

Tastaturbedienung in Web Dynpro for ABAP / for Java



Release SAP NW 7.0



Copyright

© Copyright 2008 SAP AG. Alle Rechte vorbehalten.

Weitergabe und Vervielfältigung dieser Publikation oder von Teilen daraus sind, zu welchem Zweck und in welcher Form auch immer, ohne die ausdrückliche schriftliche Genehmigung durch SAP AG nicht gestattet. In dieser Publikation enthaltene Informationen können ohne vorherige Ankündigung geändert werden.

Die von SAP AG oder deren Vertriebsfirmen angebotenen Softwareprodukte können Softwarekomponenten auch anderer Softwarehersteller enthalten.

Microsoft, Windows, Outlook, und PowerPoint sind eingetragene Marken der Microsoft Corporation. IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, OpenPower und PowerPC sind Marken oder eingetragene Marken der IBM Corporation.

Adobe, das Adobe Logo, Acrobat, PostScript und Reader sind Marken oder eingetragene Marken von Adobe Systems Inc. in den USA und/oder anderen Ländern.

Oracle ist eine eingetragene Marke der Oracle Corporation.

UNIX, X/Open, OSF/1, und Motif sind eingetragene Marken der Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, und MultiWin sind Marken oder eingetragene Marken von Citrix Systems, Inc.

HTML, XML, XHTML und W3C sind Marken oder eingetragene Marken des W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java ist eine eingetragene Marke von Sun Microsystems, Inc.

JavaScript ist eine eingetragene Marke der Sun Microsystems, Inc., verwendet unter der Lizenz der von Netscape entwickelten und implementierten Technologie.

MaxDB ist eine Marke von MySQL AB, Schweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, und weitere im Text erwähnte SAP-Produkte und -Dienstleistungen sowie die entsprechenden Logos sind Marken oder eingetragene Marken der SAP AG in Deutschland und anderen Ländern weltweit. Alle anderen Namen von Produkten und Dienstleistungen sind Marken der jeweiligen Firmen. Die Angaben im Text sind unverbindlich und dienen lediglich zu Informationszwecken. Produkte können länderspezifische Unterschiede aufweisen.

In dieser Publikation enthaltene Informationen können ohne vorherige Ankündigung geändert werden. Die vorliegenden Angaben werden von SAP AG und ihren Konzernunternehmen („SAP-Konzern“) bereitgestellt und dienen ausschließlich Informationszwecken. Der SAP-Konzern übernimmt keinerlei Haftung oder Garantie für Fehler oder Unvollständigkeiten in dieser Publikation. Der SAP-Konzern übernimmt lediglich diejenige Garantie für Produkte und Dienstleistungen, die in den gegebenenfalls mit den jeweiligen Produkten und Dienstleistungen gelieferten ausdrücklichen Garantiererklärungen festgelegt sind. Aus den in dieser Publikation enthaltenen Informationen ergibt sich keine weiterführende Garantie.

Symbole im Text

SAP: Wichtige Haftungsausschlüsse und rechtliche Informationen

Dieses Dokument dient nur zu Informationszwecken. Die darin enthaltenen Informationen können ohne vorherige Ankündigung geändert werden. SAP gewährleistet nicht, dass sie fehlerfrei sind. SAP SCHLIESST JEDE GEWÄHRLEISTUNG AUS, SEI SIE AUSDRÜCKLICH ODER KONKLUDENT, EINSCHLIESSLICH DER GARANTIE DER HANDELSÜBLICHKEIT ODER EIGNUNG FÜR EINEN BESTIMMTEN ZWECK.

Coding-Beispiele






Bei dem in der vorliegenden Dokumentation enthaltenen Quell- und/oder Objektcode für Software ("Code") handelt es sich ausschließlich um eine beispielhafte Darstellung. Dieser Code ist in keinem Fall für die Nutzung in einem produktiven System geeignet. Der Code dient ausschließlich dem Zweck, beispielhaft aufzuzeigen, wie Quelltext erstellt und gestaltet werden kann. SAP übernimmt keine Gewährleistung für die Funktionsfähigkeit, Richtigkeit und Vollständigkeit des hier abgebildeten Codes und SAP übernimmt keine Haftung für Schäden, die durch die Nutzung des Codes entstehen, sofern solche Schäden nicht durch vorsätzliches oder grob fahrlässiges Verhalten der SAP verursacht wurden.

Internet-Hyperlinks

Die SAP-Dokumentation kann Hyperlinks auf das Internet enthalten. Diese Hyperlinks dienen lediglich als Hinweis auf ergänzende und weiterführende Dokumentation. SAP übernimmt keine Gewährleistung für die Verfügbarkeit oder Richtigkeit dieser ergänzenden Information oder deren Nutzbarkeit für einen bestimmten Zweck. SAP übernimmt keine Haftung für Schäden, die durch die Nutzung solcher Informationen verursacht werden, es sei denn, dass diese Schäden von SAP grob fahrlässig oder vorsätzlich verursacht wurden.

Barrierefreiheit

Die in der Dokumentation der SAP-Bibliothek enthaltenen Informationen stellen Kriterien der Barrierefreiheit aus Sicht von SAP zum Zeitpunkt der Veröffentlichung dar und sollen keineswegs obligatorische Richtlinien sein, wie die Barrierefreiheit von Softwareprodukten zu gewährleisten ist. SAP lehnt insbesondere jede Haftung in Bezug auf dieses Dokument ab, aus dem weder direkt noch indirekt irgendwelche vertraglichen Verpflichtungen entstehen. Dieses Dokument dient nur dem internen Gebrauch und darf nicht ohne vorherige schriftliche Genehmigung von SAP außerhalb Ihres Unternehmens verbreitet werden.

Symbol	Bedeutung
	Achtung
	Beispiel
	Hinweis
	Empfehlung
	Syntax

In der SAP-Dokumentation werden weitere Symbole verwendet, die verdeutlichen, welche Art von Informationen ein Text enthält. Weitere Informationen finden Sie auf der Startseite jeder Version der SAP-Bibliothek unter *Hilfe zur Hilfe* → *Allgemeine Informationsklassen* und *Informationsklassen für das Business Information Warehouse*.

Typografische Konventionen

Format	Beschreibung
<i>Beispieltext</i>	Wörter oder Zeichen, die vom Bildschirmbild zitiert werden. Dazu gehören Feldbezeichner, Bildtitel, Drucktastenbezeichner sowie Menünamen, Menüpfade und Menüeinträge. Querverweise auf andere Dokumentationen
Beispieltext	hervorgehobene Wörter oder Ausdrücke im Fließtext, Titel von Grafiken und Tabellen
BEISPIELTEXT	Namen von Systemobjekten. Dazu gehören Reportnamen, Programmnamen, Transaktionscodes, Tabellennamen und einzelne Schlüsselbegriffe einer Programmiersprache, die von Fließtext umrahmt sind, wie z.B. SELECT und INCLUDE
Beispieltext	Ausgabe auf der Oberfläche. Dazu gehören Datei- und Verzeichnisnamen und ihre Pfade, Meldungen, Quelltext, Namen von Variablen und Parametern sowie Namen von Installations-, Upgrade- und Datenbankwerkzeugen.
Beispieltext	exakte Benutzereingabe. Dazu gehören Wörter oder Zeichen, die Sie genau so in das System eingeben, wie es in der Dokumentation angegeben ist.
<Beispieltext>	variable Benutzereingabe. Die Wörter und Zeichen in spitzen Klammern müssen Sie durch entsprechende Eingaben ersetzen, bevor Sie sie in das System eingeben.
BEISPIELTEXT	Tasten auf der Tastatur, wie z.B. die Funktionstaste F2 oder die ENTER-Taste

Tastaturbedienung von Oberflächenelementen in Web Dynpro.....	6
BreadCrumb	14
Button	15
CheckBox	15
CheckBoxGroup	15
DateNavigator.....	16
DropDownByIndex	17
DropDownByKey	17
Group.....	18
Image.....	19
InputField.....	19
ItemListBox.....	20
LinkToAction.....	20
LinkToURL.....	21
MenuBar	21
PhaseIndicator	21
RadioButton.....	22
RadioButtonGroupByIndex.....	22
RadioButtonGroupByKey	23
RoadMap	23
ScrollContainer	23
Table.....	24
TabStrip.....	30
TextEdit	31
TextView.....	31
ToggleButton	32
ToggleLink	32
Tray	32
Tree	33
TriStateCheckBox	35



Tastaturbedienung von Oberflächenelementen in Web Dynpro

Dieses Dokument beschreibt die Tastaturbedienung von Oberflächenelementen in Anwendungen, die im HTML-Client laufen und auf Web Dynpro for ABAP oder Web Dynpro for Java basieren. Die Tastaturbedienung ist für Sie relevant, wenn Sie mit Ihrer Tastatur auf der Anwendungsoberfläche navigieren und die zugehörigen Funktionen ausführen möchten.

Voraussetzungen

Das Dokument gilt ausschließlich für folgende Releases:

- SAP NetWeaver, Web Dynpro for ABAP 7.0, ab Support Package Stack (SPS) 14
- SAP NetWeaver, Web Dynpro for Java 7.0, ab SPS 14
- SAP NetWeaver, Web Dynpro for Java 7.01

Damit Ihnen alle hier dokumentierten Tastaturbefehle zur Verfügung stehen, müssen Sie den [barrierefreien Modus einschalten \[Extern\]](#). Viele Tastaturbefehle funktionieren zwar auch dann, wenn der barrierefreie Modus ausgeschaltet ist. Es gibt jedoch einige Tastaturbefehle, die nur im barrierefreien Modus funktionieren, wie z.B. die Gruppennavigation oder die Navigation bei inaktiven Oberflächenelementen.

Tastaturbefehle

Nachfolgend finden Sie eine Tabelle, die pro Oberflächenelement die möglichen Aktionen und die zugeordneten Tastaturbefehle enthält.



In der Tabelle sind die einzelnen Oberflächenelemente unter ihrem englischen Originalnamen aufgeführt. Die technische Bezeichnung wurde auf Grund ihrer Eindeutigkeit beibehalten. In Klammern finden Sie eine deutsche Entsprechung, welche jedoch keine allgemeingültige Bezeichnung des Oberflächenelements darstellt. Jedes Oberflächenelement in der Tabelle ist mit einem Link versehen, der Sie zur Dokumentation des Oberflächenelements in Web Dynpro for ABAP führt. (Um den Umfang dieser Datei minimal zu halten, wurden die für die Tastaturbedienung nicht relevanten, technischen Informationen zu den Oberflächenelementen weggelassen.)

In diesem Dokument bedeutet die Formulierung *Fokusnavigation vorwärts*, dass Sie das Fokusrechteck innerhalb der Tabulatorreihenfolge vorwärts zum Element und weiter zum nächsten Element bewegen können. Die Formulierung *Fokusnavigation rückwärts* bedeutet, dass Sie das Fokusrechteck innerhalb der Tabulatorreihenfolge rückwärts zum Element und weiter zum vorigen Element bewegen können.



Beispiel: Beim Oberflächenelement *Button* (Schaltfläche) können Sie das Fokusrechteck mit der *Tabulatortaste* vorwärts zum *Button* navigieren. Wenn der *Button* den Fokus hat, können Sie die beim jeweiligen *Button* hinterlegte Funktion mit der *Eingabetaste* oder der *Leertaste* auslösen. Mit der Tastenkombination *Umschalttaste* + *Tabulatortaste* können Sie das Fokusrechteck rückwärts vom *Button* weg navigieren.

Tabellarische Übersicht der Tastaturbedienung in Web Dynpro

Oberflächenelement	Aktion	Tastaturbefehl
BreadCrumb [Seite 14] (Navigationspfad)	Fokusnavigation vorwärts Der Fokus befindet sich dann auf dem aktuell ausgewählten Unterelement.	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Zwischen den Unterelementen des Elements navigieren	Pfeiltasten-nach-rechts/-links
	Fokussiertes Unterelement auslösen	Leertaste
Button [Seite 15] (Schaltfläche)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Fokussiertes Element auslösen	Eingabetaste oder Leertaste
	Element mit Standardaktivierung [Extern] auslösen (Element wird ausgelöst, auch wenn es nicht fokussiert ist)	Eingabetaste
CheckBox [Seite 15] (Ankreuzfeld) CheckBoxGroup [Seite 15] (Gruppe aus Ankreuzfeldern)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Zwischen den einzelnen Checkbox-Elementen navigieren	Tabulatortaste
	Fokussierte Checkbox aktivieren (Kennzeichen setzen oder entfernen)	Leertaste
DateNavigator [Seite 16] (Datumsauswahl)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Fokussiertes Element aktivieren	Leertaste
	Zwischen den Unterelementen (den einzelnen Tagen) navigieren	Pfeiltasten
	Fokussiertes Unterelement auswählen	Leertaste
DropDownByIndex [Seite 17] DropDownByKey [Seite 17] (Dropdown-Box)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Im Listenfeld zwischen den	Pfeiltasten-nach-

Oberflächenelement	Aktion	Tastaturbefehl
	Einträgen navigieren	oben/-unten
	Zum ersten Eintrag der Liste navigieren	Strg + Pos1-Taste
	Zum letzten Eintrag der Liste navigieren	Strg + Ende-Taste
	Selektierten Eintrag der Liste auswählen	Eingabetaste
	Listefeld öffnen/schließen	Alt + Pfeil-nach-unten/-oben oder F4-Taste
Group [Seite 18] (Gruppe)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Gruppennavigation vorwärts	F6
Image [Seite 19] (Bild)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
InputField [Seite 19] (Eingabefeld)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Innerhalb des Eingabebereichs navigieren	Pfeiltasten-nach-rechts/-links
	Innerhalb des Eingabebereichs den Textcursor vor das erste Zeichen setzen	Pos1-Taste
	Innerhalb des Eingabebereichs den Textcursor hinter das letzte Zeichen setzen	Ende-Taste
	Innerhalb des Eingabebereichs zum nächsten Wort navigieren	Strg + Pfeiltaste-nach-rechts
	Innerhalb des Eingabebereichs zum vorigen Wort navigieren	Strg + Pfeiltaste-nach-links
	Vom Textcursor ausgehend das nächste Zeichen markieren	Umschalt + Pfeiltaste-nach-rechts
	Vom Textcursor ausgehend das vorige Zeichen markieren	Umschalt + Pfeiltaste-nach-links
	Vom Textcursor ausgehend das nächste Wort markieren	Strg + Umschalt + Pfeiltaste-nach-rechts
	Vom Textcursor ausgehend	Strg + Umschalt +

Oberflächenelement	Aktion	Tastaturbefehl
	das vorige Wort markieren	Pfeiltaste-nach-links
	Vom Textcursor ausgehend bis zum Ende des Eingabebereichs markieren	Umschalt + Ende-Taste
	Vom Textcursor ausgehend bis zum Anfang des Eingabebereichs markieren	Umschalt + Pos1-Taste
	Markierung aufheben	Pfeiltasten-nach-rechts/-links
	Markierten Text in die Zwischenablage kopieren	Strg + c (alternativ: Strg + Einfügetaste auf dem Nummernblock)
	Markierten Text ausschneiden und in die Zwischenablage kopieren	Strg + x (alternativ: Umschalt + Entferntaste auf dem Nummernblock)
	Von der Zwischenablage einfügen	Strg + v (alternativ: Umschalt + Einfügetaste auf dem Nummernblock)
	Wertehilfe oder F4-Hilfe öffnen/schließen	Alt + Pfeil-nach-unten/-oben oder F4-Taste
	Editiervorgang beginnen	Erlaubtes Zeichen eingeben
	Editiervorgang beenden	Tabulatortaste
	Vom Textcursor ausgehend das nächste Zeichen löschen	Entfern-Taste
	Vom Textcursor ausgehend das vorige Zeichen löschen	Rücktaste
	Letzte Eingabe rückgängig machen	Strg + z
	Letzte Eingabe wiederholen	Strg + y
ItemListBox [Seite 20] (Listeneintrag)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Zwischen den Einträgen des Listenfeldes navigieren	Pfeiltaste-nach-oben/-unten
	Zum ersten Eintrag im Listenfeld springen	Pos1-Taste
LinkToAction [Seite 20] LinkToUrl [Seite 21] (Link)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Fokussierten Link aktivieren	Eingabetaste
MenuBar [Seite 21]	Fokusnavigation vorwärts	Tabulatortaste

Oberflächenelement	Aktion	Tastaturbefehl
(Menüleiste)	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Zwischen den Einträgen der Menüleiste navigieren	Pfeiltasten-nach-rechts/-links
	Zum ersten Eintrag der Menüleiste springen	Pos1-Taste
	Zum letzten Eintrag der Menüleiste springen	Ende-Taste
	Menü zu einem Eintrag der Menüleiste öffnen	Pfeiltaste-nach-unten
	Im geöffneten Menü navigieren	Pfeiltasten
	Fokussierten Menüpunkt auswählen	Eingabetaste
	Aktuelle Menüebene schließen	Escape-Taste
PhaseIndicator [Seite 21] (Phasen-Anzeiger)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Zwischen den einzelnen Phasen navigieren	Pfeiltasten-nach-rechts/-links
	Fokussierte Phase aufrufen	Eingabetaste
ProgressIndicator [Seite Fehler! Textmarke nicht definiert.] (Fortschrittsanzeiger)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
RadioButton [Seite 22] (Auswahlknopf) RadioButtonGroupByIndex [Seite 22] RadioButtonGroupByKey [Seite 23] (Gruppe von Auswahlknöpfen)	Fokusnavigation vorwärts (Fokus navigiert von außerhalb zuerst zur RadioButtonGroup, dann zum selektierten RadioButton-Element)	Tabulatortaste
	Fokusnavigation rückwärts (Bei der Rückwärtsnavigation zum Element hin navigiert der Fokus direkt von außerhalb zum selektierten RadioButton-Element innerhalb der RadioButtonGroup.)	Umschalt + Tabulatortaste
	Zwischen den RadioButton-Elementen innerhalb einer RadioButtonGroup navigieren und dabei das Auswahl-Kennzeichen mitbewegen (Selektion wird bei Navigation mitgeführt)	Pfeiltasten

Oberflächenelement	Aktion	Tastaturbefehl
	Zwischen den RadioButton-Elementen innerhalb einer RadioButtonGroup navigieren ohne dabei das Auswahl-Kennzeichen mitzubewegen (Selektion wird bei Navigation nicht mitgeführt)	Umschalt + Pfeiltasten
	Falls die Selektion bei der Navigation nicht mitgeführt wurde: Auswahl-Kennzeichen beim fokussierten, jedoch nicht selektierten RadioButton-Element setzen	Leertaste
RoadMap [Seite 23] (Roadmap, Ablaufdarstellung)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Zwischen den Ablaufschritten navigieren	Pfeiltasten-nach-links/-rechts
	Fokussierten Ablaufschritt aktivieren	Leertaste
ScrollContainer [Seite 23] (Gruppierungselement)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Fokussiertes Element eine Seite nach oben blättern (sofern es sich nicht um ein Eingabeelement handelt)	Bild-nach-oben-Taste
	Fokussiertes Element eine Seite nach unten blättern (sofern es sich nicht um ein Eingabeelement handelt)	Bild-nach-unten-Taste
Table [Seite 24] (Tabelle)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Innerhalb der Tabelle von Zelle zu Zelle (zeilenweise vorwärts) navigieren	Tabulatortaste
	Innerhalb der Tabelle von Zelle zu Zelle (zeilenweise rückwärts) navigieren	Umschalt + Tabulatortaste
	Innerhalb der Tabelle zwischen den Zellen navigieren	Pfeiltasten, solange der Fokus nicht auf eine editierbare Zelle trifft (da in diesem Fall auf die Zeichennavigation innerhalb der editierbaren Zelle umgeschaltet wird)

Oberflächenelement	Aktion	Tastaturbefehl
	Zwischen den Zeichen einer editierbaren Zelle navigieren	Pfeiltasten
	Vorwärts aus der Tabelle herauspringen (Gruppennavigation)	F6
TabStrip [Seite 30] (Register)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Fokussierten Registerindex aktivieren und zugehörige Registerkarte anzeigen	Leertaste
	Zwischen den Registerindizes navigieren	Pfeiltasten-nach-rechts/-links
	Wenn Registerkarten verdeckt sind: Nächste Registerkarte anzeigen (vorwärts blättern)	Strg + Bild-nach-unten-Taste
	Wenn Registerkarten verdeckt sind: Vorige Registerkarte anzeigen (rückwärts blättern)	Strg + Bild-nach-oben-Taste
TextEdit [Seite 31] (Text-Eingabefeld)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Textcursor innerhalb einer Zeile zwischen den Zeichen navigieren	Pfeiltasten-nach-rechts/-links
	Textcursor vor das erste Zeichen in der Zeile setzen	Pos1-Taste
	Textcursor hinter das letzte Zeichen in der Zeile setzen	Ende-Taste
	Vertikal zwischen den Zeilen navigieren	Pfeiltasten-nach-unten/-oben
	Zeichen vor dem Textcursor löschen	Rücktaste
	Zeichen nach dem Textcursor löschen	Entferntaste
	Innerhalb des Textes einen Zeilenumbruch einfügen	Eingabetaste
	Markierte Zeichen aus dem Text in die Zwischenablage kopieren	Strg + c (alternativ: Strg + Einfügetaste auf dem Nummernblock)
	Markierte Zeichen aus dem Text ausschneiden und in die Zwischenablage kopieren	Strg + x (alternativ: Umschalt + Entferntaste auf dem Nummernblock)
	Von der Zwischenablage in	Strg + v (alternativ:

Oberflächenelement	Aktion	Tastaturbefehl
	den Text einfügen	Umschalt + Einfügetaste auf dem Nummernblock)
	Innerhalb des Textes die letzte Aktion rückgängig machen	Strg + z (alternativ: Alt + Rücktaste)
	Innerhalb des Textes die letzte Aktion wiederholen	Strg + y
	Vom Textcursor ausgehend das nächste Zeichen markieren	Umschalt + Pfeiltaste-nach-rechts
	Vom Textcursor ausgehend das vorige Zeichen markieren	Umschalt + Pfeiltaste-nach-links
	Vom Textcursor ausgehend das nächste Wort markieren	Strg + Umschalt + Pfeiltaste-nach-rechts
	Vom Textcursor ausgehend das vorige Wort markieren	Strg + Umschalt + Pfeiltaste-nach-links
	Vom Textcursor ausgehend bis zum Ende der Zeile markieren	Umschalt + Ende-Taste
	Vom Textcursor ausgehend bis zum Anfang der Zeile markieren	Umschalt + Pos1-Taste
	Markierung aufheben	Pfeiltasten
	Eine Seite nach oben blättern	Bild-nach-oben-Taste
	Eine Seite nach unten blättern	Bild-nach-unten-Taste
	Direkt zur ersten Seite springen	Pos1-Taste
	Direkt zur letzten Seite springen	Ende-Taste
TextView [Seite 31] (Textanzeige)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
ToggleButton [Seite 32] (Schaltfläche zum ein-/ausschalten)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Fokussiertes Element ein- oder ausschalten	Leertaste oder Eingabetaste
ToggleLink [Seite 32] (Link zum auf-/zuklappen)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Fokussierten Link auf- oder	Leertaste oder

Oberflächenelement	Aktion	Tastaturbefehl
	zuklappen	Eingabetaste
Tray [Seite 32] (Gruppierungselement)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Fokussiertes Element expandieren	Plustaste auf dem Nummernblock
	Fokussiertes Element komprimieren	Minustaste auf dem Nummernblock
Tree [Seite 33] (Baum)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Einen Knoten nach unten navigieren	Pfeiltaste-nach-unten
	Einen Knoten nach oben navigieren	Pfeiltaste-nach-oben
	Geschlossenen Knoten expandieren	Pfeiltaste-nach-rechts
	Offenen Knoten komprimieren	Pfeiltaste-nach-links
TriStateCheckbox [Seite 35] (Ankreuzfeld mit dreifacher Statusanzeige)	Fokusnavigation vorwärts	Tabulatortaste
	Fokusnavigation rückwärts	Umschalt + Tabulatortaste
	Zwischen den drei Status des fokussiertes Elements wechseln	Leertaste



BreadCrumb

Ein BreadCrumb zeigt die aktuelle Seite im Kontext eines Navigationspfads an. Sie können zum Beispiel eine Historie der zuletzt besuchten Seiten oder die Struktur der gebotenen Informationen anzeigen. Ein BreadCrumb besteht aus einzelnen Links oder stellt als Ganzes einen Link dar.

Sie können in einen BreadCrumb zwei verschiedene Arten von Breadcrumb-Schritten einfügen:

- [BreadCrumbStep \[Extern\]](#)
- [MultipleBreadCrumbStep \[Extern\]](#)

BreadCrumbSteps werden an einzelne Context-Attribute gebunden, somit ist die Anzahl der angezeigten Schritte während der Laufzeit festgelegt. Ein MultipleBreadCrumbStep dagegen wird an einen Context-Knoten gebunden. Damit ist es möglich, die Anzahl der angezeigten Schritte zur Laufzeit dynamisch anzupassen.

Beispiel für die visuelle Darstellung

[Step 1](#) > [Step 2](#) > [Step 3](#) > [MultiStep1](#) > [MultiStep2](#) > MultiStep3



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `tooltip` überprüft.



Button

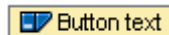
Das Oberflächenelement Button stellt eine Schaltfläche bzw. Taste am Bildschirm dar. Beim Betätigen des Button kann der Benutzer Anweisungen und Aktionen ausführen.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `tooltip` überprüft, wenn die text-Eigenschaft nicht gesetzt wurde.

Ein gesetzter Tooltip ist bei diesem Oberflächenelement sonst nicht zwingend erforderlich, könnte aber – wenn er detaillierte semantische Informationen beinhaltet – durchaus Sinn machen.

Beispiele für die visuelle Darstellung



CheckBox

Mit einer CheckBox realisieren Sie das Konzept eines einzelnen An-/Aus-Schalters in Form eines Ankreuzfeldes. Eine CheckBox erlaubt dem Benutzer die Auswahl eines Booleschen Wertes (TRUE/FALSE). Das Oberflächenelement besteht aus einer Grafik mit dazugehörigem Text. Ein Häkchen in der Grafik zeigt an, dass die Option ausgewählt und der Wert auf TRUE gesetzt wurde.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `tooltip` überprüft, wenn die Eigenschaften `text` und `label` nicht gesetzt wurden.

Ein gesetzter Tooltip ist bei diesem Oberflächenelement sonst nicht zwingend erforderlich, könnte aber – wenn er detaillierte semantische Informationen enthält – durchaus Sinn machen

Beispiel für die visuelle Darstellung



CheckBoxGroup

Die CheckBoxGroup erlaubt dem Benutzer, ein Element aus einer Menge vorgegebener Alternativen durch Ankreuzen auszuwählen. Das Oberflächenelement CheckBoxGroup ordnet die einzelnen [CheckBoxes \[Seite 15\]](#) ein- oder mehrspaltig in tabellarischer Form an.

Für die Darstellung einer angekreuzten CheckBox in einer CheckBoxGroup verwenden Sie für das dazugehörige Context-Element die Methode `SET_SELECTED` von `IF_WD_CONTEXT_ELEMENT`. Beachten Sie, dass die Selektion der CheckBoxen in einer CheckBoxGroup **nicht** über ein gebundenes Context-Attribut geschieht.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `accessibilityDescription` überprüft, wenn die Eigenschaft `tooltip` nicht gesetzt wurde.

Beispiele für die visuelle Darstellung

Einspaltig

- CheckBox 1
- CheckBox 2
- CheckBox 3
- CheckBox 4
- CheckBox 5
- CheckBox 6
- CheckBox 7

Zweispaltig

- CheckBox 1
- CheckBox 3
- CheckBox 5
- CheckBox 7
- CheckBox 2
- CheckBox 4
- CheckBox 6

Dreispaltig

- CheckBox 1
- CheckBox 4
- CheckBox 7
- CheckBox 2
- CheckBox 5
- CheckBox 6
- CheckBox 3
- CheckBox 6



DateNavigator

Das DateNavigator-Oberflächenelement unterstützt die Anzeige und Eingabe eines Datums. Unter anderem ermöglicht es das Navigieren innerhalb eines Kalenders sowie die Auswahl eines Tages, Monats, Jahres oder Datumsbereichs. In erster Linie soll das Oberflächenelement jedoch dem Benutzer bei der Eingabe eines Datums dienen und das Datum in einem entsprechenden Format eintragen.

Mit Hilfe des [DateNavigatorLegend \[Extern\]](#)- und des [DateNavigatorMarking \[Extern\]](#)-Elements kann dem DateNavigator-Oberflächenelement eine Legende hinzugefügt werden, die der Beschreibung markierter Daten dient. Damit können Sie dem Benutzer zu bestimmten Daten, die einer Kategorie zugeordnet werden, Informationen weitergeben. Beispielsweise können verschiedene Veranstaltungen im Kalender farblich hervorgehoben und die Veranstaltung mit Thema, Uhrzeit und Ort beschrieben werden.

Beispiel für die visuelle Darstellung



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `accessibilityDescription` überprüft, wenn die Eigenschaft `tooltip` nicht gesetzt wurde.

Die Eigenschaft `accessibilityDescription` gilt für den gesamten DateNavigator und wird von Bildschirmausleseprogrammen vorgelesen, wenn der DateNavigator selbst fokussiert wird. Der Tooltip wird für jeden Tag angezeigt, es sei denn, dieser Tag hat eine Markierung und diese enthält einen gefüllten Tooltip.



DropDownByIndex

Ein DropDownByIndex-Oberflächenelement bietet dem Benutzer eine Dropdown-Listbox an. Aus der Auswahlliste können Sie höchstens einen Eintrag auswählen. Das Oberflächenelement besteht aus einem Textfeld, einer Drucktaste sowie einer Auswahlliste. Dabei wird ein bereits ausgewählter Listeneintrag im Textfeld dargestellt. Eine Liste aller möglichen Werte wird angezeigt, wenn man die Drucktaste betätigt.

Beispiel für die Verwendung

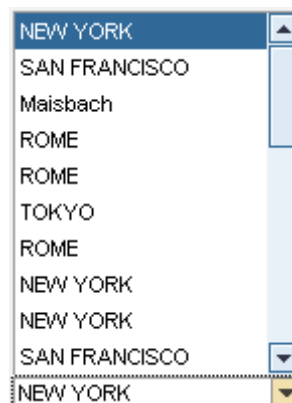
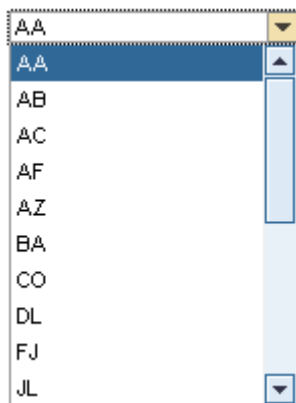
Wenn Sie eine Table mit Dropdown-Listen erstellen möchten, wobei jede Table-Zeile unterschiedliche Werte in ihrer Dropdown-Liste haben kann, verwenden Sie DropDownByIndex.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft label überprüft.

Wenn kein label gesetzt wurde und auch für das entsprechende gebundene Context-Element im ABAP-Dictionary kein Beschreibungstext angegeben wurde, findet eine Überprüfung auf die Eigenschaft tooltip statt.

Beispiele für die visuelle Darstellung



In der Darstellung am Bildschirm unterscheiden sich die beiden DropDown-Listbox-Oberflächenelemente DropDownByIndex und [DropDownByKey \[Seite 17\]](#) nicht. Jedoch verfolgt das Datenbindungsmodell für das DropDownByKey-Oberflächenelement ein völlig anderes Konzept.



DropDownByKey

Ein DropDownByKey-Oberflächenelement bietet dem Benutzer eine Dropdown-Listbox an. Aus der Auswahlliste können Sie höchstens einen Eintrag auswählen. Das Oberflächenelement besteht aus einem Textfeld, einer Drucktaste sowie einer Auswahlliste. Dabei wird ein bereits ausgewählter Listeneintrag im Textfeld dargestellt. Eine Liste aller möglichen Werte wird angezeigt, wenn die Drucktaste betätigt wird.

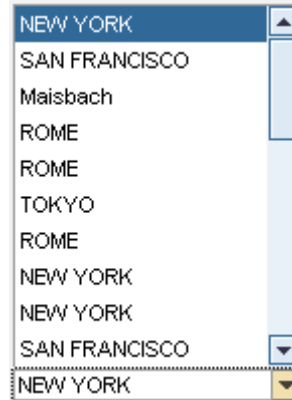
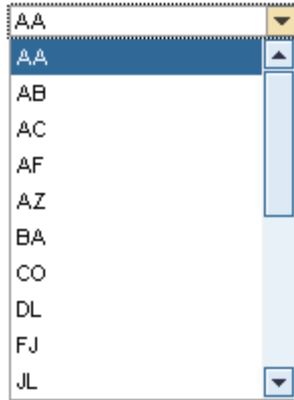
Wenn Sie eine DropDown-Liste erstellen möchten, in der manche Einträge mehrfach vorhanden sind, verwenden Sie DropDownByKey.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft label überprüft.

Wenn kein label gesetzt wurde und auch für das entsprechende gebundene Context-Element im ABAP-Dictionary kein Beschreibungstext angegeben wurde, findet eine Überprüfung auf die Eigenschaft tooltip statt.

Beispiele für die visuelle Darstellung



In der Darstellung am Bildschirm unterscheiden sich die beiden DropDownList-Oberflächenelemente [DropDownByIndex \[Seite 17\]](#) und DropDownByKey nicht. Jedoch verfolgt das Datenbindungsmodell für das DropDownByKey-Oberflächenelement ein völlig anderes Konzept.



Group

Das Group-Oberflächenelement kann dazu verwendet werden, eine Reihe von Oberflächenelementen unter einem gemeinsamen Titel zu gruppieren und zählt deshalb auch zu den Oberflächenelement-Containern. In seinem Aussehen erinnert dieses Oberflächenelement an eine Anzeigetafel mit farbigem Hintergrund.

Bei einer Group hat die Eigenschaft enabled keinerlei Auswirkung auf die Kinder, die Sie in den [UIElementContainer \[Extern\]](#) eingefügt haben. Wenn Sie beispielsweise die Eigenschaft enabled im Group-Oberflächenelement auf den Wert false setzen, wird nicht automatisch ein darin eingefügtes Eingabefeld deaktiviert. Sollen auch die Kinder in diesem Group-Oberflächenelement deaktiviert sein, dann müssen Sie die entsprechende Eigenschaft für jedes Oberflächenelement eigens dafür setzen.

Beispiele für die visuelle Darstellung

Group mit primarycolor

Organization			
Organizational Unit:	<input type="text"/>	Division:	<input type="text"/>
Sales Organization:	<input type="text"/>	Sales Office:	<input type="text"/>
<input type="button" value="Go"/> <input type="button" value="Cancel"/>			

Group mit secondarycolor und ToolBar

Organization

Create New

Organizational Unit:

Division:

Sales Organization:

Sales Office:

Go
Cancel

Group mit sapcolor

Filter By

Org Unit

Cost Center

Position

Employee



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `accessibilityDescription` überprüft, wenn die `caption`-Eigenschaft nicht gesetzt wurde.

Es findet keine Überprüfung auf die Eigenschaft `tooltip` statt.

Ein gesetzter Tooltip ist bei diesem Oberflächenelement nicht zwingend erforderlich, könnte aber – wenn er detaillierte semantische Informationen enthält – durchaus Sinn machen.



Image

Über das Oberflächenelement `Image` können Grafiken in einem Format, das der Web-Server verarbeiten kann, wie z.B. GIF-, JPG- und PNG-Format, in die Web-Anwendung eingebunden werden. Höhe und Breite der Grafik können über die Eigenschaften `height` und `width` bestimmt werden. Die Grafik wird ohne Rand dargestellt.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `tooltip` überprüft, wenn die Eigenschaften `label` und `isDecorative` nicht gesetzt wurden.



InputField

Das `InputField`-Oberflächenelement erlaubt dem Benutzer, einen einzeiligen Text zu editieren oder anzuzeigen. Mit einem `InputField` können Sie einen beliebigen einfachen Datentyp editieren. Die Konvertierung von der internen in die Anzeige-Darstellung und bei Eingabe durch den Benutzer wieder zurück in die interne Darstellung erfolgt dabei automatisch. Tritt nach der Benutzereingabe bei der Konvertierung in die interne Darstellung ein Fehler auf, so wird der Wert nicht in den Context zurückgestellt, sondern verbleibt im Data Container. Bei der nächsten Anzeige wird das `InputField` mit dem fehlerhaften Wert rot umrandet und mit einer Fehlermeldung versehen angezeigt.

Die Überprüfung einer Eingabe erfolgt erst durch das Auslösen eines Roundtrips.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft label überprüft.

Wenn kein label gesetzt wurde und auch für das entsprechende gebundene Context-Element im ABAP-Dictionary kein Beschreibungstext angegeben wurde, findet eine Überprüfung auf die Eigenschaft tooltip statt.

Beispiel für die visuelle Darstellung mit einem Label



ItemListBox

Dieses Oberflächenelement ähnelt dem klassischen GUI-Konzept einer Auswahlliste mit Einfach- und Mehrfachselektion (Listbox). Es wird eine Liste von Texteinträgen in einem Kasten von fester Größe angezeigt, wobei bei Bedarf auch geblättert werden kann. Für die Werte werden eine oder zwei Spalten angezeigt und eventuell auch eine Spalte für Symbole, die vor der Wertespalte aufgeführt werden.

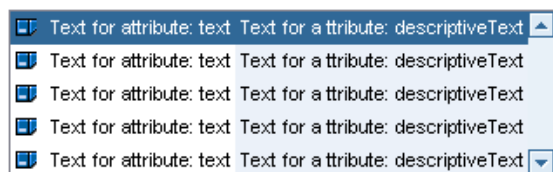
Im Fall der Einfachauswahl wird der gewählte Eintrag durch die Lead-Selection der dataSource bestimmt. Beim Ändern der Selektion durch den Benutzer wird die Eigenschaft selectionChangeBehaviour beachtet. Bei der Mehrfachauswahl werden die gewählten Einträge nur durch die Selektion der dataSource bestimmt, selectionChangeBehaviour wird nicht beachtet.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft label überprüft.

Wenn kein label gesetzt wurde und auch für das entsprechende gebundene Context-Element im ABAP-Dictionary kein Beschreibungstext angegeben wurde, findet eine Überprüfung auf die Eigenschaft tooltip statt.

Beispiel für die visuelle Darstellung



LinkToAction


Das LinkToAction-Oberflächenelement stellt eine Art Hypertext-Verknüpfung dar. Ein Ansteuern des Links löst eine Web-Dynpro-Aktion aus.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft tooltip überprüft, wenn die Eigenschaft text nicht gesetzt wurde.

Ein gesetzter Tooltip ist bei diesem Oberflächenelement nicht zwingend erforderlich, könnte aber – wenn er detaillierte semantische Informationen enthält – durchaus Sinn machen.

Beispiele für die visuelle Darstellung

Einfacher LinkToAction	LinkToAction mit Menü-Aggregation
 LinkToAction	 LinkToAction ≡



LinkToURL

Das LinkToURL-Oberflächenelement ist eine Hypertext-Verknüpfung. Ein Anwählen des Links führt zu einer beliebigen Web-Ressource (URL).


Der LinkToUrl dient ausschließlich dem Öffnen von URLs in einem separaten Fenster. Zum Verlassen der Web-Dynpro-Applikation und Anzeigen einer neuen URL verwenden Sie einen Exit-Plug.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft tooltip überprüft, wenn die Eigenschaft text nicht gesetzt wurde.

Ein gesetzter Tooltip ist bei diesem Oberflächenelement nicht zwingend erforderlich, könnte aber – wenn er detaillierte semantische Informationen enthält – durchaus Sinn machen.

Beispiele für die visuelle Darstellung

Einfacher LinkToURL	LinkToURL mit Menü-Aggregation
 LinkToURL	 LinkToURL ≡



MenuBar

Eine MenuBar dient der gegliederten Darstellung von Aktionen. Die MenuBar bildet eine Leiste, die in verschiedene Blöcke, die [Menus \[Extern\]](#), gegliedert werden kann. Unter diesen Blöcken können Sie einzelne Menüpunkte und weitere Menus anordnen.

Beispiel für die visuelle Darstellung

Text for Attribute: title ▾ Text for Attribute: title ▾ Text for Attribute: title ▾ Text for Attribute: title ▾



PhaseIndicator

Ähnlich dem [RoadMap \[Seite 23\]](#)-Oberflächenelement visualisiert das PhaseIndicator-Oberflächenelement die einzelnen Schritte eines Wizards. Jeder Schritt wird durch ein eigenes [Phase \[Extern\]](#)-Objekt repräsentiert. Im Unterschied zum RoadMap-Oberflächenelement soll die Anwendungsentwicklung damit größere Schritte, die vom Benutzer mehr Zeit in Anspruch nehmen können, visualisieren.

Beispiel für die visuelle Darstellung



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `accessibilityDescription` überprüft, wenn die `tooltip`-Eigenschaft nicht gesetzt wurde.



RadioButton

Das `RadioButton`-Oberflächenelement repräsentiert eine Drucktaste mit zwei Zuständen (An/Aus), das dem Benutzer ermöglicht, Optionen auszuwählen. Sie können das `RadioButton`-Oberflächenelement dann verwenden, wenn Sie die einzelnen Radiobuttons auf dem Bildschirm nicht tabellarisch in einer Gruppe, sondern verteilt darstellen möchten. Die Umschaltfunktion des Auswahlknopfs wird dann ermöglicht, wenn Sie die Oberflächenelemente an das gleiche `Context`-Attribut binden.

Der Auswahlknopf ist genau dann ausgewählt, wenn das `Context`-Attribut, an das die Eigenschaft `selectedKey` gebunden ist, den Wert des zu diesem Auswahlknopf gehörigen Schlüssels enthält. Dieser Schlüssel wird durch die Eigenschaft `keyToSelect` gesetzt.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `tooltip` überprüft, wenn die Eigenschaften `text` und `label` nicht gesetzt wurden.

Ein gesetzter `Tooltip` ist bei diesem Oberflächenelement nicht zwingend erforderlich, könnte aber – wenn er detaillierte semantische Informationen enthält – durchaus Sinn machen.



RadioButtonGroupByIndex

Das `RadioButtonGroupByIndex`-Oberflächenelement stellt eine Anzahl von `RadioButtons` gruppiert in Spalten und Zeilen dar. Im Unterschied zum `CheckBoxGroup`-Oberflächenelement kann der Benutzer höchstens ein Element auswählen.

Beispiel für die visuelle Darstellung

- Radiobutton 1
- Radiobutton 2
- Radiobutton 3

In der Darstellung am Bildschirm unterscheiden sich die beiden `RadioButtonGroup`-Oberflächenelemente [RadioButtonGroupByKey \[Seite 23\]](#) und `RadioButtonGroupByIndex` nicht. Jedoch verfolgt das Datenbindungsmodell für das `RadioButtonGroupByKey`-Oberflächenelement ein völlig anderes Konzept (siehe Abschnitt *Datenbindung*).



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `accessibilityDescription` überprüft, wenn die `tooltip`-Eigenschaft nicht gesetzt wurde.



RadioButtonGroupByKey

Das `RadioButtonGroupByKey`-Oberflächenelement gruppiert eine Anzahl von `RadioButtons` in einer Tabelle. Im Unterschied zum `CheckBoxGroup`-Oberflächenelement kann der Benutzer höchstens ein Element auswählen.

Beispiel für die visuelle Darstellung

- Radiobutton 1
- Radiobutton 2
- Radiobutton 3

In der Darstellung am Bildschirm unterscheiden sich die beiden `RadioButtonGroup`-Oberflächenelemente [RadioButtonGroupByIndex \[Seite 22\]](#) und `RadioButtonGroupByKey` nicht. Jedoch verfolgt das Datenbindungsmodell für das `RadioButtonGroupByKey`-Oberflächenelement ein völlig anderes Konzept (siehe Abschnitt *Datenbindung*).



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `accessibilityDescription` überprüft, wenn die `tooltip`-Eigenschaft nicht gesetzt wurde.

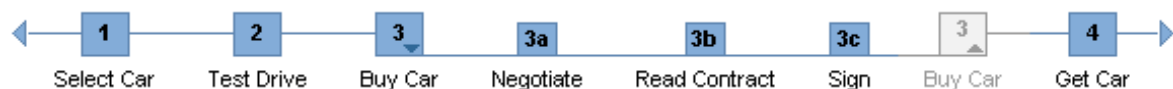


RoadMap

Das `RoadMap`-Oberflächenelement visualisiert die Schritte eines Wizards. Jeder Schritt wird durch ein eigenes `RoadMapStep [Extern]`-Objekt bzw. `MultipleRoadMapStep [Extern]` repräsentiert. Sie können die Anfangspunkte und Endpunkte dieses Oberflächenelements mit verschiedenen Symbolen markieren. Wenn Sie der Eigenschaft `startPointDesign` oder `endPointDesign` den Wert `more` zuordnen, dann können Sie damit den Benutzer darauf hinweisen, dass sich weitere Schritte vor dem ersten sichtbaren Schritt bzw. nach dem letzten sichtbaren Schritt befinden.

Das `RoadMap`-Oberflächenelement eignet sich zur schrittweisen Darstellung von Arbeitsabläufen. Die Anwendungsentwicklung kann damit kleine Einzelschritte eines klar festgelegten Arbeitsprozesses visualisieren.

Beispiel für die visuelle Darstellung



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `accessibilityDescription` überprüft, wenn die `tooltip`-Eigenschaft nicht gesetzt wurde.



ScrollContainer

Das `ScrollContainer`-Oberflächenelement stellt den sichtbaren Oberflächenelementen [Group \[Seite 18\]](#) und [Tray \[Seite 32\]](#) die Möglichkeit für horizontalen sowie vertikalen Bildlauf bereit.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, werden im Rahmen des Syntaxchecks die Eigenschaften `accessibilityDescription` und `tooltip` nicht überprüft.

Ein gesetzter Tooltip ist bei diesem Oberflächenelement nicht zwingend erforderlich, könnte aber – wenn er detaillierte semantische Informationen enthält – durchaus Sinn machen.



Table

In einer Web-Dynpro-Tabelle werden Daten zweidimensional in Tabellenzellen, die in Zeilen und Spalten angeordnet sind, dargestellt.

Die Web-Dynpro-Tabelle setzt sich aus dem übergeordneten UI-Element **Table** und mehreren View-Elementen für Tabellenspalten, [TableColumn \[Extern\]](#), zusammen. Die Table enthält Eigenschaften, die für die gesamte Tabelle gelten, wie beispielsweise die Festlegung, ob die Tabelle eingabefähig ist oder nicht (`readOnly=true`). Die TableColumn dagegen enthält sowohl Eigenschaften, die nur für die Spaltenüberschrift gelten, als auch Eigenschaften, die für die gesamte Spalte gelten und zusätzlich auch Eigenschaften, die für Zellen gelten und deren Wert pro Bindung pro Zeile variieren kann.

Die Spaltenüberschrift wird durch die Aggregation [Header \[Extern\]](#) über ein Caption-UI-Element realisiert. Die zweite Aggregation des TableColumn mit dem Namen [TableCellEditor \[Extern\]](#) enthält ein UI-Element, das für die Darstellung der Zellen in der Spalte genutzt wird. Wenn die entsprechenden UI-Elemente auf `readOnly`-Attribute gebunden sind, sind sie daher auch in der Table nicht eingabebereit.

Die Table stellt die Elemente des an `dataSource` gebundenen Context-Knotens als Zeilen dar. Alle Eigenschaften eines TableCellEditors, die auf ein Attribut der `dataSource` gebunden sind, bzw. an Context-Knoten, die innerhalb der `dataSource` liegen, nehmen in einer Zeile nicht wie sonst üblich den Wert der Lead-Selection des Knotens an, sondern den Wert der aktuellen Zeile.

Visualisierung

Eine Table besteht aus 5 horizontalen Bereichen:

- Table-Überschrift (Table-Header)
Diese Aggregation (*Header einfügen*) ist optional
- ToolBar
Diese Aggregation (*Toolbar einfügen*) ist optional
- Spaltenüberschriften/Hierarchie von Spaltenüberschriften (TableColumn-Header)
Ist für alle Spalten (auch unsichtbare) der Header auf `Caption.visibility=none` gesetzt, so wird die Zeile mit den Spaltenüberschriften ausgeblendet
- Datenzeilen mit den TableCellEditoren (*Cell-Editor einfügen*)

Der Inhalt der Datenzeilen wird bestimmt durch:

- TableColumnns
- `Table.visibleRowCount`
- Inhalt der `Table.dataSource`
- TableCellEditoren

- Scrollbars

Für jede Table können am rechten Rand Scrollbars angezeigt werden, wenn Sie bei den [Applikationsparametern \[Extern\]](#) den Parameter WDTABLENAVIGATION auf SCROLLBAR gesetzt haben.

Beispiel:

List of Cd					
	Artist	Title	Label	Price	Salary currency
	Miles Davis	Kind of Blue	Col (Sony)	5,99	EUR
	Miles Davis	Birth of the Cool	Blue Note (EMI)	4,99	EUR
	Miles Davis	In a Silent Way	Col (Sony)	9,99	EUR
	United Jazz & Rock	na endlich! Live	mood records	10,99	EUR
	Bryan Adams	Best of me	A & M Reco (Univer	11,89	EUR
	Robbie Williams	Swing When You're	Chrysalis (EMI)	3,99	EUR
	AC/DC	Highway To Hell	Epc (Sony)	7,58	EUR
	the Rolling Stones	Forty Licks - New V	Virgin UK (EMI)	21,86	EUR

Beim Scrollen durch eine Table wird ein Tooltip angezeigt. Die Texte für diese Scroll-Tipps legen Sie über die Aggregation SCROLL_TIP_PROVIDER fest. Wenn neue Daten geladen werden müssen, wird der Roundtrip zum Server durch das Loslassen der Maustaste ausgelöst.

- Fußzeile (für den barrierefreien Modus)

Die Sichtbarkeit des Footers wird über Table.footerVisible gesteuert.

In der Fußzeile der Tabelle werden Drucktasten zur Verfügung gestellt, die ein Navigieren erlauben. Dazu werden folgende Blätter-Funktionen angeboten:

Erste Seite (First page)	Auf der Seite oben (Page up)	Zeile nach oben (Line up)
Letzte Seite (Last page)	Auf der Seite nach unten (Page down)	Zeile nach unten (Line down)



Beispiel für eine einfache Table mit Header, Spaltenüberschriften, Datenzeilen und Scrollbar:

List of Cd					
	Artist	Title	Label	Price	Salary currency
	Miles Davis	Kind of Blue	Col (Sony)	5,99	EUR
	Miles Davis	Birth of the Cool	Blue Note (EMI)	4,99	EUR
	Miles Davis	In a Silent Way	Col (Sony)	9,99	EUR
	United Jazz & Roc	na endlich! Live	mood records	10,99	EUR
	Bryan Adams	Best of me	A & M Reco (Unive	11,89	EUR
	Robbie Williams	Swing When You'	Chrysalis (EMI)	3,99	EUR
	AC/DC	Highway To Hell	Epc (Sony)	7,58	EUR
	the Rolling Stones	Forty Licks - New	Virgin UK (EMI)	21,86	EUR

Neben diesen allgemeinen Funktionen kann eine Table auch wesentlich komplexere Features enthalten:

- Selektion und Selektionsspalte
- Selektionsschalter
- Hierarchien durch Masterspalten
- Popins (auf Zellen- und Zeilen-Ebene)
- Überschriften-Gruppierung
- Zeilengruppierung
- Spalten-Scrolling
- Varianten
- Summierung


Beachten Sie, dass die Summierung nur für die [SAP List Viewer Integration \[Extern\]](#) zur Verfügung steht.

Selektion und Selektionsspalte

Zusätzlich zu den durch die TableColumnns definierten Spalten kann die Tabelle noch eine Selektionsspalte enthalten. Diese wird durch die Eigenschaft [Table.selectionMode \[Extern\]](#) gesteuert.



Beispiel für eine Selektionsspalte in einer Table:

Flight data						
	Airline	Flight Number	Date	Airfare	Airline Currency	Plane Type
<input type="checkbox"/>	AA	0017	14.03.2005	4,22	JPY	747-400
<input type="checkbox"/>			16.03.2005	422,94	USD	747-400
<input type="checkbox"/>			13.04.2005	422,94	USD	747-400
<input type="checkbox"/>			11.05.2005	422,94	USD	747-400
<input type="checkbox"/>			08.06.2005	422,94	USD	747-400

Bei readOnly-Tables (readOnly=true) kann eine Zeile auch durch Klick auf den Zeileninhalt selektiert werden. Beachten Sie dabei, dass beim Klicken auf einen Link, Button oder die Auf-/Zuklappen-Ikone der Tree-Spalte die Selektion jedoch nicht geändert wird.

Selektionsschalter

Tables können Selektionsschalter enthalten. Hierbei sind zwei Fälle zu unterscheiden:

- Wenn `selectionMode=auto` und die `dataSource` gesetzt sind oder wenn `selectionMode=multi`, ist Mehrfachselektion möglich:
Der Benutzer kann über den Selektionsschalter alles (alle Zeilen) auswählen oder die Auswahl löschen.
Sie finden ein Beispiel im System in der Component `WDR_TEST_TABLE` unter `Selection`.
- Wenn die Table mit der Aggregation `TableRowArrangement` der Art `TableRowGrouping` versehen ist, kann der Benutzer zusätzlich zum *Alles auswählen* und *Auswahl löschen* auch alles aufklappen bzw. alles zuklappen:

Sie finden ein Beispiel im System in der Component WDR_TEST_TABLE unter RowGrouping.



Beispiele für Selektionsschalter:

Flight data

	Airline	Flight Number	Date	Airfare	Airline Currency	Plane Type
Select All		0017	14.03.2005	4,22	JPY	747-400
Delete Selection			16.03.2005	422,94	USD	747-400
			13.04.2005	422,94	USD	747-400
			11.05.2005	422,94	USD	747-400
			08.06.2005	422,94	USD	747-400

Table Table+TreeByNesting Table+TreeByKey Table+TableRowGrouping

Boolean

Select All
Delete Selection
Expand All
Collapse All

Hierarchie

Tabellen können auch mittels der Aggregation Masterspalte (*Masterspalte einfügen*) eine Hierarchie darstellen.

Es gibt zwei Arten der Masterspalte, die sich im Aufbau der dataSource unterscheiden:

- [TreeByKeyTableColumn \[Extern\]](#)

Die dataSource ist ein flacher Knoten, in dem die Hierarchie mittels Parent-Key-Beziehung ausgedrückt wird.

- [TreeByNestingTableColumn \[Extern\]](#)

Die dataSource ist ein Rekursionsknoten, deren Unterknoten untergeordnete Elemente enthalten.



Beispiel für eine Masterspalte in einer Table:

MIME Browser

	Name	Mime Type	Size
▼	SAP	Folder	0
▶	BC	Folder	0
▼	PUBLIC	Folder	0
▶	SVER	Folder	0
▶	ALVTest	Folder	0

Popin

Mit [TablePopin \[Extern\]](#)s können weitere Informationen zu einer Zeile oder Tabellenzelle direkt unterhalb der betreffenden Zeile angezeigt werden.



Beispiel für ein Row-Popin in einer Table:

Flight data							
		Airline	Flight Number	Date	Airfare	Airline Currency	Plane Type
▶	Details	AA	0017	14.03.2005	4,22	JPY	747-400
▼	Details	AA	0017	16.03.2005	422,94	USD	747-400
Detailed Information ✕							
Max. capacity econ.:					<input type="text" value="385"/>		
Occupied econ.:					<input type="text" value="375"/>		
▶	Details	AA	0017	13.04.2005	422,94	USD	747-400
▶	Details	AA	0017	11.05.2005	422,94	USD	747-400
▶	Details	AA	0017	08.06.2005	422,94	USD	747-400

Überschriften-Gruppierung

Wenn mehrere Zeilen oder Spalten die gleiche Überschrift haben, ist es möglich, über die Aggregation *Gruppenspalten einfügen* Zeilen und Spalten zusammenzufassen und eine Hierarchie zu erstellen. Hierbei gibt es zwei Arten:

- Gruppenspalten (TableColumnGroup)
- Table-Spalten (TableColumn)

Dabei werden die ersten drei Header-Zeilen farblich hervorgehoben.



Beispiel für eine Header-Gruppierung in einer Table:

Group 1				Group 3
Group 1			Group 3 / 1	
Gr1 H1	Gr1 H2	Header 2	Header 3 / 1 / 1	
Column 1	Column 2	Column 3	Column 4	
Column 1	Column 2	Column 3	Column 4	
Column 1	Column 2	Column 3	Column 4	
Column 1	Column 2	Column 3	Column 4	
Column 1	Column 2	Column 3	Column 4	

Zeilengruppierung

Um gleiche Felder einer Spalte zusammenzufassen, muss der Wert `TableColumn.groupingValue` auf die `dataSource` gebunden werden. Es werden dann die aufeinander folgenden Felder zu einem Feld zusammengefasst, die den gleichen `groupingValue` und `selectedCellVariant` haben.



Beispiel für eine Zeilengruppierung in einer Table:

Flight data						
✉	Airline	Flight Number	Date	Airfare	Airline Currency	Plane Type
	AA	0017	14.03.2005	4,22	JPY	747-400
			16.03.2005	422,94	USD	747-400
			13.04.2005	422,94	USD	747-400
			11.05.2005	422,94	USD	747-400
			08.06.2005	422,94	USD	747-400

Spalten-Scrolling



Generell empfiehlt es sich, nicht zu viele Spalten in eine Table einzufügen. Falls dies dennoch nötig sein sollte, dann kann durch Spalten-Scrolling das horizontale Scrolling des Browserfensters verhindert werden.

Es gibt zwei sinnvolle Varianten für das Spalten-Scrolling:

- Feste Angabe der Anzahl sichtbarer Spalten

Dieser Modus wird durch die Angabe einer festen Zahl scrollbarer Spalten in der Eigenschaft [scrollableColumnCount \[Extern\]](#) aktiviert. Es werden dann immer maximal die angegebene Anzahl an Spalten angezeigt und ein Paginator im Footer wird zum Scrollen angeboten.

- In die Table passende Spalten

Damit dieser Modus zuverlässig funktioniert, müssen eine Reihe von Voraussetzungen erfüllt werden:

- scrollableColumnCount = -2 aktiviert den Modus
- Es sollte zudem mit fixedTableLayout=true aktiviert werden, dass Spalten nur so breit wie angegeben werden
- Die Breitenangaben aller Spalten und der Table müssen gesetzt sein, und das auch in der selben Einheit (Ausnahme: die Table-Breite kann auch in anderen Einheiten sein, wenn die Spaltenbreiten in Prozent angegeben sind)

Es werden dann so viele Spalten angezeigt wie in die Table passen. Sollte eine der Spalten zu breit sein, eine falsche Einheit oder gar keine Breitenangabe haben, so wird sie einzeln in voller Table-Breite angezeigt. Beachten Sie, dass bei vorhandener Selektionsspalte die Table um diese Spalte breiter als angegeben wird.

Variante

Eine Variante ist eine Liste alternativer Zellen. Sie wird über die Aggregation *Zellenvarianten einfügen* erstellt. Die *TableStandardCell* hat einen eigenen TableCellEditor.



Flight data	
Name	Value
MANDT	000
CARRID	AA
CONNID	0017
FLDATE	14.03.2005
PRICE	422



Beachten Sie, dass von den drei Auswahlmöglichkeiten lediglich die *TableStandardCell* für eine Table verwendet werden sollte. Die anderen beiden Auswahlmöglichkeiten sind für den [ALV \[Extern\]](#) reserviert.

Summierung

Über die Summierung können Sie Zwischensummen und Endsummen bilden.



Beispiel für Summierungen in einer Table:

Flight data							
	Airline	Flight Number	Date		Airfare	Airline C...	Plane Type
	A,A	0064	02.09.2005		422,94	USD	A310-300
			30.09.2005		422,94	USD	A310-300
			28.10.2005		422,94	USD	A310-300
			25.11.2005		422,94	USD	A310-300
			23.12.2005		422,94	USD	A310-300
			20.01.2006		422,94	USD	A310-300
			17.02.2006		422,94	USD	A310-300
			17.03.2006		422,94	USD	A310-300
			06.06.2006		422,94	USD	A310-300
			A,A	▲ 0064		▪	6.767,04
	▲ A,A			▪ ▪	12.692,42	EUR	
	A,Z	0555	16.03.2005		185,00	EUR	A319
			13.04.2005		185,00	EUR	A319
			11.05.2005		185,00	EUR	A319
			08.06.2005		185,00	EUR	A319

Barrierefreiheit

Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `accessibilityDescription` überprüft, wenn die `caption`-Eigenschaft nicht gesetzt wurde.

Zusätzlich findet eine Überprüfung statt, ob für Spalten die Aggregation [Header \[Extern\]](#) gesetzt ist bzw. ob die Aggregation Header sichtbar ist.

Es findet keine Überprüfung auf die Eigenschaft `tooltip` statt.



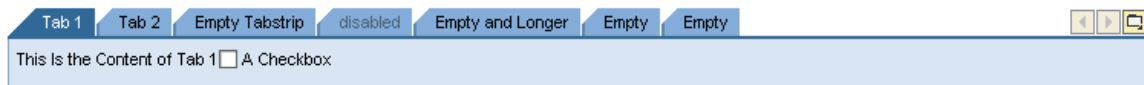
TabStrip

Das TabStrip-Oberflächenelement ermöglicht die Darstellung eines Registers. Der Benutzer kann zwischen mehreren Registerkarten durch Auswählen eines bestimmten Reiters ([Tab \[Extern\]](#)) wechseln. Zum Anzeigen des Inhalts wird der gleiche Fensterbereich benutzt und damit mit den anderen Registerkarten geteilt. Der Benutzer kann sich den Inhalt einer Registerkarte durch Auswählen des Reiters anzeigen lassen.

Wenn bei der Table kein `selectedTab` angegeben ist oder der in `selectedTab` angegebene Tab nicht sichtbar ist, dann wird stattdessen der erste sichtbare Tab angezeigt.

Dabei wird weder das onSelect-Ereignis ausgelöst noch die selectedTab-Eigenschaft verändert.

Beispiel für die visuelle Darstellung



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `accessibilityDescription` überprüft, wenn die tooltip-Eigenschaft nicht gesetzt wurde.



TextEdit

Das TextEdit-Oberflächenelement ermöglicht die Eingabe und Darstellung von mehrzeiligem Text. Der Text in diesem Oberflächenelement wird mit Schriftart, Schriftgröße und in einem einheitlichen Schriftstil dargestellt. Das Oberflächenelement wird mit Rahmen angezeigt, die Größe dieses Rahmens wird durch die `col`- und `row`-Eigenschaften bestimmt. Wenn die Anzahl der Zeilen den Wert der `row`-Eigenschaft überschreitet, dann wird ein vertikaler Bildlauf (Scrollbar) angezeigt.

Falls Sie der `wrapping`-Eigenschaft den Wert `off` zugeordnet haben, dann wird ein horizontaler Bildlauf aktiviert, wenn die Textzeilenlänge den Wert der `col`-Eigenschaft überschreitet.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `label` überprüft.

Wenn kein `label` gesetzt wurde und auch für das entsprechende gebundene Context-Element im ABAP-Dictionary kein Beschreibungstext angegeben wurde, findet eine Überprüfung auf die Eigenschaft `tooltip` statt.

Beispiel für die visuelle Darstellung: TextEdit mit Label



TextView

Das TextView-Oberflächenelement ermöglicht die Darstellung von Text.



Achten Sie bei der Verwendung des TextView-Oberflächenelements darauf, dass Sie dieses Oberflächenelement für die Barrierefreiheit einer Anwendung mit einer Beschriftung darstellen, wenn Sie kein Label vorangestellt haben.

Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `tooltip` nicht überprüft.

Ein gesetzter Tooltip ist bei diesem Oberflächenelement nicht zwingend erforderlich, könnte aber – wenn er detaillierte semantische Informationen enthält – durchaus Sinn machen.

Beispiele für die visuelle Darstellung

One Label: TextView text text text text text text text text text text text
text text text text text text



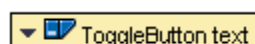
ToggleButton

Das Oberflächenelement ToggleButton stellt eine Drucktaste zum Umschalten am Bildschirm dar. Beim Betätigen des ToggleButton kann der Benutzer Anweisungen und Aktionen ausführen.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft tooltip überprüft, wenn die Eigenschaft text nicht gesetzt wurde.

Beispiel für die visuelle Darstellung



ToggleLink

Das ToggleLink-Oberflächenelement dient der Darstellung eines Hypertext-Links für die erweiterte Suche.

Beispiel für die visuelle Darstellung



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft tooltip überprüft, wenn die Eigenschaft text nicht gesetzt wurde.



Tray

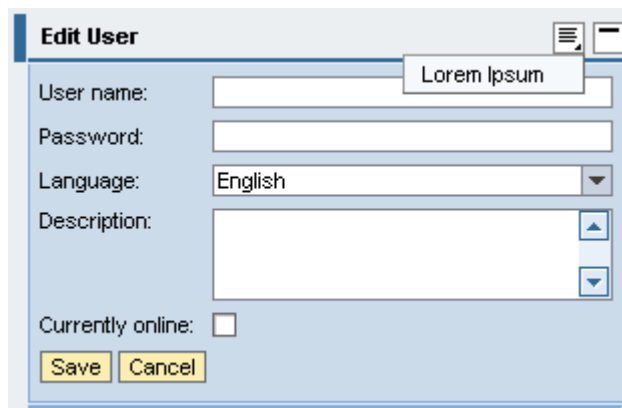
Das Tray-Oberflächenelement kann dazu verwendet werden, eine Menge von Oberflächenelementen unter einer gemeinsamen Überschrift anzuordnen und zählt deshalb wie die [Group \[Seite 18\]](#) zu den Oberflächenelement-Containern. Im Unterschied zur Group bietet es jedoch weitere Funktionen an, beispielsweise kann das Tray auf- bzw. zugeklappt werden.

Beispiele für die visuelle Darstellung

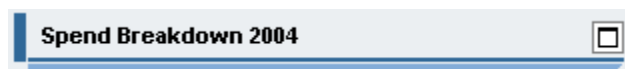
Tray mit Design plain



Tray mit Design fill und Menü



Zusammengeklapptes Tray mit Design transparent



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `accessibilityDescription` überprüft, wenn die Eigenschaft `caption` nicht gesetzt wurde.

Es findet keine Überprüfung auf die Eigenschaft `tooltip` statt.

Ein gesetzter Tooltip ist bei diesem Oberflächenelement nicht zwingend erforderlich, könnte aber – wenn er detaillierte semantische Informationen enthält – durchaus Sinn machen.



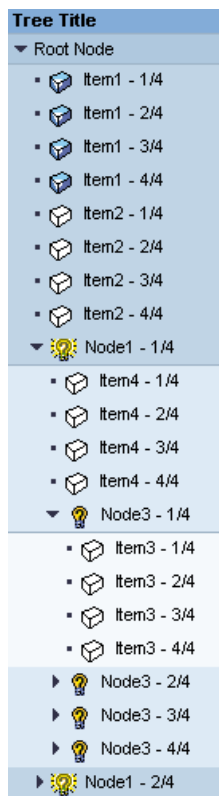
Mit Hilfe des Tree-Oberflächenelements können im Context definierte Hierarchien visualisiert werden. Die darzustellende Hierarchie wird zunächst im Context definiert. Es gibt zwei Möglichkeiten, diese Context-Struktur zu beschreiben:

- mit rekursiven Knoten, wenn zur Design-Zeit die Anzahl der Ebenen noch nicht bekannt ist

- mit nicht-rekursiven Knoten, wenn zur Design-Zeit bereits eine bestimmte Anzahl an Ebenen festgelegt werden kann

Ein Tree dient also der Navigation. Durch Anklicken eines Eintrags werden die damit verbundenen Daten an anderer Stelle angezeigt. Wenn Sie eine Baumstruktur visualisieren möchten, die zur Eingabe geeignet ist und darüber hinaus die Möglichkeit zum Scrolling anbieten möchten, verwenden Sie statt eines Tree eine hierarchische Tabelle, siehe auch [TreeByKeyTableColumn \[Extern\]](#) und [TreeByNestingTableColumn \[Extern\]](#). Ein Tree bietet nicht die Möglichkeit der Selektion von Einträgen - er kann nur mit der LeadSelection umgehen. Beachten Sie, dass dies visuell wie eine Selektion aussieht, es ist jedoch von der Performance her eine sehr teure Operation, so dass es in den meisten Szenarien eher hinderlich wäre.

Beispiel für die visuelle Darstellung



Das Tree-Oberflächenelement wird gegen den obersten Context-Knoten gebunden, der dargestellt werden soll.

Mit Hilfe von Knoten ([TreeNodeType \[Extern\]](#)-Elemente) bzw. Blättern ([TreeltemType \[Extern\]](#)-Elemente) wird festgelegt, welche der Unterknoten dargestellt werden sollen und vor allem, welche Context-Attribute auf diesen Unterknoten als Text bzw. Tooltip dargestellt sollen. Die Eigenschaft `dataSource` des `TreeNodeType`-Elements bzw. `TreeltemType`-Elements wird hierzu gegen den entsprechenden Context-Knoten gebunden, die Eigenschaften `text`, `tooltip` etc. werden gegen die entsprechenden Context-Attribute auf diesem Context-Knoten gebunden.

`TreeltemType`-Elemente können niemals Kinder haben und werden deshalb immer als Blätter angezeigt. Sie werden dann verwendet, wenn schon zur Design-Zeit feststeht, dass der entsprechende Knoten keine Kinder hat. Bei `TreeNodeType`-Elemente wird erst zur Laufzeit dynamisch bestimmt, ob es Kinder gibt.



Es können keine im Context definierten Hierarchiestufen bei der Darstellung des Oberflächenelements ausgelassen werden. Zum Beispiel muss zur Darstellung der Items für die in einem Context definierte Hierarchie *Customer* → *Orders*

→ *Items* auch ein gegen die Orders gebundenes `TreeNodeType`-Element existieren.

Bei allen Knoten, die nicht direkt unter dem Context-Wurzelknoten hängen, muss es sich um Nicht-Singleton-Knoten handeln, da in einem Baum alle Elemente angezeigt werden sollen, unabhängig von der Lead-Selection.

Sie können den Tree auch so binden, dass die `dataSource` des Trees auf einen strukturierten 1:1 Knoten bindet und die Element-Knoten des Trees sich erst darunter befinden. Dies ist für die Abbildung von Verzeichnisstrukturen notwendig. Bei rekursiven Trees können Sie die Rekursion gegen diesen 1:1-Knoten zeigen lassen. Dabei wird dieser dann beim Rendering übersprungen.



Um die Entwicklung [barrierefreier \[Extern\]](#) Anwendungen zu ermöglichen, wird im Rahmen des Syntaxchecks die Eigenschaft `tooltip` überprüft.



TriStateCheckBox

Dieses UI-Element ähnelt einer [CheckBox \[Seite 15\]](#), mit dem Unterschied, dass hier der angeklickte Zustand variabel ist:

- Option kann aktiviert (ausgewählt) sein
- Option kann nicht aktiviert (ausgewählt) sein
- Option ist un spezifiziert

Beispiele für die visuelle Darstellung

Ausgewählt

TriStateCheckBox

Nicht ausgewählt

TriStateCheckBox

Unentschieden

TriStateCheckBox