in dieser Modellgruppe verwendet werden, definieren Sie sie in der aktuellen Modellgruppe.
 - Sind diese globaler als die aktuelle Modellgruppe, müssen Sie diese am niedrigsten Punkt in der Baumstruktur der Modellgruppe definieren, die ein Vorgänger eines Modells ist, für das die Eigenschaft verwendet werden soll.

Kapitel 4. Daten archivieren

Die Verwaltung Ihrer Daten ist eine zentrale Voraussetzung zum Schutz Ihres Unternehmens. Hierzu gehört die regelmäßige Archivierung Ihrer Produktionsdaten und die Einrichtung primärer und sekundärer Standorte für die Speicherung Ihrer Datenbankarchive, vorzugsweise an einem anderen Standort. Stellen Sie einen Zeitplan für regelmäßige Archivierungsaktivitäten auf, der tägliche Teilsicherungen und wöchentliche Gesamtsicherungen umfasst (eventuell auch häufiger, wenn die Menge Ihrer Unternehmensdaten dies erfordert).

Kapitel 5. Statistikdaten aktualisieren

Die Aktualisierung von Datenbankstatistiken ermöglicht dem Optimierungsprogramm für Datenbankabfragen, Datenbankindexe erneut zu überprüfen und die meisten effizienten Pfade für den Datenabruf erneut zu berechnen. In diesem Abschnitt werden zwei Scripts vorgestellt: ein Script für die Aktualisierung der Statistikdaten einer Oracle-Datenbank und ein Script für die Aktualisierung der Statistikdaten einer SQL Server-Datenbank.

Wenden Sie sich an Ihren Datenbankadministrator, wenn Sie die Statistikdaten zu den Tabellen fehlerfrei aktualisieren wollen, oder ziehen Sie die Datenbankdokumentation zurate.

Statistikdaten für eine Oracle-Datenbank aktualisieren

Das folgende Beispiel zeigt, wie Statistikdaten für eine Oracle-Datenbank auf Schemaebene aktualisiert werden. Ersetzen Sie *schemaname*, *eignername* und *tabellenname* durch die jeweiligen Namen für Schema, Eigener und Tabelle.

```
EXEC DBMS_STATS.GATHER_SCHEMA_STATS(  
  ownname=> 'schemaname' ,  
  cascade=> TRUE,  
  estimate_percent=> DBMS_STATS.AUTO_SAMPLE_SIZE,  
  degree=> DBMS_STATS.AUTO_DEGREE,  
  granularity=>'AUTO',  
  method_opt=> 'FOR ALL COLUMNS SIZE AUTO');
```

Das folgende Beispiel zeigt, wie Statistikdaten für eine Oracle-Datenbank auf Tabellenebene aktualisiert werden.

```
exec dbms_stats.gather_table_stats(  
  ownname=> 'eignername',  
  tabname=> 'tabellenname',  
  estimate_percent=> DBMS_STATS.AUTO_SAMPLE_SIZE,  
  cascade=> DBMS_STATS.AUTO_CASCADE,  
  degree=> null,  
  no_invalidate=> DBMS_STATS.AUTO_INVALIDATE,  
  granularity=> 'AUTO',  
  method_opt=> 'FOR ALL COLUMNS SIZE AUTO');
```

Sie können je nach Ihren Anforderungen Statistikdaten auch auf Datenbank- oder Indexebene aktualisieren.

Statistikdaten für eine SQL Server-Datenbank aktualisieren

Das folgende Beispiel zeigt, wie Statistikdaten für eine SQL Server-Datenbank auf Tabellenebene aktualisiert werden. Ersetzen Sie *tabellenname* und *indexname* durch die jeweiligen Namen für die Tabelle und den Index.

```
UPDATE STATISTICS ON <tabellenname> [ . <indexname> ]  
  WITH FULLSCAN {, NORECOMPUTE }
```

Kapitel 6. Richtlinien für JVM-Speichereinstellungen und Leistungsoptimierung

Wenn Sie speicherspezifische Probleme feststellen, kann die Anpassung der JVM-Speichereinstellungen helfen, wieder eine stabile Arbeitsumgebung herzustellen. Dieser Abschnitt enthält Richtlinien zu den JVM-Speichereinstellungen und zur Leistungsoptimierung. Um diese Richtlinien anwenden zu können, müssen Sie mit Ihrer JVM- und Servlet-Container-Umgebung vertraut sein.

JVM-Speichereinstellungen anpassen

Als generelle Regel gilt, dass der JVM, auf der Ihr Anwendungsserver läuft, so viel Speicher wie möglich zugeordnet werden sollte. Dies können Sie durch Einstellung der JVM-Speicherkonfiguration wie folgt erreichen:

- "-Xmx" muss auf einen Wert zwischen 80 % und 100 % des physischen Systemspeichers gesetzt werden. Wenn Sie einen zu niedrigen Wert für "-Xmx" angeben, kann der Anwendungsserver mit einem OutOfMemory-Fehler ausfallen. Wenn Sie für "-Xmx" einen zu hohen Wert angeben, ist der Speicherbedarf zu groß und Sie laufen Gefahr, dass es beim Java-Heapspeicher zu einer Swap-out-Situation kommt, wodurch andere Leistungsprobleme verursacht werden.
- Der Wert für "-Xms" muss ca. die Hälfte des Werts für "-Xmx" betragen. Wenn Sie Langzeitdaten zum stabilen Speicherbelegungspunkt Ihres Anwendungsservers haben, können Sie für "-Xms" in etwa diesen Wert angeben. Als weitere Option können Sie für "-Xms" den Speicherbelegungswert am InitServlet-Ende angeben. Dadurch wird mindestens sichergestellt, dass die DEBS-Initialisierung mit nur minimalem Garbage-Collection-Aufwand abgeschlossen wird. Um diesen Speicherbelegungswert zu erhalten, müssen Sie wie folgt vorgehen:
 1. Geben Sie für "-Xms" denselben Wert wie für "-Xmx" an.
 2. Starten Sie Visual Modeler und warten Sie, bis die Initialisierung abgeschlossen ist.
 3. Greifen Sie auf die Homepage Ihrer E-Commerce-Site zu.
 4. Öffnen Sie die Datei **debs.log** in einem Texteditor und prüfen Sie den Protokolleintrag, der in etwa wie folgt aussieht:

```
2003.03.18 ... END Request ... Mem=129380744/388726784/391291344 ...
```
 5. Die erste Zahl nach **Mem=** ist die aktuelle Speicherbelegung nach der Initialisierung. Geben Sie für "-Xms" diese Zahl an: Im obigen Beispiel wäre dies "-Xms128m".
- "-XX:MaxPermSize" steuert die Zuordnungsgröße für systemähnliche, reflektive Objekte wie Klasse oder Methode. Es wird empfohlen, als Anfangswert "128m" anzugeben.

Bei Webanwendungen wird der zugeordnete Speicherbereich schnell belegt, da die *.jsp-Dateien zuerst in *.java-Dateien und dann in *.class-Dateien konvertiert werden. Diese werden dann in den Hauptspeicher geladen, der durch "-XX:MaxPermSize" angegeben wurde. Ab Java Version 1.4.2 können Sie "-XX:+PrintGCDetails" für die Überwachung der Details zu diesem Speicherbereich verwenden. Dies wird auch als permanente Generierung bezeichnet.

Achten Sie darauf, dass Sie keine speicherbezogenen Änderungen vornehmen. Diese könnten zu Konflikten mit den vom Anwendungsserver unterstützten Komponenten führen. DEBS muss fehlerfrei zusammen mit dem Anwendungsserver in derselben virtuellen Maschine laufen. Wenn Sie also Zweifel haben, ziehen Sie die Dokumentation zum Anwendungsserver zurate oder wenden Sie sich an die Systemunterstützung für Ihren Anwendungsserver. Hierzu folgendes Beispiel: Angenommen, die aktuelle Dokumentation zum Anwendungsserver gibt an, dass die JVM-Einstellung "-server" nicht unterstützt wird. In diesem Fall dürfen Sie keinen Wert für "-server" angeben.

Als Abhilfemaßnahme für einen solchen Fehler starten Sie die virtuelle Maschine ohne Angabe zusätzlicher Argumente, fügen Sie schrittweise ein Argument hinzu, starten Sie die Maschine erneut, beobachten Sie die Ergebnisse, fügen Sie ein weiteres Argument hinzu usw. Fahren Sie damit so lange fort, bis Sie korrekte Ergebnisse erzielen.

Zusätzliche Leistungsoptimierung

Eine weitere Leistungsoptimierung kann bei den Java-Garbage-Collection-Aktivitäten und durch Anpassung der Speichereinstellungen für andere Bereiche wie Threads, JVM-Stacks oder native Strukturen und Codes erzielt werden. Verwenden Sie hierzu das Tool Log Analyzer oder prüfen Sie die Datei **debs.log** direkt auf Bereiche mit Leistungsproblemen.

Garbage-Collection-Aktivitäten verfolgen

Wenn Sie unerklärliche Pausensituationen feststellen, kann es sein, dass die virtuelle Maschine wegen eines vollständigen Garbage-Collection-Durchgangs angehalten wird. Um dies zu bestätigen, können Sie die JVM-Einstellung "-verbose:gc" verwenden und damit die Aufzeichnung von Garbage-Collection-Ereignissen in der Datei **debs.log** aktivieren. Garbage-Collection-Ereignisse sehen wie folgt aus:

```
[GC 325816K->83372K(776768K), 0.2454258 secs]
[Fu11 GC 267628K->83769K(776768K), 1.8479984 secs]
```

Ein untergeordneter Garbage-Collection-Vorgang dauert in der Regel weniger als eine Sekunde. Ein umfassender Garbage-Collection-Vorgang ist normalerweise in weniger als drei Sekunden abgeschlossen. Dauert der Vorgang länger als drei Sekunden, weist dies auf eine "Out-of-Range"-Bedingung hin, die genauer untersucht werden muss.

Andere Garbage-Collection-Traceeinstellungen, die für Sie von Interesse sein könnten, sind nachfolgend aufgeführt:

- Die JVM-Einstellung "-server": Mit dieser Einstellung werden einige Anfangseinstellungen für den Java-Heapspeicher angepasst, sodass sie für eine Serverumgebung besser geeignet sind. Legen Sie den "-server"-Wert nur fest, wenn Ihr Anwendungsserver diesen unterstützt.

Es gibt ein bekanntes Problem mit der "-server"-Einstellung im Zusammenhang mit einem Fehler bei Just-in-time-Kompilierungen, wodurch der vom Datenservice verwendete Wert unerwartet geändert wird. Als Ergebnis daraus wird DEBS nicht initialisiert (InitServlet schlägt fehl). Wenden Sie sich in diesem Fall an Ihren Ansprechpartner, um zu erfahren, wie die Just-in-time-Kompilierung bei bestimmten Methoden inaktiviert wird.

Bei einigen Anwendungsservern wird die VM-Einstellung "-server" empfohlen. Insbesondere lautet der "-XX:NewRatio"-Wert für "-server" 2 (der Standardwert

für die "-client"-Einstellung lautet 8). Weitere Informationen zu den Einstellungen für "-server" und "-client" finden Sie in der Sun-Dokumentation bei den folgenden URLs:

http://java.sun.com/docs/books/performance/1st_edition/html/JAppHotspot.fm.html#998292

http://java.sun.com/docs/books/performance/1st_edition/html/JAppHotspot.fm.html#998359

- Die Einstellung "-Xincgc": Diese Einstellung aktiviert die schrittweise Garbage-Collection. Durch diese Einstellung werden lange Pausen aufgrund von vollständigen Garbage-Collection-Vorgängen reduziert. Wenn Sie diese Einstellung verwenden, müssen Sie beachten, dass Sie den Zeitaufwand für die Ausführung eines komplexen Garbage-Collection-Vorgangs auf mehrere kleine Vorgänge verschieben. Mit einer solchen Verlagerung sind Aufwandskosten verbunden, die in der Regel bei 10 % liegen.
- Wenn Sie OutOfMemoryError-Nachrichten erhalten, müssen Sie zuerst den Wert für "-Xmx" erhöhen, wobei sichergestellt werden muss, dass dieser Wert nicht größer ist als der Wert für den physischen Speicher des Systems. Wenn Sie OutOfMemoryError-Nachrichten erhalten, wenn die aktuelle Heapspeichernutzung (bei der neue Objekte zugeordnet werden) nicht annähernd dem Wert für "-Xmx" entspricht, besteht die Möglichkeit, dass andere Bereiche der Speicherzuordnung erschöpft sind. Überprüfen Sie den Log Analyzer-Bericht und auf dieser Basis die folgenden Bereiche:
 - Klassen: Versuchen Sie die Einstellung "-XX:MaxPermSize=128m".
 - Threads: Versuchen Sie, den Stack mithilfe von "-Xss=512k" anzupassen.
 - JVM-Stacks: Versuchen Sie, den Stack mithilfe von "-Xss=512k" anzupassen.
 - Native Datenstrukturen: Versuchen Sie, die Größe des Auslagerungsspeichers im Betriebssystem anzupassen.
 - Native Codes: Versuchen Sie, die Größe des Auslagerungsspeichers im Betriebssystem anzupassen.

Kapitel 7. Log Analyzer

Log Analyzer ist ein Open-Source-Tool, das Sie bei der Analyse der **debs.log**-Einträge in Visual Modeler unterstützen kann. Das Tool enthält eine Ansicht mit wesentlichen Leistungsindikatoren oder KPIs: Threads, Speicher, Anforderungen, Sitzungen sowie Antwortzeiten, die nach Benutzer und Anforderungstyp sortiert sind.

Der Einsatz von Log Analyzer im Rahmen der täglichen, routinemäßigen Überwachung Ihres Systems bietet zahlreiche Vorteile:

- Tägliche Protokollanalyseberichte für mehr Zuverlässigkeit und Stabilität Ihres Systems. Die generierten Daten können frühzeitig Warnungen zu möglichen Problemen ausgeben, wodurch Ausfallzeiten verhindert werden können. Mit den täglichen Protokollanalyseberichten können Sie beispielsweise proaktiv planen, einen Anwendungsserver neu zu starten, sobald der Speicher nahezu vollständig belegt ist.
- Solche Berichte bilden die Basis für die Behebung eines aktuellen Problems. Durch die genaue Prüfung dieser Berichte können Sie ermitteln, wann das Problem zum ersten Mal auftrat, und es mit Ereignissen wie einem Betriebssystemupgrade in Beziehung setzen.
- Tägliche Protokollanalyseberichte sind zudem ein zentraler Anlaufpunkt für schrittweise Verbesserungen. Wenn Sie diese Berichte täglich prüfen, können Sie eine To-do-Liste erstellen und den Neustart des Anwendungsservers zeitlich planen, Ausnahmelisten bereinigen und blockierte Threads verfolgen. Sie können auch Rückmeldungen an Entwickler zu lange laufenden Anforderungen oder Anforderungen geben, die wichtige Ressourcen belegen (z. B. durch Rückgabe langer Zeilen aus der Datenbank).

Das Tool Log Analyzer erhalten Sie über Ihren Ansprechpartner. Log Analyzer ist eine **.jar**-Datei, die gespeichert und an der von Ihnen gewünschten Position platziert werden kann. Log Analyzer setzt voraus, dass das Format von DEBS-Protokolleinträgen in etwa dem folgenden Format entspricht:

```
<JJJJ.MM.TT HH:MM:SS:mss threadname:protokollebene:protokolltag:nachrichten>
```

Beispiel:

```
2006.10.12 06:00:00:171 Env/http-8580-Processor48:INFO:WrappingFilter ...
```

Da die Analyse einer Protokolldatei sehr speicherintensiv sein kann, geben Sie so viel JVM-Speicher wie möglich an, um OutOfMemoryError-Nachrichten zu vermeiden. Starten Sie Log Analyzer beispielsweise wie folgt:

```
java -Xms256m -Xmx512m -jar logAnalyzer-1.1.1-SNAPSHOT-app.jar
```

Die Eingangsanzeige von Log Analyzer wird angezeigt:

Geben Sie die folgenden Informationen ein:

- **Source:** Geben Sie den vollständigen Pfadnamen der Position der DEBS-Protokolldatei oder ein Verzeichnis ein, das mehrere DEBS-Protokolldateien enthält.
- **Input:**

- **DEBS 6.4 oder höher** wird automatisch markiert. Wenn Sie eine Protokolldatei aus einem Visual Modeler-Release vor Release 6.4 analysieren, müssen Sie die Markierung dieses Kontrollkästchens aufheben.
- **From WebLogic:** Klicken Sie dieses Kontrollkästchen an, um anzugeben, dass die Protokolldateien aus einem Oracle WebLogic-Anwendungsserver generiert werden.
- **Output:** Klicken Sie das Kontrollkästchen **Output** an, um ein Antwortzeitdiagramm nach Nachrichtentyp gruppiert zu generieren.
- **Output dir:** Geben Sie den vollständigen Pfadnamen zum Verzeichnis mit der Berichtsausgabe ein.

Klicken Sie auf **Start analyzing**, um den Protokollanalyseprozess zu starten. Log Analyzer zeigt während des Verarbeitungsfortschritts Nachrichten an und stellt nach Abschluss die Protokollanalyseausgabe in das angegebene Ausgabeverzeichnis.

Tägliche Log Analyzer-Berichte konfigurieren

In diesem Abschnitt wird ein Verfahren zur automatischen Erstellung der täglichen Protokollanalyseberichte beschrieben. Bei dem hier beschriebenen Verfahren wird Ant verwendet, da Ant portierbar, bewährt und gut dokumentiert ist. Ant ist verfügbar über <http://ant.apache.org>.

Bei diesem Verfahren sollen die folgenden Ziele erreicht werden:

- Definition eines CRON-Jobs zur Ausführung von Berichten über Nacht und Zusammenfassung der Ausgabe nach Datum (Jahr/Monat/Tag), um die Navigation zu erleichtern.
- Komprimierung der Protokolldateien, um Speicherplatz zu sparen.
- Konfiguration der Automatisierung in einer Weise, die einfach zu kopieren ist, sodass Protokolldateien aus mehreren Systemen über einen zentralen Protokollserver gehostet werden können.

Die automatische Erstellung von Log Analyzer-Berichten setzt Folgendes voraus:

- Java und Ant
- Lesezugriff auf die DEBS-Protokolldateien
- Schreibzugriff auf das Verzeichnis für die Berichtsausgabe (`<out.dir>`). Der Inhalt von `<out.dir>` muss über einen Web-Server zugänglich sein.

Tägliche Log Analyzer-Berichte automatisieren

Informationen zu diesem Vorgang

Im Folgenden wird der allgemeine Ablauf für die Automatisierung der täglichen Protokollanalyseberichte beschrieben.

Vorgehensweise

1. DEBS erstellt Protokolldateien im **logs**-Verzeichnis des Anwendungsservers oder Servlet-Containers.

Das **logs**-Verzeichnis in einem Tomcat-System lautet beispielsweise `<tomcat-home>\logs`.

Die Protokolldatei hat den Namen **debs.log.n**, wobei *n* für eine Zahl steht. Beispiel: `debs.log.1`, `debs.log.2` usw.

2. Definieren Sie einen CRON-Job, der täglich ausgeführt wird (vielleicht sehr früh am Morgen), um alle Protokolldateien aus dem "log"-Verzeichnis in einer temporären Datei zu verknüpfen.
3. Extrahieren Sie aus der temporären Datei die Protokolleinträge vom Vortag in das Log Analyzer-Ausgabeverzeichnis und verwenden Sie dabei das Verzeichnisnamensmuster `jahr/monat/tag/log.suffix`.
4. Die Datei `jahr/monat/tag/log.suffix` wird mithilfe von `gzip` weiter komprimiert, um Platz zu sparen.
5. Starten Sie Log Analyzer, um die Datei `jahr/monat/tag/log.suffix.gz` zu parsen und den Bericht im Verzeichnis `jahr/monat/tag/html/` zu erstellen.

Tägliche Log Analyzer-Berichte konfigurieren

Vorgehensweise

1. Wenden Sie sich an Ihren Ansprechpartner, um folgende Dateien zu erhalten:
 - Log Analyzer-**jar**-Datei
 - **logAnalyzer-daily.xml**
 - `logAnalyzer-daily.properties`
2. Speichern Sie die Log Analyzer-Dateien in einem temporären Verzeichnis.
3. Informationen zu Konfigurationswerten finden Sie in „Log Analyzer konfigurieren“ auf Seite 38.
4. Führen Sie über den folgenden Befehl den täglichen Protokollanalysebericht aus:


```
ant -Dproperties.file.name="logAnalyzer-daily.properties" -f logAnalyzer-daily.xml
```
5. Prüfen Sie die Ausgabe. Die Position entspricht in etwa der folgenden:


```
sites/default/app-server/logAnalyzer-out.d/dailySplit/JJJJ/MM/TT/html/index.html
```

Empfohlene Verzeichnisstruktur

Die folgende Abbildung zeigt die empfohlene Verzeichnisstruktur für Log Analyzer. Diese Struktur wird besonders empfohlen, wenn Sie Protokolldateien von mehreren Standorten aus hosten wollen.

```
# where to keep the log analyzer jar file
bin/
  logAnalyzer-1.1.1-SNAPSHOT-app.jar

# ant script
logAnalyzer-daily.xml

# sites data
sites/
  site1/
  ...
  site2/
  ...
  siteN/
    logAnalyzer-daily.properties
  app-server/
    logs/
      debs.log
      debs.log.1
      debs.log.2
```

Siteinformationen werden im Verzeichnis **sites** abgelegt, das jeweils ein Verzeichnis für jede Site enthält. Der Name für dieses Siteverzeichnis kann eine beliebige eindeutige Zeichenfolge sein. Im obigen Beispiel wurde "siten" verwendet, wobei *n* eine Zahl ist: site1, site2 usw.

Jedes Siteverzeichnis enthält eine Datei namens **logAnalyzer-daily.properties**, die wiederum die für diese Site spezifischen Einstellungen enthält.

Die Protokolldateien jeder Site werden im Verzeichnis **siten/app-server/logs/** abgelegt.

Das Verzeichnis **sites** ist schreibgeschützt. Ausgaben werden in das Verzeichnis **siten/app-server/logAnalyzer-out.d** geschrieben.

Mithilfe der oben genannten Struktur können Sie einen CRON-Job nur durch Angabe des Sitenamens starten. Beispiel für eine Site namens "bbfb-01":

```
# Tell Ant to set the site.name and use a build script name:
# logAnalyzer-daily.xml
ant -Dsite.name=bbfb-01 -f logAnalyzer-daily.xml
```

Wenn Sie **logAnalyzer-daily.xml** in **build.xml** umbenennen, können Sie das Argument **-f logAnalyzer-daily.xml** überspringen. Beispiel für eine Site namens "bbfb-01":

```
ant -Dsite.name=bbfb-01
```

Log Analyzer konfigurieren

Die systemspezifischen Einstellungen werden in einer Eigenschaftendatei festgelegt. Die Standardeigenschaftendatei lautet **sites/\${site.name}/logAnalyzer-daily.properties**. Sie können den Namen der Eigenschaftendatei auch wie folgt über die Befehlszeile festlegen:

```
ant -Dproperties.file.name="pfad_zu_datei.properties" ...
```

Nachfolgend sind die Konfigurationseigenschaften in der Datei **logAnalyzer-daily.properties** aufgeführt.

- **log.dir**: Der vollständige Pfad zur Position des Verzeichnisses mit den DEBS-Protokolldateien. Beispiel:

```
# default is ./logs
log.dir=/home/h1e/tomcat/logs
```
- **out.dir**: Das Ausgabeverzeichnis, in das die erstellten Berichte geschrieben werden. Beispiel:

```
# default is logAnalyzer-out.d
out.dir=/home/h1e/public_html/logAnalyzer-out.d
```
- **logAnalyzer.jar**: Die Position der Datei **.jar**-Datei in Log Analyzer. Beispiel:

```
# default is ./logAnalyzer-1.1.1-SNAPSHOT-app.jar
logAnalyzer.jar=target/logAnalyzer-1.1.1-SNAPSHOT-app.jar
```
- **is.weblogic**: Wird auf "true" gesetzt, wenn die Protokolldateien von WebLogic erstellt wurden. Beispiel:

```
# default is false
is.weblogic=true
```
- **genChart.perMessageType**: Wird auf "false" gesetzt, wenn die messageType-Diagrammerstellung übersprungen werden soll. Beispiel:

```
# default is true
genChart.perMessageType=false
```

- `log.prefix`: Das DEBS-Protokollpräfix. Dieser Wert muss nur selten geändert werden. Beispiel:

```
# default is debs.log  
log.prefix=Midwest.log
```
- `target.date.offset`: Automatische Festlegung des Zieldatums. Der Standardwert lautet 1; dies steht für gestern. Beispiel: Geben Sie für "target.date.offset" den Wert 7 an, um Protokolldateien von vor einer Woche zu extrahieren:

```
# default is yesterday: 1  
target.date.offset=7
```
- `target.date`: Einschränkung der Verarbeitung auf Protokolleinträge für diesen Tag. Die wahrscheinlichste Verwendung für diese Einstellung ist die manuelle Neugenerierung einer alten Gruppe von Protokolldateien. Beispiel:

```
# default is yesterday (auto-evaluated)  
target.date=2006/07/24
```

Index

Sonderzeichen

-Xmx, Einstellung 33

A

Archivieren von Daten 27

D

Daten

schützen 3

Zeitplan für Wiederherstellung 3

Datenarchivierung 27

debs.log, Analyse 35

Deploymentarchitektur 1

Bereitstellungsbereich 2

Buildumgebung 1

Qualitätssicherungsbereich 1

E

Eigenschaften 12

an der richtigen Position
definieren 25

Eigenschaftsnamen 13

I

Infrastruktur

Anwendungsschicht 2

Datenbankschicht 2

typisch 2

Webschicht 2

J

JVM

-server 32

-verbose

gc 32

-Xincgc 32

L

Leistung

Speicherfehler 33

Speicherprobleme 31

Leistungsoptimierung

Garbage-Collection 32

Log Analyzer, Tool 35

Eigenschaftendatei 38

Verzeichnisstruktur 37

Log Analyzer-Berichte automatisieren 36

M

Modellgröße 8

R

Redundanz 2

S

Schrittweise Garbage-Collection

-Xincgc 33

Sicherungen

Gesamtsicherung 3

inkrementelle Sicherung 3

Prüfpunktsicherung 3

tägliche Sicherung 3

wöchentliche Sicherung 3

Sicherungs- und

Wiederherstellungsstrategien 3

Speicherzuordnung

zu prüfende Bereiche 33

Statistikdaten aktualisieren

Oracle 29

SQL Server 29

Strategien

Sicherung und Wiederherstellung 3

U

Untermodelle 9

W

Wesentliche Leistungsindikatoren

(KPIs) 35

Wiederherstellungsrichtlinien 4

Wiederherstellungsszenario 4

Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte von IBM verletzen. Die Verantwortung für den Betrieb von Produkten, Programmen und Services anderer Anbieter liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Director of Licensing

IBM Europe, Middle East & Africa

Tour Descartes 2, avenue Gambetta 92066 Paris La Defense

France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die hier enthaltenen Informationen werden in regelmäßigen Zeitabständen aktualisiert und als Neuauflage veröffentlicht. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an den Hersteller.

IBM Corporation

J46A/G4

555 Bailey Avenue

San Jose, CA 95141-1003

USA

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des in diesen Informationen beschriebenen Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer kontrollierten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können davon abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Aussagen über Pläne und Absichten von IBM unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Alle von IBM angegebenen Preise sind empfohlene Richtpreise und können jederzeit ohne weitere Mitteilung geändert werden. Händlerpreise können u. U. von den hier genannten Preisen abweichen.

Diese Veröffentlichung dient nur zu Planungszwecken. Die in dieser Veröffentlichung enthaltenen Informationen können geändert werden, bevor die beschriebenen Produkte verfügbar sind.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufs. Sie sollen nur die Funktionen des Lizenzprogramms illustrieren und können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Musteranwendungsprogramme, die in Quellsprache geschrieben sind und Programmier Techniken in verschiedenen Betriebsumgebungen veranschaulichen. Sie dürfen diese Musterprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle für die Betriebsumgebung konform sind, für die diese Musterprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten. Die Musterprogramme werden ohne Wartung (auf "as-is"-Basis) und ohne jegliche Gewährleistung zur Verfügung gestellt. IBM übernimmt keine Haftung für Schäden, die durch die Verwendung der Musterprogramme entstehen.

Kopien oder Teile der Musterprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

© IBM 2011. Teile des vorliegenden Codes wurden aus Musterprogrammen der IBM Corp. abgeleitet. © Copyright IBM Corp. 2011.

Wird dieses Buch als Softcopy (Book) angezeigt, erscheinen keine Fotografien oder Farbabbildungen.

Marken

IBM, das IBM Logo und ibm.com sind Marken oder eingetragene Marken der IBM Corporation in den USA und/oder anderen Ländern. Weitere Produkt- und Servicenamen können Marken von IBM oder anderen Unternehmen sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Webseite "Copyright and trademark information" unter <http://www.ibm.com/legal/copytrade.shtml>.

Adobe, das Adobe Logo, PostScript und das PostScript-Logo sind Marken oder eingetragene Marken der Adobe Systems Incorporated in den USA und/oder anderen Ländern.

IT Infrastructure Library ist eine eingetragene Marke der Central Computer and Telecommunications Agency. Die Central Computer and Telecommunications Agency ist nunmehr in das Office of Government Commerce eingegliedert worden.

Intel, das Intel Logo, Intel Inside, das Intel Inside Logo, Intel Centrino, das Intel Centrino Logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium und Pentium sind Marken der Intel Corporation oder ihrer Tochtergesellschaften in den USA und anderen Ländern.

Linux ist eine eingetragene Marke von Linus Torvalds in den USA und/oder anderen Ländern.

Microsoft, Windows, Windows NT und das Windows Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

ITIL ist als eingetragene Marke und eingetragene Gemeinschaftsmarke des Office of Government Commerce beim US Patent and Trademark Office registriert.

UNIX ist eine eingetragene Marke von The Open Group in den USA und anderen Ländern.

Java und alle auf Java basierenden Marken und Logos sind Marken oder eingetragene Marken der Oracle Corporation und/oder ihrer verbundenen Unternehmen.

Cell Broadband Engine wird unter Lizenz verwendet und ist eine Marke der Sony Computer Entertainment, Inc. in den USA und/oder anderen Ländern.

Linear Tape-Open, LTO, das LTO Logo, Ultrium und das Ultrium Logo sind Marken von HP, der IBM Corp. und Quantum in den USA und anderen Ländern.

Connect Control Center[®], Connect:Direct[®], Connect:Enterprise, Gentran[®], Gentran:Basic[®], Gentran:Control[®], Gentran:Director[®], Gentran:Plus[®], Gentran:Realtime[®], Gentran:Server[®], Gentran:Viewpoint[®], Sterling Commerce[™], Sterling Information Broker[®] und Sterling Integrator[®] sind Marken oder eingetragene Marken der Sterling Commerce, Inc., einem IBM Unternehmen.

Weitere Unternehmens-, Produkt- oder Servicenamen können Marken anderer Hersteller sein.



Gedruckt in Deutschland