

ORACLE

Hardware and Software

ORACLE®

Engineered to Work Together




ORACLE®

Validierung der Dateneingabe mit Oracle ADF

Jürgen Menge

Oracle Deutschland B.V. & Co. KG



The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decision. The development, release, and timing of any features or functionality described for oracle's products remains at the sole discretion of oracle.

Agenda

- Einführung
- Validierung im Business Service
- Validierung im Binding Layer
- Validierung in ADF Faces
- Zusätzliche Informationen



Validierung der Dateneingaben

- Validierung ist notwendiger Bestandteil aller Unternehmens-Applikationen
- Notwendige Validierung durch intelligente Benutzerführung verringern (z.B. Wertelisten anstelle von Textfeldern)
- Validierungen müssen teilweise mehrfach in verschiedenen Schichten der Applikation implementiert werden
 - Konsistenz der Datenbank vs. ergonomische Benutzerführung



Motto

Java EE / Oracle ADF ist nicht Forms !!!

Validierung in Oracle Forms und ADF

- Übersicht „ADF Equivalents of Common Oracle Forms Triggers“
aber

Besonderheiten von Java EE/ADF-Applikationen

- Zahlreiche Schichten der Applikation
- Request/Response-Modell im Web

- Zeitpunkte der Validierung

Ebene der Applikation	Zeitpunkt
Benutzereingabe auf dem Client	JavaScript Events
Application Server-seitige Validierung	Submit
Datenbank-seitige Validierung	Post / Commit

Validierungsebenen in ADF Fusion Applikationen

Ebene	Zweck	Mittel	Zeitpunkt der Ausführung
ADF Faces client-seitig	Frühestmögliche Reaktion auf fehlerhafte Eingaben	Validierungs-Eigenschaften, Default und Custom Validatoren (JavaScript); explizite Validierung mittels JavaScript	Client-seitige Events
ADF Faces server-seitig	Abfangen fehlerhafter Eingaben vor der Übermittlung an den Business Service	Validierungs-Attribute, Default und Custom Validatoren (Java)	Submit
ADF Binding Layer	Abfangen fehlerhafter Eingaben vor der Übermittlung an den Business Service (z.B. Web Service)	Deklarative Validierung von Binding-Attributen im Pagedef-File	Submit
ADF Business Components	Abfangen fehlerhafter Eingaben vor der Übermittlung an die Datenbank	Deklarative Validierungen (darunter Regular Expressions, Groovy Scripts) Programmatische Validierungen (Java)	Submit
Datenbank	Gewährleistung der Integrität und Konsistenz der Daten	DB Constraints, DB Trigger (PL/SQL)	Post / Commit



Validierung im Business Service (ADFbc)

Validierung im Business Service (ADFbc)

- Wo kann validiert werden ?
 - Validierung beim Entity Object
 - Validierung beim EO-Attribut
 - Validierung auf Domain-Ebene
 - Validierung gegen View Accessor (Model-driven LOV)
- Wie kann validiert werden ?
 - deklarative Validierung (Validatoren)
 - große Zahl deklarativer Validatoren
 - es können eigene Validatoren hinzugefügt werden
 - Regular Expressions
 - Groovy Scripte
 - programmatische Validierung (Custom Methods)

Validierung im Business Service (ADFbc)

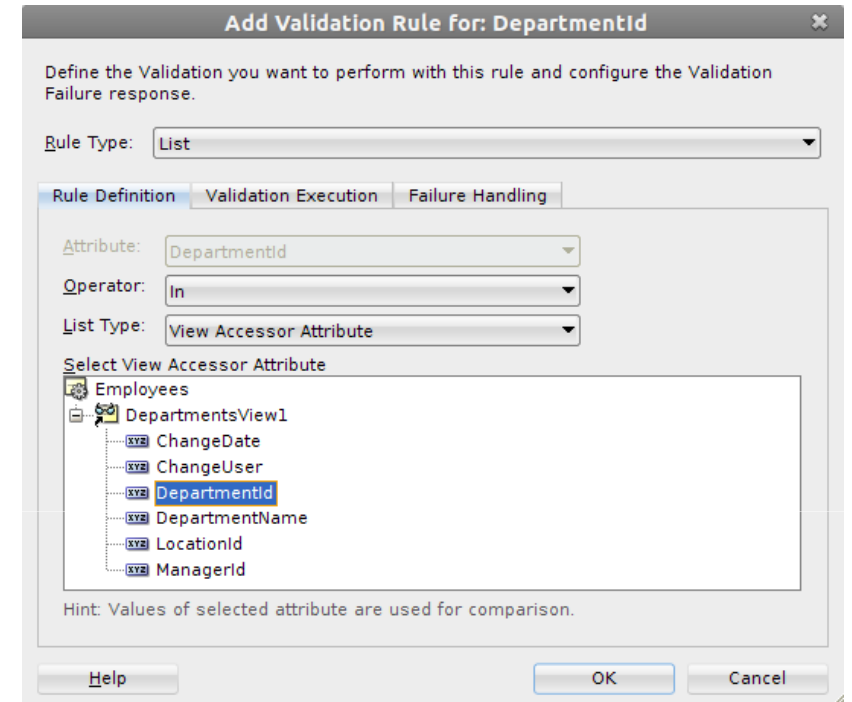
- Vorteile
 - Validierung aller Pages, die auf dem Model basieren (EO)
 - zahlreiche Möglichkeiten der Validierung auf deklarativem Weg
- Einschränkungen
 - Validierung von Suchkriterien (z.B. Suche in einem Zeitraum)

Deklarative Validatoren in ADFbc

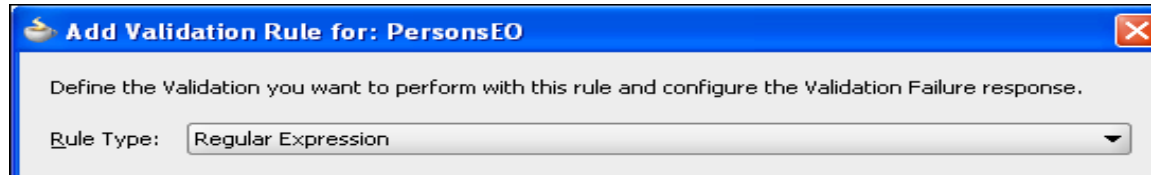
Validator	Zuordnung	Charakteristik
Collection Validator	Entity Object (EO)	Gruppenoperationen auf Attributen (Sum, Avg, etc.)
Unique Key Validator	Entity Object (EO)	Prüfung auf eindeutige Schlüssel
Compare Validator	EO, EO Attribut	Vergleich von Werten
Key Exists Validator	EO, EO Attribut	Prüfung auf existierenden Schlüssel
Length Validator	EO, EO Attribut	Prüfung der Länge
List Validator	EO, EO Attribut	Prüfung auf Existenz eines Wertes in einer Liste
Range Validator	EO, EO Attribut	Prüfung, ob Wert in einem Bereich liegt
Regular Expression Validator	EO, EO Attribut	Package java.util.regex
Script Validator	EO, EO Attribut	Verwendung von Groovy-Skripten

List Validator

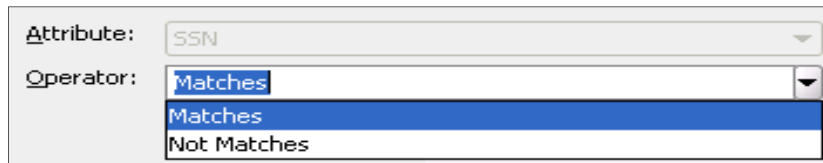
- Prüfung auf Existenz eines Wertes in einer Liste
- Typ von Liste
 - Statische Werte
 - Query
 - Attribut eines View Objects
 - Attribut eines View Accessors \Rightarrow Validierung gegen LOV



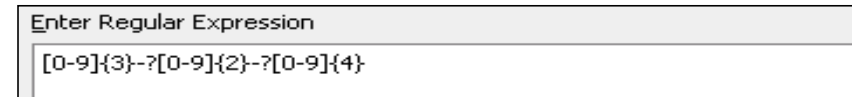
Regular Expression Validator



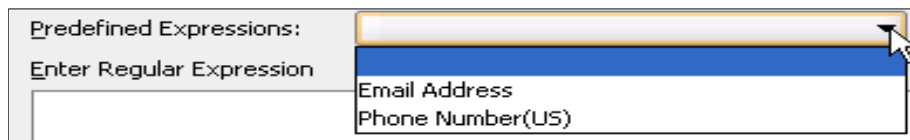
- Operator



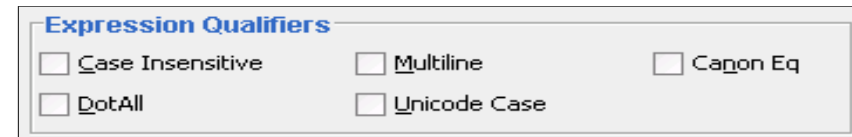
- Regular Expression



- Vordefinierte Expressions



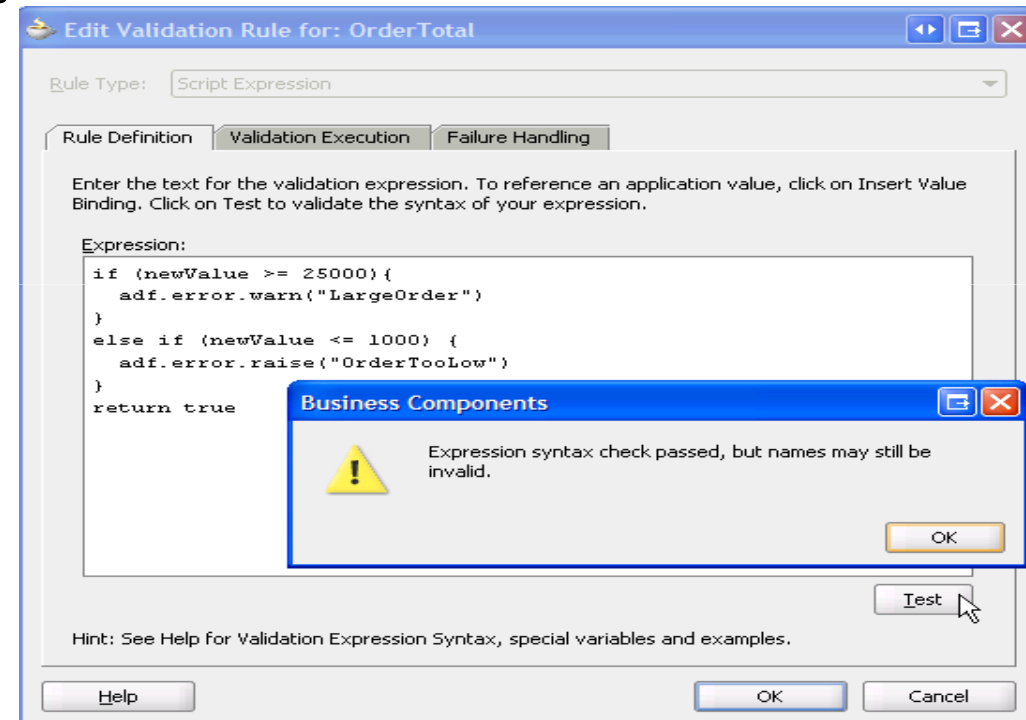
- Qualifier



- Syntax gemäß *java.util.regex* (ab JDK 1.4)
- Tipp: Verwendung eines Regex Testers (<http://www.regextester.com>)

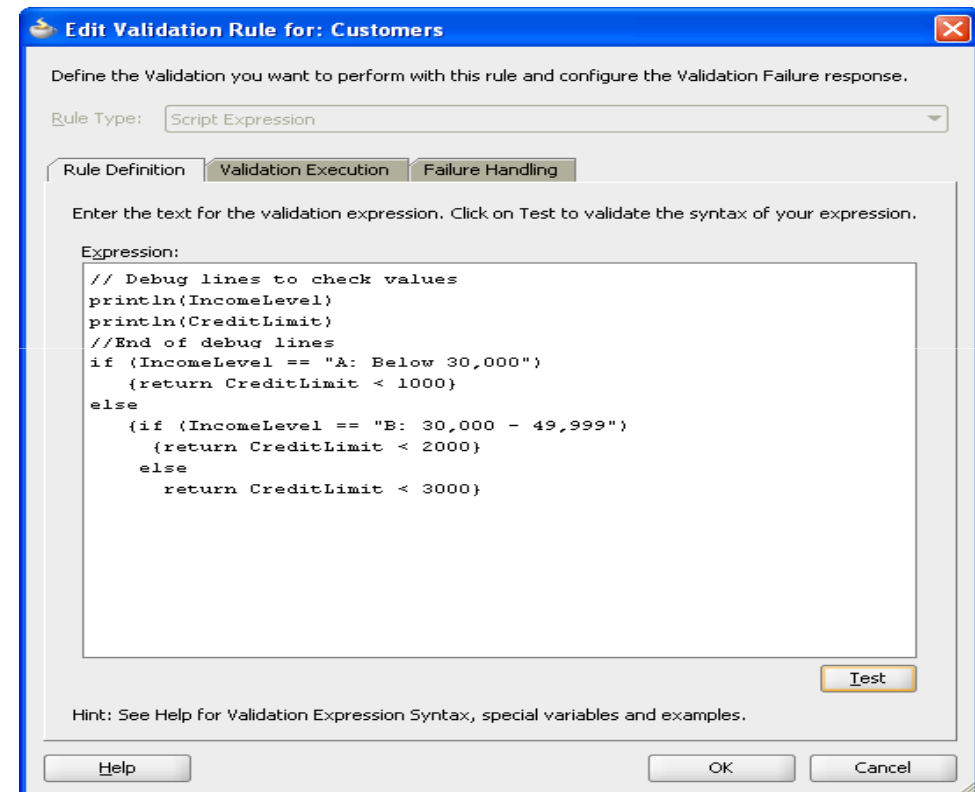
Script Expression Validator

- Verwendung von Groovy Expressions
 - nicht in Klammern einschließen { }
 - newValue für neuen Wert des ausgewählten Attributs
 - true oder false zurückgeben bzw. `adf.error.raise` oder `adf.error.warn` aufrufen
 - kann Gruppenfunktionen auf dem RowSet ausführen
 - Test der Syntax möglich



Verwendung von Groovy

- von Java abgeleitete Scripting-Sprache
 - Übersetzung in Byte Code
 - Ausführung innerhalb der JVM
 - Zugriff auf Java APIs
 - Zugriff auf eigene Java-Klassen
- vereinfachte Syntax
- Compilierung zur Laufzeit
 - Speicherung als Metadaten
 - Anpassungen vor der Ausführung möglich (Customizing)

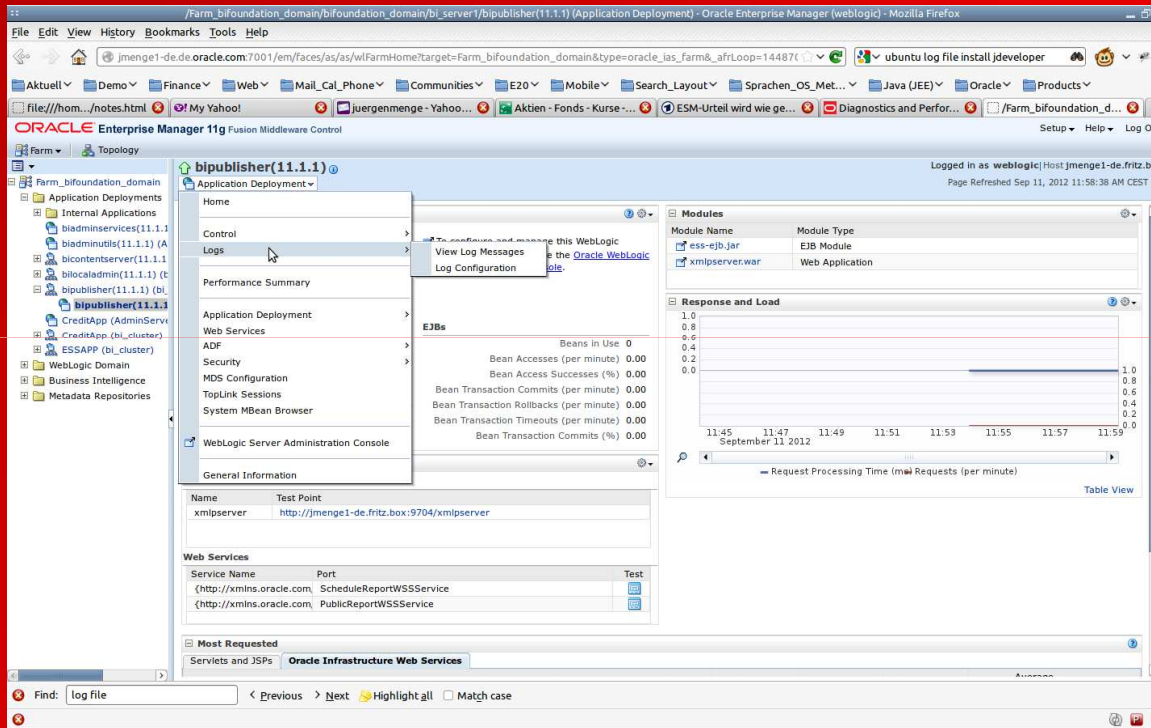


Programmatische Validierung

- Java-Methode im Entity Object (Method Validator)
 - wird automatisch im Validierungszyklus aufgerufen
 - Voraussetzungen
 - Methode muss als `public` deklariert sein
 - Methode muss `boolean` zurückgeben
 - Namenskonvention `validateXXX()`
- Erweiterung der deklarativen Validatoren (Custom Validation Rule)
 - Java Class im Package `oracle.jbo.rules`
 - Registrierung im Projekt oder im IDE

```
public boolean validateOrder(){
    if( // add your validation code )
        return true;
    else
        return false;
}
```

DEMONSTRATION



The screenshot displays the Oracle Enterprise Manager 11g Fusion Middleware Control interface. The main content area shows the configuration for the 'bipublisher(11.1.1)' application deployment. A context menu is open over the 'Logs' link, with options for 'View Log Messages' and 'Log Configuration'. The 'Response and Load' section features a line graph showing request processing time and requests per minute over a period of time in September 2012. The graph shows a steady increase in activity starting around 11:45.

Module Name	Module Type
ess-ejb.jar	EJB Module
xmlpserver.war	Web Application

Service Name	Port	Test
{http://xmlns.oracle.com/ScheduleReportWSSService}		[Test]
{http://xmlns.oracle.com/PublicReportWSSService}		[Test]



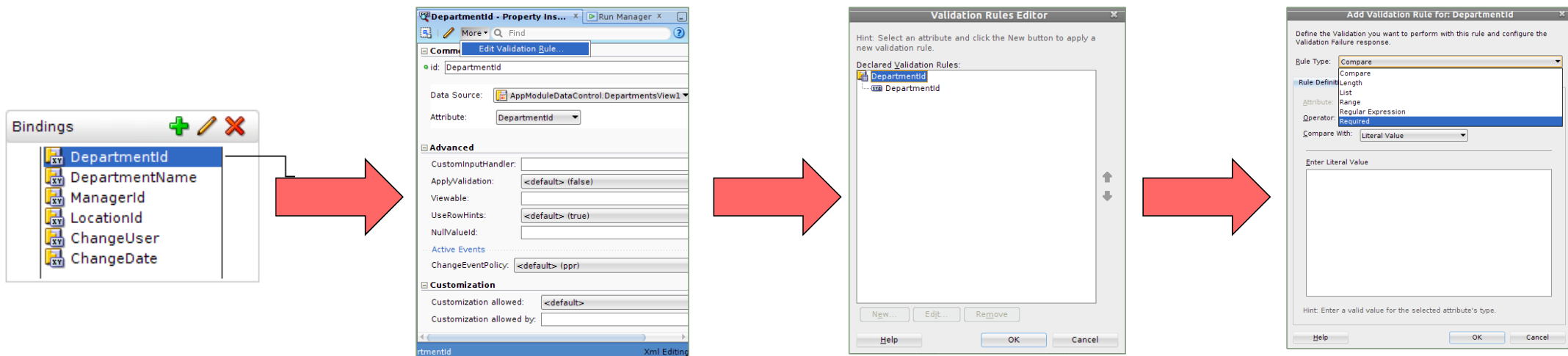
Validierung im Binding Layer

Validierung im Binding Layer (Model)

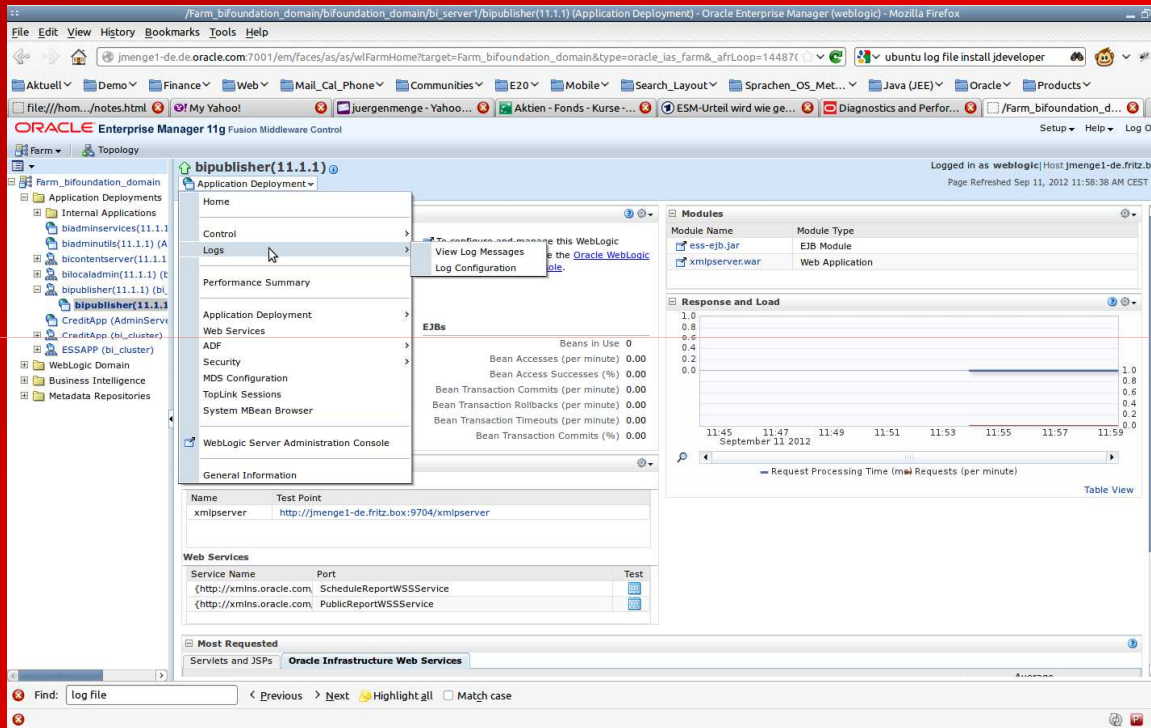
- Anwendungsfälle
 - keine deklarative Validierung im Business Service möglich
 - Validierung vor einem Network Roundtrip (z.b. vor Aufruf eines Web Service)
 - Validierung von Parametern einer Parameter-Form vor der Ausführung

Validierung im Binding Layer (Model)

- Vorgehen
 - *pagedef*-Datei öffnen
 - Attribute Binding auswählen
 - im Property Inspector *More* ⇒ *Edit Validation Rule* auswählen
 - Validierungsregel anlegen (Compare, Length, List, Range, RegEx, Required)



DEMONSTRATION



The screenshot displays the Oracle Enterprise Manager 11g Fusion Middleware Control interface. The browser address bar shows the URL: `/Farm_bifoundation_domain/bifoundation_domain/bi_server1/bipublisher(11.1.1) (Application Deployment) - Oracle Enterprise Manager (weblogic) - Mozilla Firefox`. The page title is "ORACLE Enterprise Manager 11g Fusion Middleware Control".

The main content area shows the configuration for the **bipublisher(11.1.1)** application deployment. A context menu is open over the "Logs" link, with options: "View Log Messages" and "Log Configuration".

The "Response and Load" section contains a line graph showing "Request Processing Time (min) Requests (per minute)" for September 11, 2012. The graph shows a steady increase in request processing time over the period.

Module Name	Module Type
ess-ejb.jar	EJB Module
xmlpserver.war	Web Application

Service Name	Port	Test
{http://xmlns.oracle.com/ScheduleReportWSSService}		
{http://xmlns.oracle.com/PublicReportWSSService}		



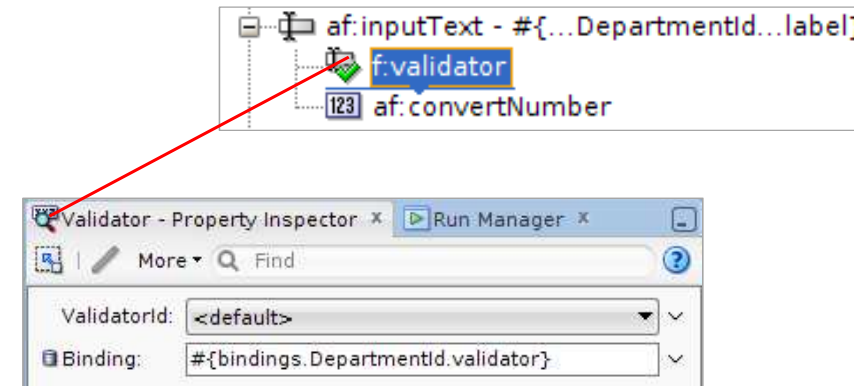
Validierung in ADF Faces

Validierung in ADF Faces

- Möglichkeiten der Validierung
 - Validierungs-Eigenschaften (*Required*, ...)
 - Default Validatoren (JSF)
 - Default Validatoren (ADF Faces)
 - zusätzliche client-seitige Prüfungen (JavaScript)
 - zusätzliche Eigenschaften
 - Custom Validatoren
 - server-seitige Prüfung (Java Class)
 - optionale client-seitige Prüfung (JavaScript)
- Validatoren werden in der Reihenfolge der Zuordnung ausgeführt

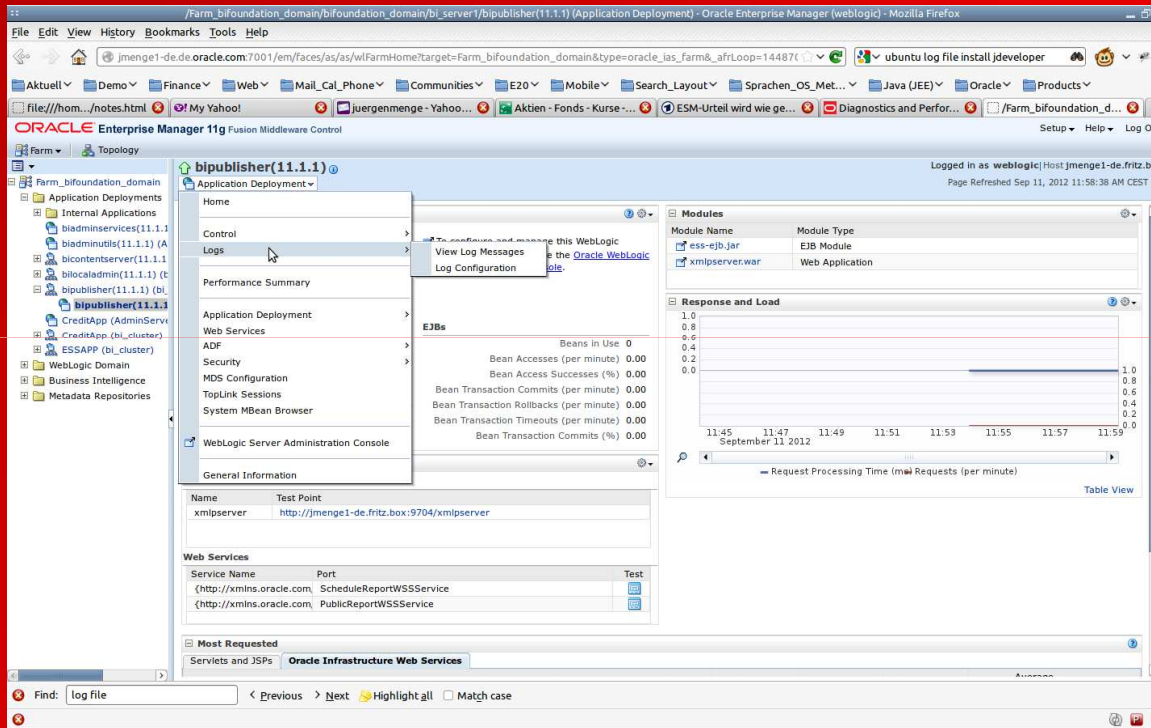
Default Validatoren (JSF)

- nur server-seitige Validierung (Submit)
- f:validator mit Verweis auf Binding wird für jede Input-Komponente angelegt



Validator	Syntax	Charakteristik
Validator	f:validator	Validator mit Binding an ein Objekt, das das Interface javax.faces.validator.Validator implementiert
Validate Bean	f:validateBean	Delegiert die Validierung des Wertes an das Bean Validation API (JSF 2.0)
Validate Double Range	f:validateDoubleRange	Prüfung, ob Zahl vom Typ Double innerhalb eines Bereiches liegt
Validate Length	f:validateLength	Prüfung, ob Länge eines Strings innerhalb eines Bereiches liegt
Validate Long Range	f:validateLongRange	Prüfung, ob Zahl vom Typ Long innerhalb eines Bereiches liegt
Validate Regex	f:validateRegex	Prüfung auf Regular Expression (java.util.regex)
Validate Required	f:validateRequired	Prüfung der Vorhandensein eines Wertes

DEMONSTRATION

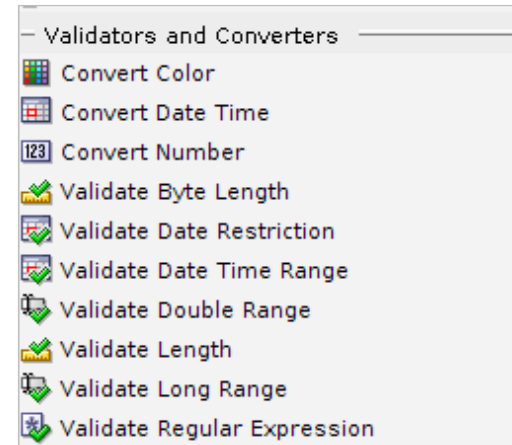


The screenshot displays the Oracle Enterprise Manager 11g Fusion Middleware Control interface. The main window shows the configuration for the 'bipublisher(11.1.1)' application deployment. The 'Logs' menu is open, showing options like 'View Log Messages' and 'Log Configuration'. The 'Response and Load' section features a line graph showing request processing time and requests per minute over a period of time. The graph shows a steady increase in requests per minute starting around 11:45 AM on September 11, 2012.

Time	Request Processing Time (min)	Requests (per minute)
11:45	~0.2	~0.2
11:47	~0.2	~0.2
11:49	~0.2	~0.2
11:51	~0.2	~0.2
11:53	~0.2	~0.2
11:55	~0.2	~0.2
11:57	~0.2	~0.2
11:59	~0.2	~0.2

Default Validatoren (ADF Faces)

- fügen eine client-seitige Validierung hinzu
- zusätzliche Eigenschaften (Hints, Fehlermeldungen, ...)



Validator	Syntax	Charakteristik
Validate Byte Length	af:validateByteLength	Prüfung auf max. zulässige Länge in Bytes
Validate Date Restriction	af:validateDateRestriction	Prüfung auf zulässige Tage und Monate
Validate Date Time Range	af:validateDateTimeRange	Prüfung auf zulässiges Zeitintervall
Validate Double Range	af:validateDoubleRange	Prüfung, ob Zahl vom Typ Double innerhalb eines Bereiches liegt
Validate Length	af:validateLength	Prüfung, ob Länge eines Strings innerhalb eines Bereiches liegt
Validate Long Range	af:validateLongRange	Prüfung, ob Zahl vom Typ Long innerhalb eines Bereiches liegt
Validate Regular Expression	af:validateRegExp	Prüfung auf Regular Expression (java.util.regex)

Custom Validatoren

- Schreiben eigener Validierungen
- für einzelne Page
 - Backing Bean mit Validierungslogik
- für mehrere Pages und/oder zusätzliche client-seitige Validierung
 - JSF validator class (muss *javax.faces.validator.Validator* implementieren)
 - Registrierung in der *faces-config.xml*
 - Erstellen eines eigenen Tags oder Verwenden des *f:validator* und ID im Property *Binding* referenzieren

Server-seitige Validierung erzwingen

EmployeeId	FirstName	LastName
>yeeld.inputValue}	tName.inputValue}	tName.inputValue}
>yeeld.inputValue}	tName.inputValue}	tName.inputValue}
>yeeld.inputValue}	tName.inputValue}	tName.inputValue}

AutoSubmit: <default> (false)

autoSubmit=true (für Input-Komponenten)

- wenn eine Aktion stattfindet (z.B. Änderung des Wertes), wird die äußere Form von der Komponente abgeschickt (submit)
- für Komponenten, die Werte enthalten (z.B. Input Text oder Checkbox), werden so die Änderungen zum frühestmöglichen Zeitpunkt zum Server gesendet (z.B. bei Verlassen der Komponente oder einer Navigation)

Client-seitige Validierung

- Implementierung in JavaScript
 - Validierung erfolgt auf dem Client (client-seitiger Request Life Cycle)
 - teilweise einfachere Implementierung von komplexen Prüfungen
 - bei auftretenden Fehlern wird Page nicht submitted
 - Vor- und Nachteile von JavaScript
- Möglichkeiten der Implementierung
 - implizit durch Verwendung der ADF Faces Validatoren
 - explizit durch eigene Implementierung eines Custom Validators (ADF Faces JavaScript API)



Verwandte Themen



Verwandte Themen

- JSF / ADF Faces Converter
- Fehlermeldungen
- Exception Handling

Weiterführende Informationen zur Validierung

- Fusion Developer's Guide for Oracle Application Development Framework - 11.1.2.3.0
 - ADF Business Components Kapitel 7 und 8
 - ADF Model Kapitel 16
- Web User Interface Developer's Guide for Oracle Application Development Framework - 11.1.2.2.0
 - Converter und Validatoren Kapitel 7

Deutsche Oracle ADF Community

- Kunden und Partner, die das Oracle Application Development Framework einsetzen (wollen)

- Oracle ADF Community auf XING
- Twitter: ADFCommunityDE



- regelmäßige Treffen
- News Sessions (30-minütige Web Seminare an jedem 2. Freitag, 8.30 Uhr)
- Project Sessions (Serie von aufeinander aufbauenden ADF-Workshops)
- Hackers Event am 24. und 25. Januar 2013 in München
- Zusammenarbeit mit der DOAG (Jahreskonferenz, SIG)
- Meeting am 22.11., 11.00 Uhr im Raum Sidney

Oracle Kundenumfrage



Nehmen Sie an unserer Umfrage teil und **gewinnen Sie ein Apple iPad!**

Fragebögen am Oracle Stand oder im Smartphone ausfüllen. Einfach den QR-Code scannen.

<http://apex.oracle.com/pls/apex/f?p=201211>

ORACLE

Fragen & Antworten

Jürgen Menge

jürgen.menge@oracle.com



ORACLE®

Hardware and Software

ORACLE®

Engineered to Work Together

ORACLE®

ORACLE®