



Framework Studio 3.11.20

Neuheiten

inkl. Service-Release-Informationen

© 2017 Framework Systems GmbH

Alle Rechte vorbehalten. Die Software Framework Studio® und alle seine Teile inkl. Handbuch und Dokumentationen sind urheberrechtlich und markenrechtlich geschützt. Jede Verwendung in anderen als den gesetzlich und vertraglich zugelassenen Fällen bedarf der vorherigen schriftlichen Genehmigung des Herausgebers.

Dieses Dokument wurde mit größter Sorgfalt erstellt. Trotzdem können Fehler nicht vollständig ausgeschlossen werden. Für Mängel haftet die Framework Systems GmbH gemäß Ihren Allgemeinen Geschäftsbedingungen.

Technische Änderungen vorbehalten.

Das Dokument erhebt keine Ansprüche auf Vollständigkeit.

MS-Windows, MS-WindowsNT, MS-Windows2000, MS-Windows98, MS Windows XP, MS Windows Server 2003, MS Windows 7, MS SQLServer, MS-Word (Winword) und MS-Excel sind eingetragene Warenzeichen der Microsoft Corporation.

Framework Studio ist eingetragenes Warenzeichen der Framework Systems GmbH.

Crystal Reports ist eingetragenes Warenzeichen der Business Objects.

Oracle ist eingetragenes Warenzeichen der Oracle Corporation.

Inhaltsverzeichnis

Framework Studio 3.11.18 - 3.11.20 (04.04.2017).....	7
Wichtig: Alle Broker aktualisieren!	7
Use Saved Code Status	8
Korrigierte Fehler.....	8
Framework Studio 3.11.16 / 3.11.17 (24.02.2017).....	10
Wichtiges Rpository-Script ausführen!	10
Korrigierte Fehler.....	10
Framework Studio 3.11.15 (27.01.2017).....	12
Oracle Managed Provider – Produktiv-Freigabe	12
Vorteile	12
Konfiguration	12
Richtige Oracle-Version?	13
Neue Suche.....	14
Korrigierte Fehler.....	15
Framework Studio 3.11.14 (27.12.2016).....	17
Korrigierte Fehler.....	17
Framework Studio 3.11.13 (23.12.2016).....	18
Korrigierte Fehler.....	18
Framework Studio 3.11.12 (25.11.2016).....	20
Server-Culture im Publish-Wizard.....	20
Benutzerkontensteuerung (UAC).....	20
Development Host auf „localhost“	21
Probleme mit gelöschten Namespaces beheben	22
Korrigierte Fehler.....	23
Framework Studio 3.11.11 (28.10.2016).....	24
Print-Service – Owner im Druck-Job.....	24
CORS-Support für Services.....	24
Dokumentationssystem - Ausblenden von Controls.....	24
Dokumentationssystem - HTML-Export	25
Wizard-Optionen.....	25
Webapplikation	27
Problem-Lösungen	29
Hilfe-Url für eNventa ERP	30
Export mit FSConsole.....	30
Export mit PBE	31
Korrigierte Fehler.....	32
Framework Studio 3.11.10 (30.9.2016)	34

Disablen von Panels	34
Korrigierte Fehler	35
Framework Studio 3.11.9 (8.9.2016)	36
Korrigierte Fehler	36
Framework Studio 3.11.8 (26.08.2016)	37
Neuer Print-Service	37
Usability bei DBColumns	38
Publish2Go über FSConsole	39
Installation-Name im Runtime Supervisor	39
Neue Action Picture.SetImage()	39
Korrigierte Fehler	40
Framework Studio 3.11.7 (29.07.2016)	41
Korrigierte Fehler	41
Framework Studio 3.11.6 (24.06.2016)	42
Korrigierte Fehler	42
Framework Studio 3.11.5 (27.05.2016)	44
Neuer Development Broker ohne IIS	44
Hinweis zum Debugging von Development Broker und Service Hosts	44
Trace Information und Icons im Context-Menü	44
Client Start Page	45
GlobalObject / Performance in Service-Methoden	46
Korrigierte Fehler	47
Framework Studio 3.11.4 (29.04.2016)	49
Print-Service	49
Korrigierte Fehler	49
Framework Studio 3.11.3 (31.03.2016)	51
MLColumns benutzen FSROWID	51
ACHTUNG! Registerkarten komplett disabled	52
Print-Service	53
Korrigierte Fehler	53
Framework Studio 3.11.2 (26.02.2016)	56
HTML-Export für Component-Properties	56
Markieren der Components	56
Export-Dialog	57
Ausgabe	58
Korrigierte Fehler	59
Framework Studio 3.11.1 (29.01.2016)	60
Authentication Service 3.11.1 (Update optional)	60
Dynamische Gridspalten	60
Überarbeitung von Copy & Paste	61

Korrigierte Fehler.....	61
Framework Studio 3.11 Neuheiten	63
Änderungen bei Systemanforderungen	63
Update-Hinweise	63
FSROWID	63
Update der Datenbank.....	63
Neuer Typ FSSystemGuid	64
Änderungen in der Programmierung.....	65
Datenbank-Trigger.....	65
Fremd-Datenbanken / importierte Strukturen	65
Java-Client	66
Vereinheitlichung des Kommunikationsformats zwischen Broker und Client	66
Usability in der IDE	68
Lazy Global Objects und Safe Initialize	68
Copy & Paste.....	69
Beim Erstellen von Publish2Go Lizenzen exportieren.....	74
Verbesserung beim Sortieren von Gridspalten im Formdesigner	74
Usability im Service Release	75
Vererbung von Namespace-Referenzen.....	75
Datatype-Auswahl.....	75
Adjust moved Customizations.....	76
Warnings	77
Steuerelemente	78
Grid – Neues Event OnRowSelectionChanged.....	78
Browser – Erstellen von Snapshots	78
Package Manager	78
Publish & Publish2Go aus dem Package Manager heraus	78
Method Observer beim Import eines Service Release prüfen	78
Administrative Mode – Benutzersitzungen beenden	79
Auslagerung historischer Daten.....	80
Systemvoraussetzungen	80
Entwicklungs-Rechner	80
Anwendungs-Server	81
Print-Server	82
Authentication-Server	82
Client-Rechner	82
Java-Runtime	83
Crystal Reports Runtime.....	83
Crystal Reports Designer.....	83
Datenbank-Server	83

Framework Studio 3.11 Neuheiten



Oracle-Client	83
Korrigierte Fehler.....	84

Framework Studio 3.11.18 - 3.11.20 (04.04.2017)

Wichtig: Alle Broker aktualisieren!

Es gab Optimierungen bei der Broker-Kommunikation. Die Änderungen betreffen sowohl den Broker als auch den Java-Client.

Bei diesem Update ist es deshalb besonders wichtig, dass auch wirklich **alle Broker** – inkl. Verteil-Broker – **aktualisiert** werden. Wenn Probleme auftreten, könnte eine Lösung das Löschen des Java-Caches am Client-Rechner sein.

Die Änderungen dienen insbesondere der Stabilität der Anwendung bei schlechten Netzwerk-Verbindungen. Es konnte passieren, dass bei einem Timeout oder einer kurzzeitigen Verbindungs-Unterbrechung der Java-Client in einen invaliden Zustand geraten ist oder komplett blockiert war.

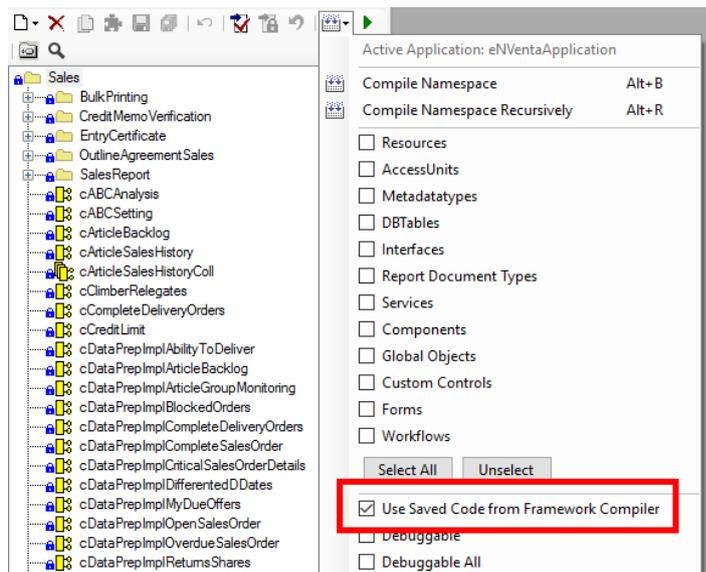
Weitere Informationen

Eine grundlegende Änderung besteht darin, dass der Java-Client für einen Request an den Broker jetzt die HTTP-Verbindung unterbricht, wenn in den letzten **20 Sekunden** weder Daten gesendet noch empfangen wurden. Für länger dauernde Requests, die z.B. bei komplexen Operationen wie Statistiken oder Massendruck auftreten, wird das Ergebnis des Requests durch die Fortschritts-Abfrage (Bitte Warten... – Dialog) vom Broker abgeholt. Damit verursacht die Broker-Client Kommunikation jetzt keine dauerhaft geöffneten HTTP-Verbindungen mehr.

Konnte der Request nicht verarbeitet werden, wird dieser erneut vom Client an den Broker gesendet. Der Broker erkennt doppelte Requests und gibt in so einem Fall denselben Response noch einmal zurück. Der Client **wiederholt** den Request jetzt bis zu **2-mal vollautomatisch**. So kann der Client kurzzeitige Unterbrechungen selbstständig ohne Benutzer-Interaktion lösen. Erst wenn diese 2 Versuche ebenfalls Fehlschlagen, wird ab dem 3. Versuch die gewohnte Meldung angezeigt, mit der der Benutzer weitere Versuche starten oder die Anwendung beenden kann. Die Versuche werden in der Java-Konsole dokumentiert.

Wird der Java-Client zur Entwicklungszeit **aus Framework Studio gestartet**, dann gilt der 20-Sekunden-Timeout NICHT. Ansonsten würde längeres Debuggen und Pausieren der Anwendung zu permanenten Kommunikations-Fehlern im Java-Client führen.

Use Saved Code Status



Die Checkbox "Use Saved Code from Framework Compiler" in den Compile-Einstellungen war bisher standardmäßig nach einer Installation von Framework Studio nicht gesetzt. Weiter wurde deren Status beim Start auf den zuletzt verwendeten Wert gesetzt.

Dieses Verhalten führte oft zu Verwirrungen. Deswegen wird der Status der Checkbox ab sofort nicht mehr persistent gespeichert und ist bei jedem Start von Framework Studio **aktiviert**.

Korrigierte Fehler

ID	Text
2017037206	Java-Client: Eine Combobox zeigt einen ungültigen Wert an (##123...). Wenn die Combobox-Liste keinen Eintrag besitzt und es wird dieser Wert gelöscht, dann gibt es eine Exception, die den kompletten Client blockiert.
2017037139	Java-Client: Das Aus- und wieder Einblenden von Controls im StackPanel kann zu Darstellungsfehlern führen.
2017037005	Package Manager: Der Button 'Import License' funktioniert nicht wenn der Ordner 'Licenses' selektiert ist.
2017026825	Package Manager: Beim Importieren eines Packages/Service-Releases mit einer Betriebssystemsprache ungleich "de-DE" wird eine Exception beim parsen eines Datums geworfen.
2017026813	Package Manager: Wird der Import eines Packages/Service-Releases durch den Benutzer abgebrochen (z.B. durch verneinen einer MessageBox), so werden Folgeprozesse (Method Observer Prüfungen) dennoch ausgeführt.
2017026585	Broker: Wenn beim Start einer Anwendung das Lizenz-Limit erreicht ist und die entsprechende Fehlermeldung angezeigt wird, dann wird diese Sitzung vom Broker nicht sauber abgeräumt. (Diese Sitzung verbraucht dabei aber keine zusätzliche Lizenz.)
2017026784	Framework: Das Klonen einer TextCollection kann zu Exceptions führen, wenn zuvor in der Quell-TextCollection Einträge entfernt wurden.
2017026781	IDE: Wenn man im Form-Designer nach dem Löschen einer Spalte ein Undo durchführt, dann kann es zu einer Exception kommen: System.InvalidOperationException: Aus einem DataGrid, das nicht an eine DataTable gebunden ist, können keine Daten gelesen werden.
2017037137	IDE: RowID-Update wirft Exception unter SQL Azure

2017037277	IDE: Versucht man im ServiceRelease eine Codestelle über den Find-/Replace Dialog zu ersetzen, wird angezeigt das der Codeeditor nicht editierbar ist.
2017037326	IDE: Wenn ein ServiceHost über den RunWizard lokal ausgeführt wird und CORS für diesen aktiviert ist, kann es beim Aufruf von Service-Methoden zu einer Exception kommen.
2017036833	Oracle Managed: Beim Compile-Wizard über den Package-Manager wird die Information über den Managed Provider nicht korrekt an die Unter-Prozesse übergeben. Diese verwenden deshalb den ODP-Provider.
2017036835	Oracle Managed: Mit dem Oracle Managed Provider wird bei Get DLLs aufgrund eines Fehlers im Oracle-Provider eine Exception geworfen.
2017037044	FSConsole: Unterstützt nun auch Publish2Azure

Framework Studio 3.11.16 / 3.11.17 (24.02.2017)

Wichtiges Repository-Script ausführen!

Aufgrund eines Fehlers in der Update-Routine ist es notwendig, ein Script auf dem Repository auszuführen.

Dieser Fehler kann Exceptions beim Check-In von Forms auslösen, welche schon seit einer sehr langen Zeit ausgecheckt sind. Das Script korrigiert die betroffenen Datensätze. Die Korrektur betrifft auch importierte Package-Versionen - insbs. auch **eNventa** und **eNventa WS**. Nach dieser Korrektur sind für importierte Packages – auch für Service Releases – keine gesonderten Aktionen mehr erforderlich.

WICHTIG! Bitte führen Sie die folgende Aktion auf allen Repositories aus, die bereits den Stand Framework-Studio 3.11 haben:

- Im Programm-Verzeichnis von Framework-Studio 3.11 das Programm **FrameworkStudioUpgradeDatabase** ausführen und am Repository anmelden.
- Es wird ein neues Script „2015.12.29 DocumentationFix“ angeboten. Dieses mit dem Button „**Update**“ ausführen.

Korrigierte Fehler

ID	Text
2017016411	Java-Client: Auf Mac OS X geht mit dem Java 8 Update 121 in modalen Dialogen kein Button-Klick mehr.
2017026718	JavaClient: Beim Laden von GridColumns mit extrem langen Texten kommt es zu erheblicher Wartezeit.
2017026528	Publish2Go: Wenn man bei einem Service-Host keinen Port angibt, dann wird beim Publish eine unspezifische Fehlermeldung angezeigt: System.FormatException - Die Eingabezeichenfolge hat das falsche Format.
2017026704	<p>IDE: Durch sehr umfangreiches Customizing steigt die Anzahl der Assemblies sehr stark an. Dies kann beim Compile der Workflows zu einem Compile-Fehler führen: CS1647 - An expression is too long or complex to compile. Der C-Sharp-Compiler produziert diesen Fehler, wenn die Anzahl der referenzierten Assemblies ca. 1.900 erreicht.</p> <p><i>Um die Anzahl der Referenzen zu reduzieren, kann man in der FrameworkStudio.exe.config einen Schalter ergänzen. Dieser unterdrückt beim Compile der Workflows die Referenzen auf die Report-Document-Types und auf die Metedatatypes.</i></p> <pre><add key="FixworkflowReferences" value="true"/></pre> <p><i>Achtung! Bei einer Update-Installation von Framework Studio geht diese Einstellung verloren.</i></p>
2017026475	IDE: Wenn man im Customizing oder in einer Ableitung an einem TreeView die Eigenschaft InitialDepth ändert, dann führt dies zu einem Compile-Error.
2017026477	IDE: TreeView InitialDepth kann nicht auf den Basis-Wert zurückgesetzt werden. Beim Leeren des Properties gibt es eine Meldung "Ungültiger Eigenschaftswert"
2017016404	IDE: Im Check-In-Fenster kann im Comment-Feld der Text nicht mit Strg+A markiert werden.

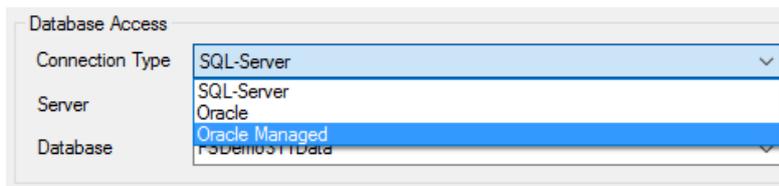
2017026583	IDE: In einer DBTable können Werte in den Feldern "Name" und "Metadatatype" eingegeben werden, obwohl keine Spalte existiert.
2017026521	IDE: Im MessageBox- und Exception-Wizard ist es möglich, in TextCollections zu suchen. Im neuen SuchFenster ist es nicht mehr möglich, über alle TextCollections zu suchen.
2017026733	IDE: Der Doppelklick auf Parents von Suchergebnissen funktioniert nicht.
2017026779	IDE: Wenn bei DBTables die Namen länger als 30 Zeichen sind, dann führt dies beim Compile zu Warnings und Errors. Handelt es sich dabei um importierte DBTables, dann dürfen diese Absicherungen nicht greifen, weil die importierten Namen vom Entwickler nicht beeinflusst werden können.
2017026513	FSConsole: Wird mit der FSConsole ein Publish-Vorgang gestartet und es wird dafür die Meldung für die Benutzerkonten-Steuerung angezeigt, dann kann es sein, dass die Kommandozeilen-Parameter nicht korrekt weitergegeben werden. Wenn ein Parameter – z.B. für die LogDatei – ein Leerzeichen beinhaltet, dann wird das nicht verstanden und es kommt zu Fehlern.

Framework Studio 3.11.15 (27.01.2017)

Oracle Managed Provider – Produktiv-Freigabe

Ab Framework Studio 3.11.15 steht der Oracle Managed Provider für den produktiven Einsatz zur Verfügung. Sollte der Managed Provider Probleme verursachen, kann als Fallback jederzeit auf den ODP-Provider („Oracle“) zurückgeschaltet werden.

Der neue Provider **Oracle Managed** kann im Connection Type neben **SQL-Server** und **Oracle** (Oracle ODP) ausgewählt werden.



Alle bestehenden Oracle-Connections verwenden automatisch den normalen Oracle ODP-Provider. Um den neuen Managed Provider einzusetzen, müssen die Connections, z.B. in Settings und Registered Databases, manuell angepasst werden.

Die Anpassung von Registered Databases kann dazu führen, dass diese in älteren FS-Versionen nicht mehr verwendet werden können. Die aktuellen Service-Releases von FS 3.8 und FS 3.10 beinhalten aber eine entsprechende Fallback-Logik.

Bei Connections-Strings – z.B. in Config-Dateien – muss als **FSProviderType** der Wert „**OracleManaged**“ angegeben werden. Beispiel:

```
<Connections>  
  <cnMVDData ConnectionString="FSProviderType=OracleManaged;Server=Nv2"  
</Connections>
```

Vorteile

Der neue Oracle Managed Provider bringt einige Vorteile.

- 1.) **Kein Oracle-Client nötig:** Framework Studio bringt die notwendige DLL bereits mit. Es muss kein extra Oracle-Client mehr installiert werden.
- 2.) **Performance:** Im Vergleich zum ODP-Provider hat der Managed Provider insbes. bei der Verarbeitung großer Datenmengen eine deutlich bessere Performance.
- 3.) **32-Bit / 64-Bit:** Der Managed Provider ist nicht mehr Bit-spezifisch.

Konfiguration

Einfache Verbindungen mit Server und SID können normal über die Connection eingestellt werden. Framework Studio erstellt daraus, wie bisher auch, automatisch einen TNS-String.

Wird der Einsatz von ora-Dateien (tnsnames.ora, sqlnet.ora, ldap.ora) nötig, muss dies entsprechend konfiguriert werden. Oracle unterstützt im Managed Provider verschiedene Wege für die Konfiguration.

- 1.) Application.config
- 2.) Verweis auf ora-Dateien in der Application.config
- 3.) ora-Dateien im aktuellen Programm-Verzeichnis.
- 4.) tnsnames.ora / sqlnet.ora im Ordner **%TNS_ADMIN%**.
ldap.ora im Ordner **%LDAP_ADMIN%**

- 5.) tnsnames.ora / sqlnet.ora im Ordner %ORACLE_HOME%\network\admin\
ldap.ora im Ordner %ORACLE_HOME%\ldap\admin

Die Wege 1 und 2 können bei Framework Studio nicht (so einfach) genutzt werden, weil die config-Dateien (z.B. beim Publish) neu erzeugt werden. Für den Weg 3 könnten die Dateien als Ressource abgebildet werden. Dies zieht dann aber nur nach dem Publish.

Empfohlen werden 4 oder 5. Mit diesen Wegen wird der komplette Rechner konfiguriert und man hat so praktisch denselben Komfort wie mit dem ODP-Provider. Bei den anderen Wegen erfolgt die Konfiguration nur im aktuellen Programm-Kontext und muss somit auch in jedem Kontext extra vorgenommen werden.

Es müssen die entsprechenden SYSTEM-Umgebungsvariablen eingerichtet werden. Dies kann über die Windows Systemeigenschaften (Registerkarte Erweitert) oder über die Eingabeaufforderung erfolgen. Die Eingabeaufforderung als Administrator starten und z.B. den folgenden Befehl ausführen:

```
setx ORACLE_HOME "C:\Oracle" /m
```

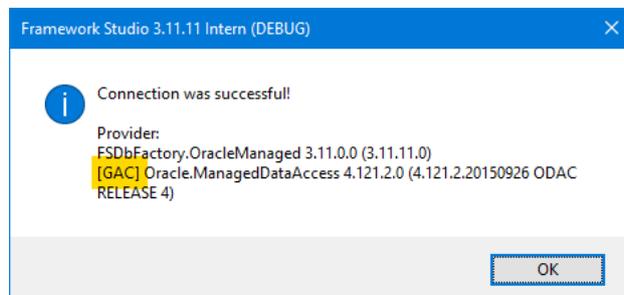
Anschließend muss das System neu gestartet werden.

Richtige Oracle-Version?

Framework Studio bringt eine eigene Oracle.ManagedDataAccess.dll mit, welche auch beim Publish in die entsprechenden Ordner gepackt wird.

Wenn ein Oracle-Client mit ODP-Provider installiert ist, dann kann es sein, dass diese DLL bereits im Global Assembly Cache (GAC) des .NET Framework registriert ist. Diese DLL aus dem GAC hat immer die höchste Priorität und sie zieht vor der mit Framework Studio ausgelieferten DLL. (Dieses Verhalten des .NET Framework lässt sich nicht beeinflussen.)

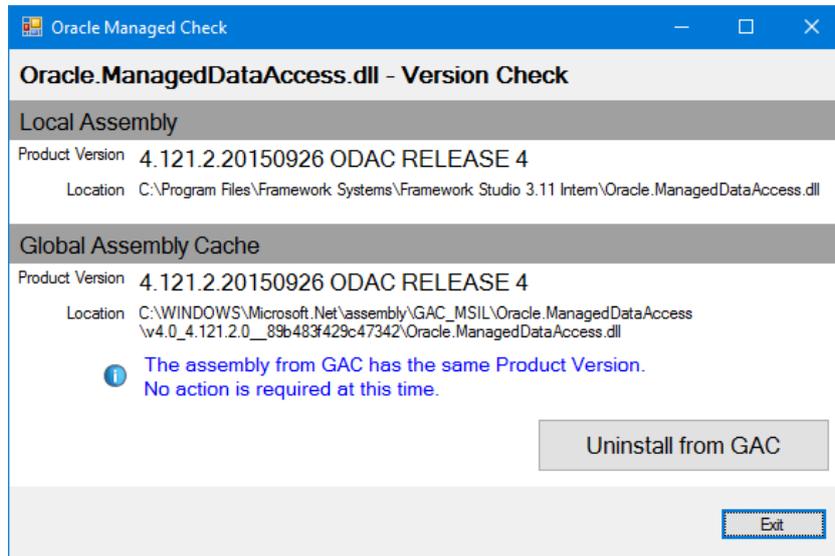
Bei der Test-Connection wird angezeigt, wenn die DLL aus dem GAC gezogen wird.



Solange die korrekte Version des Oracle-Client installiert ist und die Versions-Nummer stimmt (aktuell Oracle 12c Release 4), ist dies kein Problem. Wenn jedoch eine veraltete oder nicht kompatible Version installiert ist, dann muss dies gelöst werden.

Im Framework Studio Programm-Verzeichnis gibt es ein Programm **FSDbFactory.OracleManaged.Check.exe**. Damit kann die Version geprüft und bei Bedarf die DLL aus dem GAC

entfernt werden kann. Dieses Programm steht auch im Publish2Go-Verzeichnis zur Verfügung.



ACHTUNG! Das Entfernen der DLL aus dem GAC kann ggf. Probleme in anderen Programmen verursachen, wenn diese Oracle.ManagedDataAccess verwenden.

Vor dem Entfernen legt das Programm im aktuellen Verzeichnis einen Ordner „GACBackup“ an. Dort wird die DLL aus dem GAC abgelegt. So kann sie bei Problemen diese Aktion jederzeit rückgängig gemacht und die DLL wieder in den GAC installiert werden.



Neue Suche

In der IDE wurde eine neue Suche eingeführt.

Die ursprüngliche Namespace-Auswahl wurde entfernt. Um in einem Namespace zu suchen, muss man von der Namespace-Ansicht aus die Suche öffnen.

Die Packages in denen gesucht werden soll, können nun aus einer Liste ausgewählt werden. Es können auch mehrere Packages ausgewählt und so gleichzeitig durchsucht werden.

Korrigierte Fehler

ID	Text
2017016179	Package-Manager: Wenn in einer Package-Hierarchie mehrere Package-Versionen denselben Assembly-Prefix haben, dann kommt es zu Compile-Fehlern. Diese Situation müsste schon vor dem Compile erkannt und ein entsprechender Fehler ausgegeben werden.
2016125942	Java-Client / Broker: Set Enabled-Actions für Tab-Pages werden nicht verdichtet. Wenn mehrere widersprüchliche Actions (Disable und Enable) ausgeführt werden, werden diese auch alle vom Client abgearbeitet. Dies kann unerwartete Effekte z.B. bei der Fokus-Steuerung auslösen. Im Java-Client werden Enable-TabPage-Actions zum falschen Zeitpunkt ausgeführt. Diese müssen zusammen mit den anderen Enabled- und Visible-Actions vor den SetFocus-Actions ausgeführt werden.
2016126001	Java-Client: Es ist möglich in eine ComboBox ohne Einträge und ohne Free Text Input Texte einzufügen. Dies führt zu einer Exception.
2017016250	Java-Client: Eingabefelder, welche aufgrund des FormatPatterns Nachkommastellen ausblenden, melden den angezeigten Wert als einen durch den Benutzer eingegebenen Wert an den Broker zurück.
2017016087	Print-Service: Beim Druck von Crystal-Reports kann es passieren, dass Papier-Formatierungen nicht korrekt gezogen werden. In der Folge werden z.B. Etiketten in der falschen Größe oder sogar verdreht gedruckt.
2016125901	Auth-Service: Wird mit Domain Login gearbeitet, so läuft der Arbeitsspeicherbedarf langsam hoch.
2016115412	Oracle: FSRowID-Umstellung - Materialized Views können nicht geändert werden und dürfen deshalb bei der Umstellung nicht berücksichtigt werden. Bei temporären Tabellen darf kein Index für die FSROWID-Spalte angelegt werden.
2017016185	SQL-Server: Wenn in der Datenbank eine Spalte auf Uniocode steht und in der DBTable die Länge der Spalte mit > 4000 angegeben ist, dann erzeugt der Update-Assistent einen Fehler. Eine Spalte soll z.B. auf nvarchar(5000) umgestellt werden.
2016125926	Publish: Auf einem Rechner werden mehrere Services mit einem Setting gepublished. Passen mehrere Server-Namen auf den Rechner (z.B. Localhost, Rechnernamen, IP-Adresse) und es werden für diese Rechner unterschiedliche Service-Directories angegeben, dann kommt es in der Folge zu Problemen beim Revoke. 2.) Wenn bei mehreren Servern dasselbe Verzeichnis angegeben wird, dann führt dies zu einer Exception beim Publish. 3.) Enthält ein Service-Host in seiner Base-Address einen festen Host-Name, dann wird dies im Publish2Go nicht richtig berücksichtigt.
2016125977	FSBrokerHost: Das Anzeigen von Reports im rtf- oder im rpt-Format funktioniert nicht.
2016115348	IDE: Der Development-Broker startet nicht, wenn in der URL Sonderzeichen - wie z.B. Umlaute - enthalten sind (Datenbank, Package, Version, Benutzer)
2016125839	IDE: Änderungen an einer Dokumentation oder der Caption hat einen Eintrag in die "Compile Changes" Liste zur Folge
2015110263	IDE: Wird eine DBTable oder deren Column umbenannt, so verweisen an den Components verknüpfte VirtualColumns immer noch auf eine Spalte mit dem alten Namen. Es gibt keinen Kompilierfehler. <i>Jetzt wird eine entsprechende Compile-Warning erzeugt.</i>

Framework Studio 3.11 Neuheiten



2016125858	IDE: CustomControls können nicht kopiert werden. Daher kann es beim Einfügen von Forms mit CustomControls zu Fehlern kommen.
2016115356	IDE: Beim Kopieren von Controls werden zugehörige FormMethoden und WorkflowEvents nicht im MergeDialog angeboten.
2017016199	IDE: DefaultControlStyles können unter Oracle nicht customized werden.
2016125887	IDE: Wenn in einer Ableitung oder einem Customizing eines Forms von einem geänderten Control der Container nicht geändert ist, dann ziehen diese Änderungen zur Laufzeit nicht. Entstehen kann so eine Situation, wenn in der Basis des Foms das Control in einen anderen Container verschoben wird.
2017016156	Dokumentationssystem: In der HTML-Hilfe werden für Suchbegriffe aus mehreren Wörtern keine Ergebnisse zurückgeliefert.

Framework Studio 3.11.14 (27.12.2016)

Korrigierte Fehler

ID	Text
2016125992	<p>Beim Repository-Update wird das System-Package nicht korrekt angelegt. In der Folge gibt es Fehlermeldungen beim Package-Update. Im SytemPackage wird eine Package-Version "SystemVersion_11.11.0" angelegt anstatt "SystemVersion_3.11.0".</p> <p><i>Wenn der Fehler bereits aufgetreten ist, dann muss das Repository von Hand korrigiert werden. Setzen Sie sich in diesem Fall unbedingt mit dem Support in Verbindung.</i></p>

Framework Studio 3.11.13 (23.12.2016)

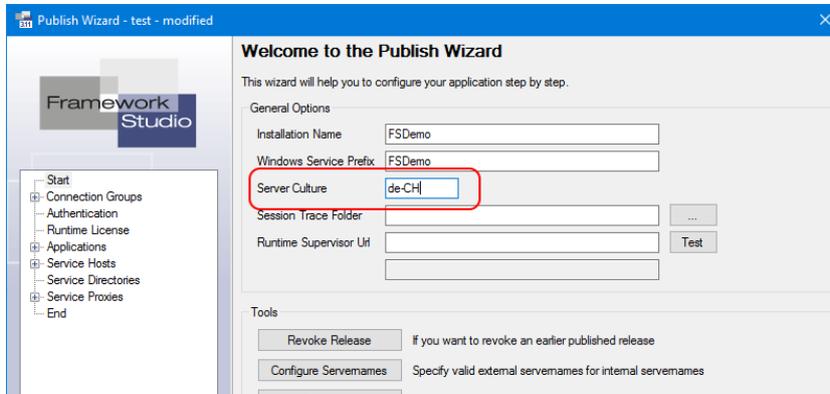
Korrigierte Fehler

ID	Text
2016115497	Java-Client: Bei der automatischen Größenanpassung eines Forms wird dieses fokussiert. (Das führt zu Folgefehlern im Client Side Customizing: Tastatureingaben und Shortcuts werden nicht geblockt).
2016125753	Java-Client: An Controls, die auf einem StackPanel platziert sind, können keine Tooltips über die Client-API (z.B. Client Side Customizing) gesetzt werden.
2016104766	Java-Client: Wird auf einen Tree, bei dem Nodes nachgeladen werden können, ein ExpandAll für ein bestimmtes Level abgesetzt, so kommt es zu folgender Exception: <i>Java-Exception: null</i> <i>at java.util.ArrayList\$Itr.checkForComodification(Unknown Source)</i> <i>at java.util.ArrayList\$Itr.next(Unknown Source)</i> <i>at com.jidesoft.grid.TreeTableModel.buildRows(Unknown Source)</i> <i>at com.jidesoft.grid.TreeTableModel.a(Unknown Source)</i> <i>at com.jidesoft.grid.TreeTableModel.refresh(Unknown Source)</i> <i>at FrameworkSystems.FSJavaClient.UIWrapper.UITreeView.</i> <i>ExpandTreeView(UITreeView.java:360)</i>
2016125883	Broker: Wird im unsichtbaren Bereich eines TreeViews ein Knoten verschoben, dann kann dies zu Exceptions oder Anzeige-Fehlern führen. Fehler treten bei noch nicht aufgeklappten oder bei noch nicht vollständig an den Client übertragenen Listen auf.
2016125590	Package-Manager: Beim Import eines Packages in ein Oracle-Repository dauert das Disablen und Enablen der Constraints sehr lange.
2016125840	Package-Manager: Ein Publish über den Package-Manager ist nicht mehr möglich. Beim Öffnen des Publish-Wizards erscheint folgende Fehlermeldung: <i>System.NullReferenceException: Der Objektverweis wurde nicht auf eine Objektinstanz festgelegt.</i> <i>bei FrameworkSystems.FrameworkStudio.General.DevObjects.RunWizard.</i> <i>RunWizardForm.setTitle()</i>
2016115220	Runtime License Manager: Der Button zum Speichern einer Lizenz wird nicht aktiv, wenn im Access Unit Tree eine Änderung vorgenommen wird, bevor eine andere Lizenz ausgewählt oder eine andere Eigenschaft geändert wird.
2016115436	IDE: Wenn die IDE im Maintenance-Mode gestartet wird, und wird z.B. bei Get Latest CheckIns ein Neustart ausgeführt, dann geht der Maintenance-Mode verloren.
2016115440	IDE: In Package-Konstellationen mit Kontext-Versionen kann es bei der Arbeit mit Grid-Controls zu folgender Laufzeit-Fehlermeldung kommen: <i>getGridView: The Method getGridView has to be overridden in the code generation of the form.</i> Grund ist ein Fehler in der Code-Generierung.
2016115515	IDE: Beim Kopieren von Components wird eine Exception geworfen wenn sich Sonderzeichen in der Description befinden.

2016125759	IDE: Im Meldungstext "The version of this element in your workspace is out of date [...]" werden ein falsches Checkin-Datum und ein falscher Checkin-Benutzer angezeigt, wenn es sich um einen Customizing Vorgang handelt.
2016125534	Add-Ins: Wunsch: Das Package-Merge Add-In soll eine Möglichkeit bieten, zu prüfen, durch welche anderen Benutzer benötigte Elemente ausgecheckt sind.
2016125568	Add-Ins: Wunsch: Das Package-Merge Add-In soll in Methoden, die im Ziel bereits existieren alle out-Parameter sowie den Rückgabewert initialisieren, um Kompilierfehler zu unterbinden.
2016125584	Add-Ins: Beim Package-Merge werden Änderungen an Controls nicht übernommen, wenn das Zielform keine Customization ist.

Framework Studio 3.11.12 (25.11.2016)

Server-Culture im Publish-Wizard



Im Publish- und im Run-Wizard kann jetzt bei den General Options eine Server-Culture angegeben werden. Diese Angabe zieht sowohl in den Applications als auch in allen Service-Hosts. Damit kann die im Betriebssystem eingestellte Culture übersteuert werden. Besonders praktisch ist dies im Run-Wizard, weil so ohne Betriebssystem-Konfiguration serverseitig mit abweichenden Regions-Einstellungen getestet werden kann.

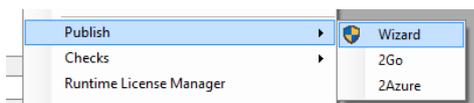
Bisher konnte die Server-Culture lediglich an der Application angegeben werden. Diese Möglichkeit besteht weiterhin und sie übersteuert die General Options. Es ist jedoch zu empfehlen, wenn, dann die Einstellung in den General Options zu verwenden. So arbeitet das gesamte System konsistent mit denselben Regions-Einstellungen. Neben der Formatierung von Zahlen- und Datums-Werten – z.B. bei der **Erzeugung von Crystal Reports** – beeinflusst die Culture auch einige kalendarische Funktionen – wie z.B. die **Berechnung der Kalenderwoche**.

Wichtig: es handelt sich hierbei um die Serverseite. Wenn es um die Formatierung von Werten für den Client geht, muss die Client-Culture verwendet werden. Diese wird vom Java-Client gemeldet.

Benutzerkontensteuerung (UAC)

Die Framework Studio IDE benötigt jetzt keine Administrator-Berechtigungen mehr. Das vereinfacht das Arbeiten bei aktivierter Benutzerkontensteuerung (UAC).

Mit dem neuen Development Host, der mit Framework Studio 3.11.5 eingeführt wurde, werden durch den Entwickler keine Änderungen mehr am IIS vorgenommen und damit braucht der Entwickler auch keine erweiterten Rechte mehr. Lediglich der Publish-Wizard, der Dienste installiert und den IIS konfiguriert, benötigt diese Rechte.

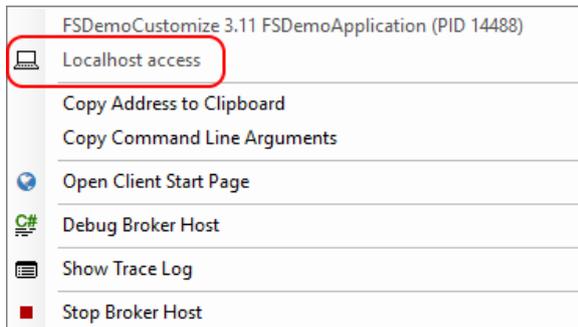


In einer späteren Version wird dies auch noch auf den Package-Manager und weitere Programme ausgeweitet.

Development Host auf „localhost“

Weil mit Framework Studio die Development-Hosts (Broker, Services) jetzt mit eingeschränkten Rechten gestartet werden, können diese im Regelfall auch nur noch vom eigenen Rechner mit „localhost“ angesprochen werden. Über Berechtigungen kann dies beeinflusst werden.

Im Menü des Development-Hosts wird durch einen Eintrag angezeigt, wie auf den Broker oder den Service zugegriffen werden kann:



Localhost access

Es kann nur lokal vom eigenen Rechner zugegriffen werden. Dabei muss unbedingt für den Hostname „localhost“ verwendet werden. Mit dem eigenen Rechnernamen oder der eigenen IP-Adresse funktioniert der Zugriff nicht. Der http-Endpoint wird mit einer URL `http://localhost:Port/...` registriert.

Intranet / Domain access (+)

Es kann nur innerhalb des Netzwerkes zugegriffen werden – der Host akzeptiert Anfragen, die an den eigenen Rechnernamen adressiert sind. Der http-Endpoint wird mit der URL `http://+:Port/...` registriert.

Public access (*)

Es kann von überall aus zugegriffen werden – der Host akzeptiert auch Anfragen mit einem fremden Host-Namen in der URL, z.B. einer Web-Adresse. Der http-Endpoint wird mit der URL `http://*:Port/...` registriert.

Wird der Development-Host ohne Administrator-Rechte gestartet, dann erhält man standardmäßig nur **Localhost access**. Wenn ein erweiterter Zugriff benötigt wird, dann kann man Framework Studio gezielt als Administrator starten. Dadurch erhält dann auch der Development-Host erweiterte Rechte und es kann ein **Intranet / Domain access** oder **Public access** erfolgen.

Besser ist es jedoch, auf dem Entwicklungs-Rechner entsprechende Rechte zu konfigurieren. Starten Sie die Eingabeaufforderung als Administrator. Mit dem folgenden Befehl kann einem User das Recht zur Registrierung eines öffentlichen http-Endpoints eingeräumt werden:

```
netsh http add urlacl url=http://+:8080/ user=Domain\User
```

In der Regel reicht ein „+“ für den Intranet / Domain access. Ist Public access gefordert, dann muss stattdessen ein „*“ verwendet werden.

Der Port kann in Framework Studio unter Tools / Options bei der Einstellung Broker Port bzw. Host Port eingestellt werden.

Um allen Benutzern Zugriff zu erlauben – wenn z.B. auf einem Terminal-Server entwickelt wird – kann user=Jeder bzw. user=Everyone verwendet werden.

Mit dem folgenden Befehl kann die Berechtigung wieder entfernt werden:

```
netsh http delete urlacl url=http://+:8080/
```

Probleme mit gelöschten Namespaces beheben

Wird im Basis-Package ein Namespace gelöscht, dann sind die im Customizing-Package in diesem Namespace angelegten Elemente nicht mehr sichtbar. Beim Compilieren kann das ein Fehler verursachen. Um solche Fehler zu beheben wurde im ‚Maintenance Mode‘ unter Tools der Menüpunkt ‚Adjust deleted namespaces‘ geschaffen. Mit diesem Dialog ist es möglich die Elemente einem neuen Namespace zu zuweisen oder sie zu löschen.

Korrigierte Fehler

ID	Text
2016104900	<p>Java-Client: (1) Combobox und Listbox feuern for jedem OnSelectionChanged -Event zusätzlich ein OnLeave-Event.</p> <p>(2) Ein einer Listbox wird beim OnEnter-Event ein zusätzlichen OnLeave-Event gefeuert.</p> <p>(3) Wenn ein Grid ein OnRowClick-Event hat, dann wird unmittelbar nach einem OnEditingStarting-Event ggf. ein zusätzliches OnEditingStopped-Event ausgelest.</p>
2015067371	<p>Java-Client: Bei einem TreeNode kann es ab dem 2. Doppelklick passieren, dass der TreeView fälschlicherweise den Fokus erhält obwohl dieser in ein anderes Form gesetzt werden soll. Das tritt z.B. auf, wenn so mit einem vorgeschalteten modalen Dialog ein Form geöffnet wird (z.B. eNventa ERP - Kasse)</p>
2015015335	<p>Java-Client: Wenn der Fokus in einer Gridzelle im Bearbeitungsmodus ist und man über einen Menüpunkt einen Dialog öffnet, so ist der Fokus nach dem Schließen des Dialogs manchmal nicht mehr im ursprünglichen Frame.</p>
2016115285	<p>Java-Client: werden in einem ColumnChanged modale Forms angezeigt, dann wird eine währenddessen am Grid aufgerufene SetCell-Action nicht korrekt ausgeführt.</p>
2016115326	<p>Java-Client: Der Java-Client erzeugt Darstellungsfehler, wenn z.B. zwischen Remote-Desktop und lokaler Anmeldung gewechselt wird.</p>
2016115112	<p>Broker: Beim Beenden eines Brokers wird eine Exception in das Event-Log geschrieben: "Error cleaning up SessionInfo!"</p>
2016115225	<p>Oracle: Wenn in einer Tabelle für die FSROWID-Spalte ein Primary-Key-Constraint definiert ist, dann möchte die Routine einen neuen Index anlegen, was zu einer Fehlermeldung führt.</p>
2016115351	<p>Publish: Beim Publish eines ServiceHosts kann es zur folgenden Exception kommen, wenn kein Passwort hinterlegt ist und im gespeicherten Setting auch noch nie eines hinterlegt war:</p> <p><i>System.ArgumentNullException: Der Zeichenfolgenverweis ist nicht auf einen Zeichenfolgeninstanz festgelegt. Parametername: s bei System.Text.Encoding.GetBytes(String s) bei FrameworkSystems.FrameworkStudio.Base.FrameworkCrypto .Encrypt(String text, Byte[] key, Byte[] iv) bei FrameworkSystems.FrameworkStudio.General.DevObjects.ServiceHost. ServiceHostRecord.WriteConfigFile(String fileName, String relativFileName, Boolean developmentBroker)</i></p>
2016115042	<p>Package-Manager: Beim Import eines Service-Release werden Warnings des Method Observers in gewissen Fällen mehrmals angezeigt.</p>
2016115099	<p>IDE: Beim Customizing einer Textcollection werden doppelte Einträge nicht korrekt behandelt. Wenn die ID eines Textes mit dem Basis-Package kollidiert, dann gibt es keine Kompilier-Warnung. Stattdessen führt dies zu einer Laufzeit-Fehlermeldung. Außerdem wird der Basis-Eintrag mit der betroffenen ID nicht im Costomizing angezeigt.</p>
2016115163	<p>IDE: Nach dem Ändern eines ControlStyles ist das Speichern nicht mehr möglich.</p>
2016094528	<p>IDE: Wird an einem Component-Property das Query-Mapping entfernt und löscht man direkt anschließend die Query-Column, dann wird das Property mit gelöscht.</p>
2016115299	<p>IDE: Beim Mergen von Forms kann es vorkommen das WorkflowEvents doppelt angelegt werden.</p>

Framework Studio 3.11.11 (28.10.2016)

Print-Service – Owner im Druck-Job

Bei Crystal-Reports wird jetzt der Benutzer-Name mit in die Druck-Warteschlange übergeben. So kann von entsprechenden Druckern dieser Name interpretiert und die Druck-Jobs bestimmten Benutzern zugeordnet werden.

In der Drucker-Warteschlange hat der Druck-Job jetzt folgende Bezeichnung:

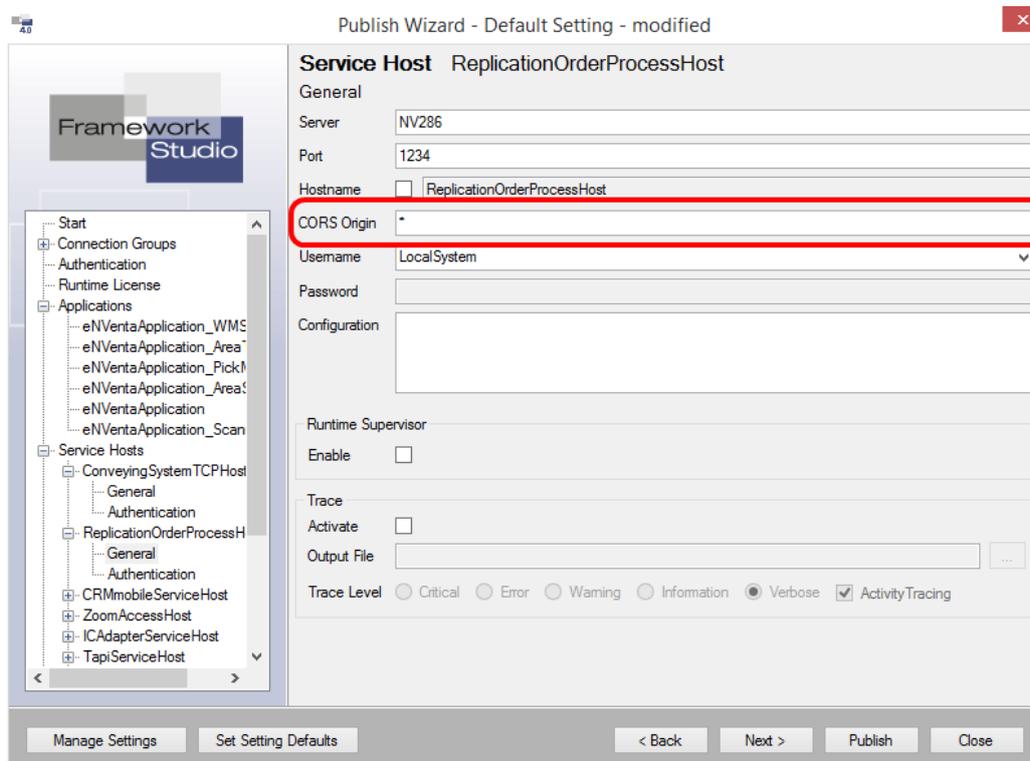
```
CrystalReports - <Owner> - <ID> - <ReportName>
```

Beim Owner handelt es sich um den eNventa Benutzer-Namen.

CORS-Support für Services

Wenn ein Framework Studio Webservice direkt per Javascript aus einem Browser angesprochen wurde, gab es bisher keine Möglichkeit CORS zu unterstützen und den entsprechenden HTTP-Header „Access-Control-Allow-Origin“ and den Response anzuhängen.

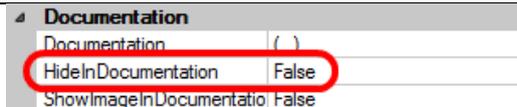
Der Run/Publish/Publish2Go Wizard bietet nun diese Möglichkeit:



Der angegebene Wert wird global am Service registriert und bei jedem Response ausgegeben. Eine lokale Anpassung des Headers in jeder Service-Methode ist nicht nötig.

Dokumentationssystem - Ausblenden von Controls

An Controls gibt es einen neuen Schalter „HideInDocumentation“, mit dem aktiv Control-Dokumentationen beim Exportieren ausgeblendet werden können.



Die kann z.B. dann nützlich sein, wenn in einem abgeleiteten Form Controls aus der Basis nicht in die Dokumentation gerendert werden sollen.

Dokumentationssystem - HTML-Export

Framework Studio ist nun in der Lage, die gesamte Dokumentation als HTML 5 Webapplikation zu generieren. Dadurch ist es nicht mehr zwingend nötig, eine Online-Hilfe mittels Doc2Help aus einem Word-Dokument zu generieren, was ab gewissen Größen problembehaftet ist.

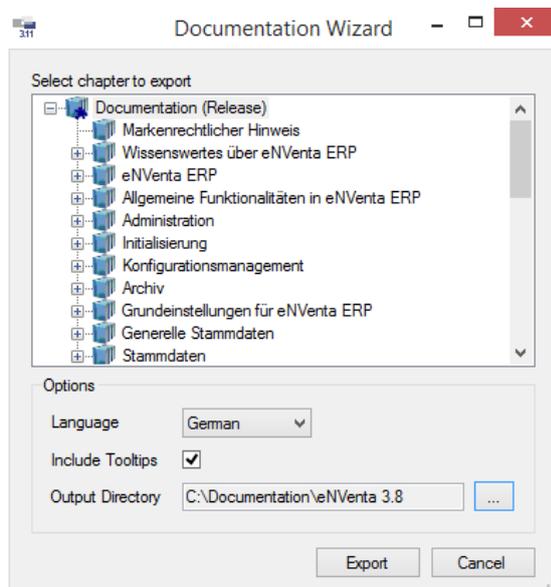
Dieser Export-Modus exportiert die gesamte Dokumentation oder ein ausgewähltes Kapitel als Webapplikation, welche auf einem beliebigen Webserver (z.B. IIS, Apache, Nginx) gehostet werden kann und somit von nahezu allen Endgeräten mit Webbrowser erreichbar ist.

Somit können jetzt ab eNVenta 3.6 auch in Customizing-Packages selbstständig Dokumentationen exportiert werden. Es ist aber unbedingt notwendig, den aktuellsten Service-Release von eNVenta 3.6 bzw. 3.7 zu verwenden. Die neue Hilfe-URL ist mit älteren Versionen von eNVenta nicht kompatibel (siehe auch →**Hilfe-Url für eNVenta ERP**).

Der Modus HTML (Properties) ist in diesem Export-Modus integriert (siehe Technische Dokumentation). Es werden alle Component-Properties von Components, die mit dem Marker **{ExportDoc}** markiert sind, exportiert.

Der Wizard für den Export der Dokumentation als HTML Webapplikation befindet sich im Menü **Tools** → **Documentation** → **Export** → **HTML**.

Wizard-Optionen



Die Baumansicht bietet die Struktur des Release-Knotens an. Hier muss ausgewählt werden, welches Kapitel exportiert werden soll. Wird der Release-Knoten selbst ausgewählt, wird die gesamte Dokumentation, die im Framework Studio hinterlegt ist, exportiert.

Language

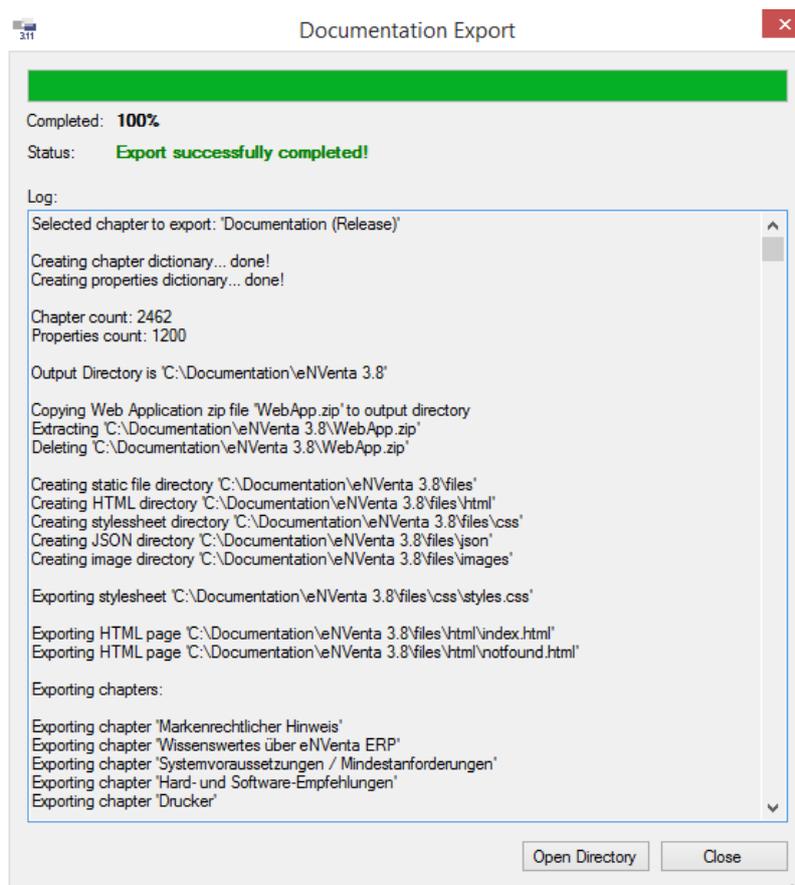
Definiert die Sprache, für die die Dokumentation exportiert werden soll. Dies ändert auch die Sprache der automatisch generierten Screenshots.

Include Tooltips

Ist an einem Steuerelement eines Formulars ein Tooltip hinterlegt, wird dieser mit in die Dokumentation des Steuerelements aufgenommen.

Export-Vorgang

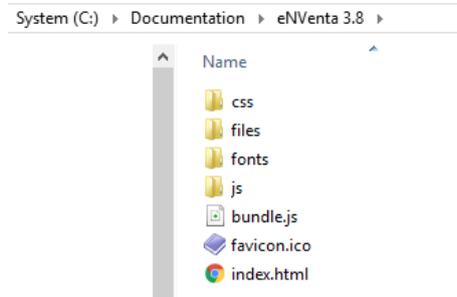
Für den eigentlichen Export-Vorgang wird ein Dialog mit Statusinformationen bereitgestellt. Über den Button „Abort“ kann der Vorgang jederzeit abgebrochen werden.



Treten beim Export-Vorgang Infos, Warnungen oder Fehler auf, werden diese detailliert in das Ausgabefenster protokolliert und der Statustext ändert ggf. seine Farbe (orange für Warnungen, rot für Fehler).

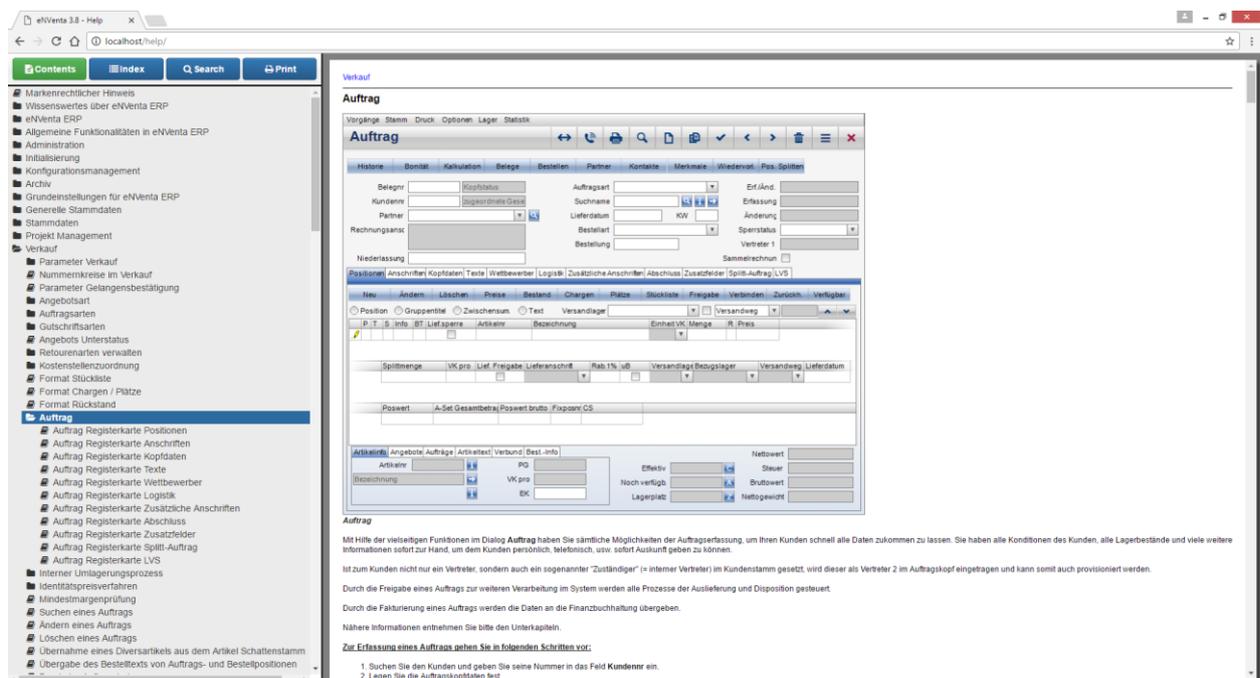
Webapplikation

Nach dem Export-Vorgang befindet sich die Webapplikation im zuvor ausgewählten Ausgabeordner.



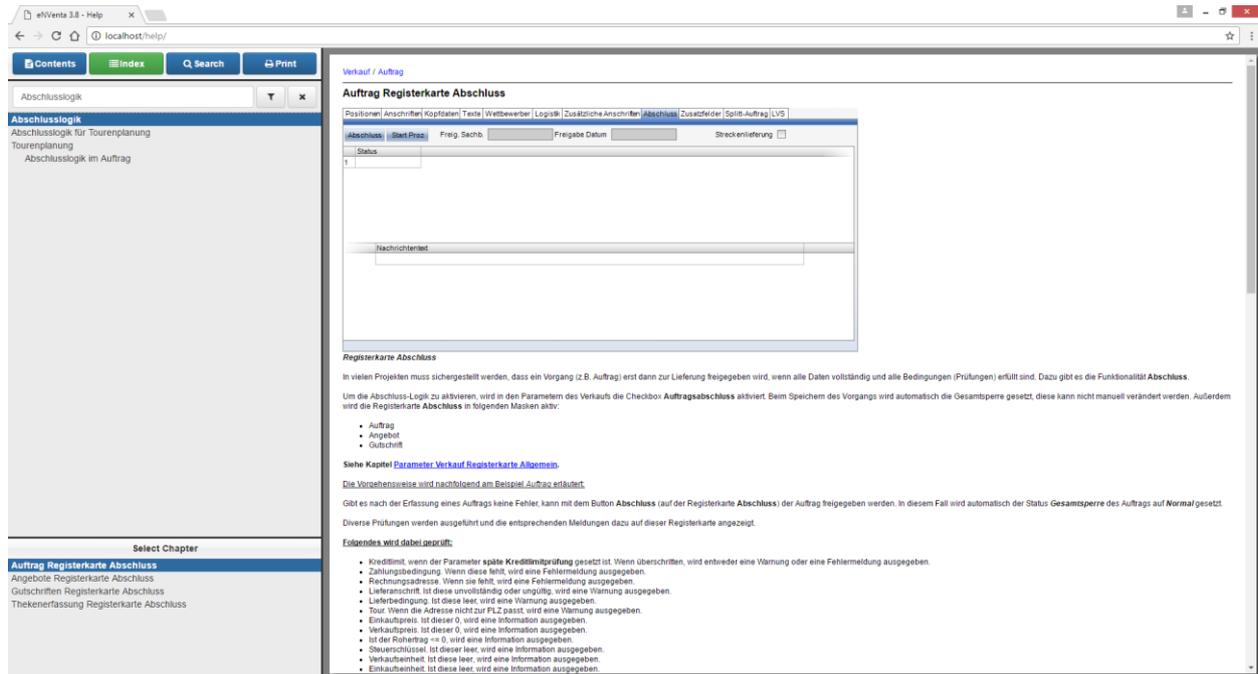
Die Webapplikation ist eine HTML 5 Anwendung, die ausschließlich auf den Technologien HTML, JavaScript und CSS basiert. Damit ist sie kompatibel zu nahezu allen Endgeräten mit einem Browser (→ **Webapplikation Voraussetzungen** beachten).

Inhalt



Nach dem Öffnen der Webapplikation im Browser ist standardmäßig der Bereich "Contents" selektiert. Hier kann im Baum auf der linken Seite das gewünschte Kapitel ausgewählt werden, welches dann rechts im Viewer angezeigt wird. Bei einem Klick auf einen Link wird der Baum automatisch aktualisiert und zum neuen Kapitel gesprungen.

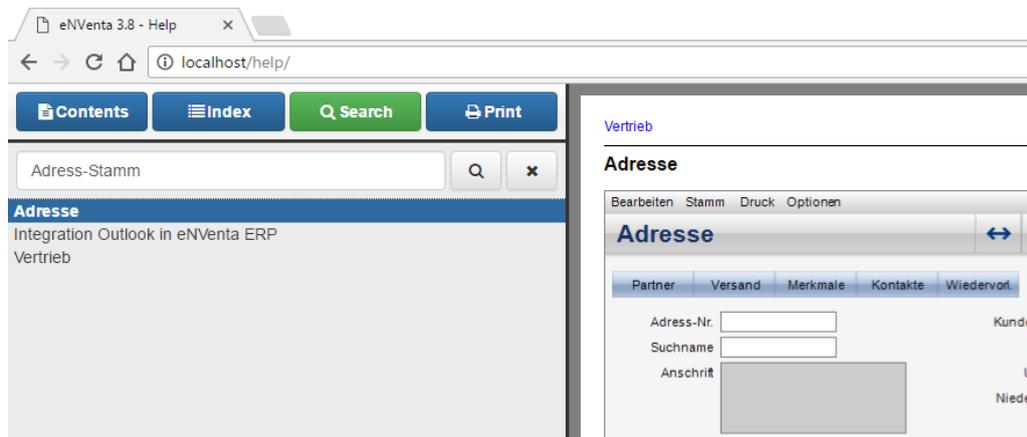
Index



Im Bereich "Index" kann der gesamte Index der Dokumentation durchsucht werden. Ist einem Index-Eintrag nur ein einziges Kapitel zugeordnet, wird dieses sofort nach dem Klick in den Baum ausgewählt.

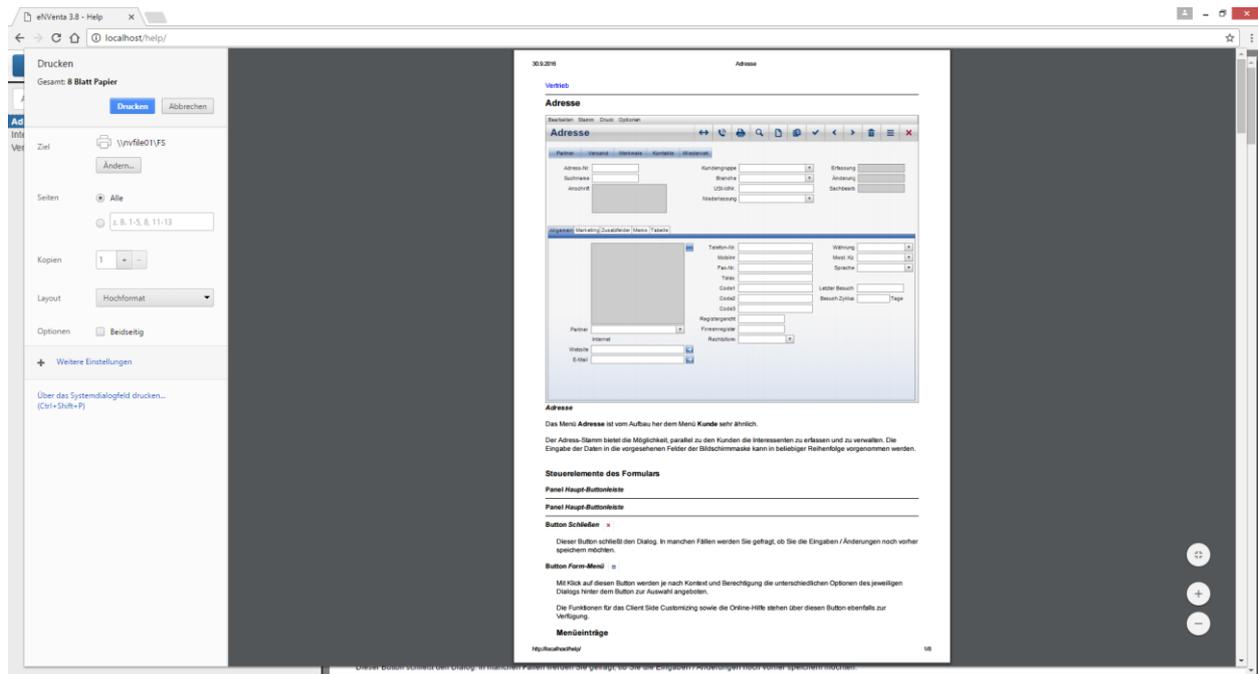
Sind einem Index-Eintrag mehrere Kapitel zugeordnet, kann das gewünschte Kapitel im Bereich "Select Chapter" links unten im Baum ausgewählt werden (siehe Abbildung).

Suche



Mit Hilfe der Suchfunktion können alle Kapitel nach Wörtern durchsucht werden. In der Ergebnisliste werden dann alle Kapitel angezeigt, die das eingetragene Wort enthalten.

Drucken



Um den Inhalt des gerade angezeigten Kapitels zu drucken, steht der Button „Print“ zur Verfügung. Anders als die Druckfunktion des Browsers, welche die gesamte Webapplikation inkl. Baum drucken würde, wird über den Button „Print“ nur der Inhalt des Viewers berücksichtigt.

Webapplikation Voraussetzungen

Um die von Framework Studio generierte Webapplikation zu hosten, wird einer der folgenden Webserver vorausgesetzt:

- IIS 7 oder höher
- Apache 2.2.x oder höher
- Nginx 1.9 oder höher

Es wird empfohlen, einen der folgenden Browser zur Anzeige der Webapplikation zu verwenden:

- Google Chrome
- Mozilla Firefox
- Microsoft Edge
- Microsoft Internet Explorer 11

Hinweis: Internet Explorer 10 und niedriger werden NICHT unterstützt.

Problem-Lösungen

Bei älteren Version des IIS kann es sein, dass einige für die Web-Application erforderlichen MIME-Types nicht gepflegt sind. Dies hat zur Folge, dass die Anzeige der Hilfe im Browser nicht funktioniert.

Folgende MIME-Types werden benötigt und müssen ggf. im IIS nachgetragen werden:

- .json application/json

- .otf font/otf
- .woff font/x-woff
- .woff2 application/font-woff2

Desweiteren werden die folgenden MIME-Typen. Diese sollten allerdings auch in älteren IIS-Versionen standardmäßig vorhanden sein:

- .css text/css
- .eot application/octet-stream
- .html text/html
- .ico image/x-icon
- .js application/x-javascript
- .png image/png#
- .svg image/svg+xml
- .ttf application/octet-stream

Hilfe-Url für eNventa ERP

Um beim Aufruf der Hilfe direkt ein bestimmtes Kapitel zu öffnen, muss dies in der URL in folgender Weise übergeben werden:

```
http://server/ eNventa_Help_de/?load=frmsalesOrder
```

In den Parametern von eNventa ERP muss die Hilfe-URL entsprechend auf die neue URL angepasst werden. Dabei werden die Platzhalter für Kapitel {0} und Sprache {1} verwendet:

```
http://server/eNventa_Help_{1}/?load={0}
```

Export mit FSConsole

Neben dem Export-Wizard kann die Webapplikation auch über die FSConsole.exe generiert werden. Hierzu gibt es die neuen Parameter \DOCUMENTATION, \ISO und \OUTPUT.

\DOCUMENTATION

Dieser Parameter ist nur ein Marker für die FSConsole, dass die Dokumentation gerendert werden soll.

\ISO

Gibt die Sprache an, in der die Dokumentation gerendert werden soll. Der Parameter hat auch Einfluss auf die automatisch generierten Screenshots.

\OUTPUT

Gibt das Verzeichnis an, in welches die Webapplikation gerendert werden soll.

Zusätzlich zu den neuen Parametern müssen die obligatorischen Connection- und Package-Version Parameter angegeben werden (\PACKAGE, \VERSION, \SERVER, \DBUSER usw.)

Beispiel:

```
FSConsole.exe \DOCUMENTATION \ISO de \OUTPUT „C:\Temp\Dokumentation“ \PACKAGE MyPackage  
\VERSION 3.8 \CONNECTIONTYPE sqlServer \SERVER Server01 \DATABASE MyRepository \DBUSER  
demo \DBPASSWORD demo123
```

Export mit PBE

Die Parallel Batch Engine wurde um die Aktion „ExportDoc“ erweitert. So kann der Export der HTML-Dokumentation einfach in einen Nachlauf integriert werden.

Weitere Details dazu finden Sie in der PBE-Dokumentation.

Korrigierte Fehler

ID	Text
2016104647	Java-Client: Tooltips für Registerkarten werden nicht angezeigt.
2016104786	Java-Client: Die vom JavaClient an den Broker gesendete Sprache wird vom diesem nicht korrekt verarbeitet und fällt immer auf den Wert "de-DE" zurück.
2016104691	Java-Client: Wird in einem OnLeave-Event das Form geschlossen, führt dies zu einer null-Exception im Java-Client.
2016104772	Java-Client: Wird aus einem TreeView ein Node gelöscht, kann es ggf. zu einer Null-Exception kommen.
2016104872	Java-Client: Von einem Mac aus lässt sich der Client nicht gegen einem Development Broker starten.
2016104823	<p>Package-Manager: Wird eine Package-Version inkl. Basis-Packages exportiert, dann werden von diesen Basis-Packages nicht alle Context-Versionen exportiert, wenn sie eine andere Hierarchie als das zu expordierende Haupt-Package haben.</p> <p>HINWEIS: Wenn ein Export mit so einer Package-Konstellation erstellt wird, dann kann dieses nicht mit einem älteren Service-Release von Framework Studio importiert werden.</p>
2016104905	Broker: Die an der Application eingestellte Server Culture zieht nicht - weder im Development-Broker noch nach einem Publish / Publish 2 Azure.
2016104860	Broker: In seltenen Fällen kann es vorkommen, dass sich im Broker die Prüfung der Session-Timeouts aufhängt. In der Folge werden dann die Sessions dieses Brokers von den anderen Brokern bereinigt,. Die führt dann zu "Session was lost" Meldungen.
2016104776	<p>Broker: Einige mehrfach ausgeführte Actions werden vom Broker nicht zusammengefasst. Dadurch können, wenn im Extremfall Actions sehr oft aufgerufen werden, sehr große XML-Mengen entstehen, welche zu Speicher-Engpässen im Java-Client führen.</p> <p><i>Neben den Focus-, Enabled- und Visible-Actions werden jetzt auch die folgenden Actions zusammengefasst: SetCellEditable, SetCellForeColor, SetCellBackColor, SetCellFont, SetColumnCaption, SetColumnVisible, SetForeColor, SetBackColor, SetToolTip, SetCaption, SetEditable.</i></p>
2016104645	Auth-Service: Bei der Verwendung eines Oracle-Repository muss auf die Groß-Klein-Schreibung des Benutzer-Namen geachtet werden. In eNventa ERP wirkt sich auf den Login-Dialog aus und führt dazu dass die Benutzerrechte-Übersicht nicht funktioniert.
2016104886	Azure: Bei einem Azure Role-Update wird der AuthService beendet und nicht mehr neu gestartet.
2016094527	IDE: Im Service-Release wird bei den Control-Properties "IsVisiableAU" und "IsEditableAU" das Editieren angeboten.Diese Properties dürfen aber nicht bearbeitet werden.
2016094448	IDE: Die Textbox "Binding Name" am Service Host Endpoint wurde entfernt, da sie nur redundant die Auswahl der Combobox "Binding Configuration" angezeigt hat.
2016104618	IDE: Wenn ich ein Objekt aus dem System-Package (z.B. cGlobal) eingecheckt wird, dann wird in der Source Control Hisotry kein Elementnamen angezeigt.
2016104816	IDE: Exception beim Hinzufügen einer dynamischen Combobox-Gridspalte, wenn der ListDataSource eine TextCollection ist.
2016104810	IDE: Wird einer Component über den Wizard eine neue Virtual Column

und/oder ein neues Property hinzugefügt, dann werden die Interfaces nicht zum automatischen Compile hinzugefügt.

Framework Studio 3.11.10 (30.9.2016)

Disablen von Panels

Jetzt können auch Panels und StackPanels per „IsEditable“-Property im Form-Designer oder per „SetEnabled“-Action im Code disabled werden. Dadurch werden, wie schon bei den TabPages, alle enthaltenen Controls disabled. Die enthaltenen Controls behalten aber ihre ursprüngliche Information, die zieht, wenn das Panel wieder enabled wird.

Im JavaClient haben Panels und StackPanels das IsEditable-Property bisher nicht ausgewertet. Dadurch kann es, wie schon im März bei den TabbedWindows, vorkommen, dass ein Panel, welches im Form-Designer aus Versehen auf IsEditable=false gestellt wurde, jetzt tatsächlich disabled ist.

Aus diesem Grund gibt es auch diesmal wieder ein SQL-Script, das im Repository Forms mit solchen Controls findet.

SQL-Server - Script:

```
select lbl_Version as [FS-Version], P_Name as [Package], lbl_Name as [Version],
COALESCE(F_Name, (select MAX(F_Name) from tblRep_Form F2 where F2.F_ID = F1.F_ID)) AS
[Form], F_ID from tblRep_Control
inner join subRep_Control on subCtrl_ID = Ctrl_ID and subCtrl_Version = Ctrl_Version and
subCtrl_PackageID = Ctrl_PackageID
inner join tblRep_Form F1 on subCtrl_F_ID = F_ID and subCtrl_F_Version = F_Version and
subCtrl_PackageID = F_PackageID
inner join weRep_Form on OID = F_ID and OVersion = F_Version and PackageID = F_PackageID
inner join (select P_Name, lbl_Name, lbl_Version, (select top 1 WS_WorkspaceID from
tblRep_Workspace where WS_LogonLbl_ID = Lbl_ID and WS_UserID = '101' and WS_CompleteDate
is not null order by WS_CompleteDate desc) as WS_WorkspaceID from tblRep_Label
inner join tblRep_Package on P_ID = Lbl_P_ID and lbl_Version in ('3.8.0', '3.9.0',
'3.10.0', '3.11.0')) as ws on ws.WS_WorkspaceID = WorkspaceID
where (ctrl_Dataxml like '<Ctrl_Dataxml><Type v="DevControlDataPanel"% ' OR ctrl_Dataxml
like '<Ctrl_Dataxml><Type v="DevControlDataStackPanel"% ') and ctrl_Dataxml like
'%<Editable v="false" />% ' and COALESCE(F_IsVersionDeleted, '0') = '0'
order by lbl_Version, P_Name, lbl_Name, F_Name;
```

Oracle - Script:

```
select lbl_Version AS "FS-Version", P_Name AS "Package", lbl_Name AS "Version",
COALESCE(F_Name, (select MAX(F_Name) from tblRep_Form F2 where F2.F_ID = F1.F_ID)) AS
"Form", F_ID from tblRep_Control
inner join subRep_Control on subCtrl_ID = Ctrl_ID and subCtrl_Version = Ctrl_Version and
subCtrl_PackageID = Ctrl_PackageID
inner join tblRep_Form F1 on subCtrl_F_ID = F_ID and subCtrl_F_Version = F_Version and
subCtrl_PackageID = F_PackageID
inner join weRep_Form on OID = F_ID and OVersion = F_Version and PackageID = F_PackageID
inner join (select P_Name, lbl_Name, lbl_Version, (select MAX(WS_WorkspaceID) over (order
by WS_CompleteDate) from tblRep_Workspace where WS_LogonLbl_ID = Lbl_ID and WS_UserID =
'101' and WS_CompleteDate is not null and rownum = 1) as WS_WorkspaceID from
tblRep_Label
inner join tblRep_Package on P_ID = Lbl_P_ID and lbl_Version in ('3.8.0', '3.9.0',
'3.10.0', '3.11.0')) ws on ws.WS_WorkspaceID = WorkspaceID
where (ctrl_Dataxml like '<Ctrl_Dataxml><Type v="DevControlDataPanel"% ' OR ctrl_Dataxml
like '<Ctrl_Dataxml><Type v="DevControlDataStackPanel"% ') and ctrl_Dataxml like
'%<Editable v="false" />% ' AND COALESCE(F_IsVersionDeleted, '0') = '0'
order by lbl_Version, P_Name, lbl_Name, F_Name;
```

Je nach Größe des Repositories kann es sein, dass dieses Script mehrere Minuten benötigt.

Korrigierte Fehler

ID	Text
2016094324	Java-Client: Memory-Leak beseitigt.
2016094369	Java-Client: TextFelder und unter-Controls von Containern (mit Ausnahmen derTabPage) können wenn ihr Containers deaktiviert ist, aktiv geschaltet werden.
2016094342	Broker: Werden mit Drop oder Remove Elemente aus einer Collection entfernt, welche als Unter-Unterobjekt an eine Combobox als Listdatasource gebunden ist, so kommt es unter Umständen zu einer NullReferenceException.
2016094492	Oracle: beim automatischen Datenbank-Update wird nicht überprüft, für die Spalte FSROWID ein Uniqie-Index existiert. Ein Fehlen dieses Index kann insbes. bei großen Tabellen zu erheblichen Performance-Einbußen führen.
2016094461	IDE: Bei Namens-Textboxen kann es sein, dass sie einen Fehler „Invalid Identifier“ anzeigen wenn kein Element ausgewählt und die Textbox deswegen leer ist.
2016094431	Publish: Bei der FSRowID-Konvertierung führt die Analyse einer SQL-Server 2010-Datenbank zu einem Fehler, wenn in der Datenbank Volltext-Indexe definiert sind.

Framework Studio 3.11.9 (8.9.2016)

Korrigierte Fehler

ID	Text
2016094286	<p>IDE: Werden in einem Grid in unterschiedlichen voneinander unabhängigen Packages dieselben Properties als Grid-Columns angelegt, dann führt dies zu einem Compile-Error (FSErr[1076]) wenn diese beiden Packages in einer Context-Version zusammengeführt werden.</p> <p><i>Ursache ist die Korrektur von Bug 2015120583 am 29.01.2016. Jetzt wird nur eine Warning (FSWarn[2032]) ausgegeben werden, wenn die Grid-Columns durch den Package-Code einen unterschiedlichen Namen haben.</i></p>
2016094292	IDE: An einem GlobalObject mit Versions-Nummer ≥ 2 kann der InitCode nicht geändert werden. Es wird eine Exception angezeigt.
2016084053	IDE: Wird im Documentation-Editor die Sprache gewechselt, wird ein langer Text an das Ende gescrollt.
2016084054	IDE: Wird in eine Documentation (z.B. für ein Property) per Strg+V derselbe Text eingefügt, wie an der Basis-Dokumentation (z.B. DBColumn oder Metadatatype), dann wird dieser Text nicht übernommen.
2016062984	IDE: Im Service Proxy können Service Operations nicht über die Spaltenüberschriften sortiert werden.

Framework Studio 3.11.8 (26.08.2016)

Neuer Print-Service

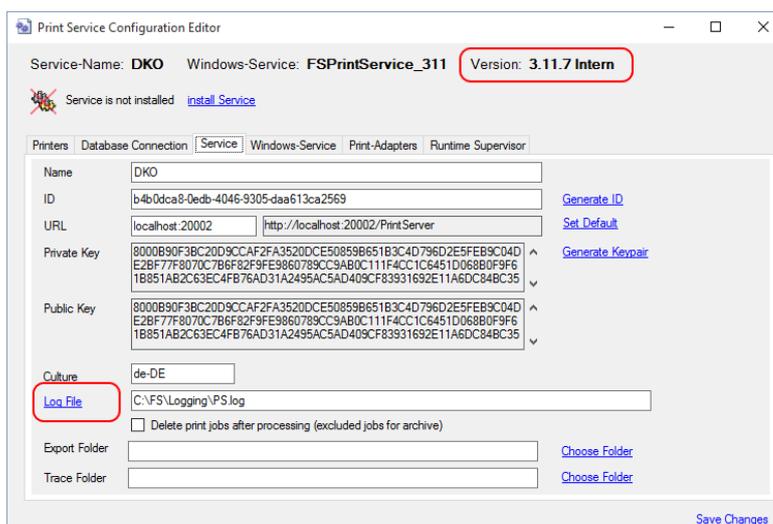
Der PDF-Print-Adapter wurde überarbeitet. Im PrintServiceConfigEditor kann man jetzt bei einem PDF-Adapter spezielle Klassen für AcrobatReader, FoxitReader oder SumatraPDF auswählen. Genaue Details zu diesen Adaptern können der Print-Service-Dokumentation entnommen werden.

Die Integration in den Runtime Supervisor wurde verbessert. Beim Start auftretende Fehler werden jetzt an den Runtime Supervisor gemeldet – z.B. Probleme beim Repository-Connect, Fehler bei den Druckern, Fehler in den Print-Adapter-Settings usw..

Level	Timestamp	Message
✖	24.08.2016 14:04:24	Invalid Printer: ID = e9c1a759-153b-4c92-9301-b1d157847c7e Nar Invalid Printer: ID = e9c1a759-153b-4c92-9301-b1d157847c7e Name = LVS Path = \\nvfile01\LVS
✖	24.08.2016 14:04:24	PDF PrintAdapter: invalid reader path specified in PrintService.exe PDF PrintAdapter: invalid reader path specified in PrintService.exe.config. path:"C:\Program Files (x86)\Adobe\Reader 11.0\Reader\AcroRd32.exe"
✖	24.08.2016 14:04:23	Invalid Printer: ID = dee162ff-08f4-45ca-89ca-dd214447b7fe Name
ℹ	24.08.2016 14:04:23	Connection established! Connection ID: fca68278-93fa-4943-a3b7-

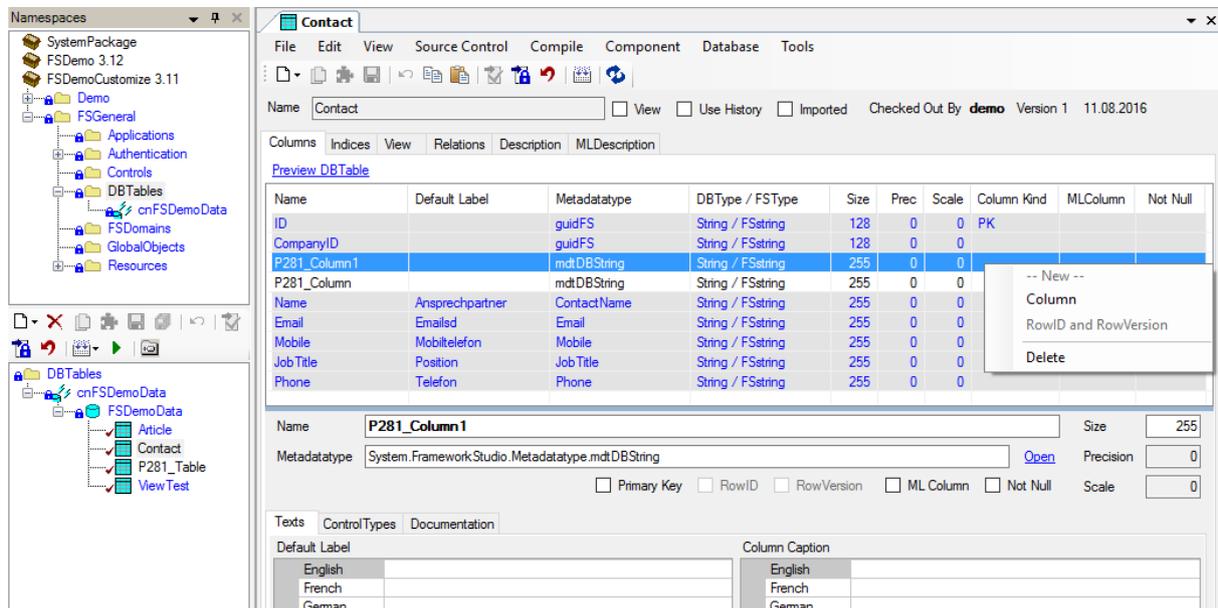
Im PrintServiceConfigEditor gab es kleinere Optimierungen:

- 1.) Es wird jetzt die Version des Print-Service angezeigt.
- 2.) Die Log-Datei kann über den Link geöffnet werden.



Usability bei DBColumns

Es gab einige Verbesserungen in der Usability von Framework Studio 3.11.



Bei den DBTables wurde die Bearbeitung der Column komplett überarbeitet. Wie an vielen anderen Stellen auch, werden jetzt eine Liste und ein getrennter Detail-Bereich eingesetzt. Dies bringt einige Vorteile.

- 1.) In der Liste werden Columns, die aus der Basis kommen, mit einem **grauen Hintergrund** angezeigt. Im Customizing geänderte Columns erhalten zusätzlich einen **blauen Text**. Diese farbliche Kennzeichnung betrifft auch alle anderen derartigen ListViews in Framework Studio – z.B. DBTable-Indizes, Component-Properties, Form-Properties.
- 2.) Die Liste stellt jetzt zusätzlich das „Default Label“ dar. Dies verbessert die Orientierung bei großen Tabellen. In der Spalte „Column Kind“ werden die 3 Schalter „Primary Key“, „RowID“ und „RowVersion“ zusammengefasst.
- 3.) Über das Context-Menü kann eine neue Column angelegt werden. Für neue Tabellen gibt es eine Abkürzung, um direkt RowID und RowVersion anzulegen.
- 4.) Die New-Befehle werden direkt auf der ersten Ebene des Context-Menüs angezeigt. Auch dies betrifft alle ListViews in Framework Studio – z.B. Properties, Methods.
- 5.) In den Details wird der Name der Column direkt bei der Eingabe validiert. Im Falle eines Fehlers wird dies – wie bei vielen anderen Namen auch – sofort kenntlich gemacht.
- 6.) Der Metadatatype der DBColumn kann über die bereits aus den Properties bekannte Vorschlag-Textbox angegeben werden. Der lästige TreeView entfällt damit.
- 7.) In den Details ist jetzt besser erkenntlich, welche Daten bearbeitet werden können oder welche Daten **geändert** wurden.
- 8.) Wie in allen anderen Editoren können die Position und Ausrichtung des Splitters und die Breite der Spalten angepasst und bei Bedarf über das Menü „File“ / „Save Layout“ gespeichert werden.

Außerdem wurde im Namespace-Baum die farbliche Kennzeichnung etwas angepasst. Das **leuchtende Blau** kennzeichnet jetzt die Elemente, die aus einem Basis-Package kommen

und customized wurden. Das **dunkle Blau** kennzeichnet Elemente, die von einer Basis erben – z.B. Forms oder Metadatentypen. Bisher wurden die Farben anders herum eingesetzt. Weil es aber wichtiger ist, zu erkennen, welches Element gecustomized ist, wird dafür jetzt das besser sichtbare leuchtende Blau verwendet.

Publish2Go über FSConsole

Die FSConsole.exe hat eine weitere Option „\Publish2Go“ erhalten. Mit dieser können für vorbereitete Settings Publish2Go-Pakete erzeugt werden.

Im verwendeten Publish2Go-Setting muss ein gültiger Folder angegeben sein. Dieser darf für den Export keinen Inhalt haben, ansonsten führt dies zu einer Exception.

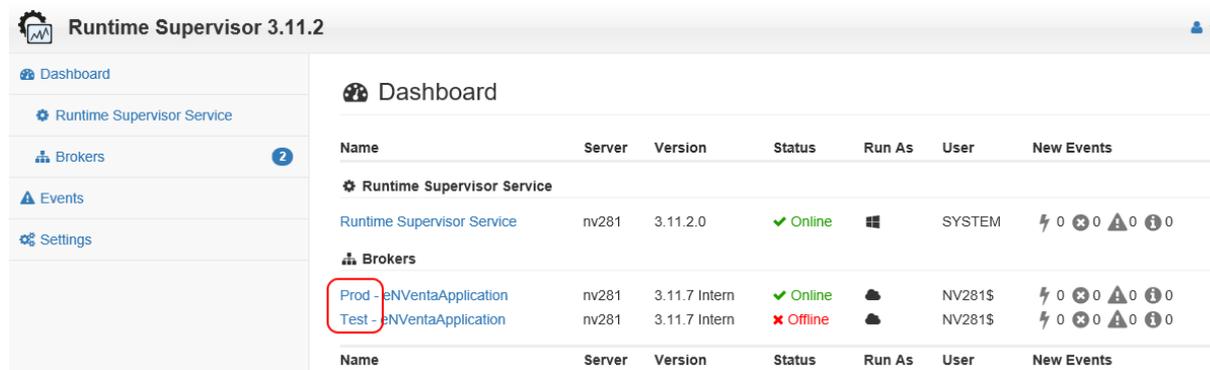
In der **PBE.exe** wurde diese Aktion ebenfalls entsprechend ergänzt. Die aktuelle Version dieses Programms finden Sie im Framework Studio Programmverzeichnis im Unterordner „PBE“. Kopieren sie einfach alle darin enthaltenen Dateien.

Beispiel:

```
FSConsole.exe \ConnectionType SqlServer \Server DBServer \Database eNVenta \DBUser sa \DBPassword sa \Package eNVenta \Version 3.7 \Publish2Go \Setting MeinSetting
```

Installation-Name im Runtime Supervisor

Der Runtime-Supervisor zeigt jetzt bei Applications und Services den Installation Name an. So können unterschiedliche Publish-Settings besser auseinander gehalten werden.



Eine Aktualisierung des Runtime-Supervisors ist nicht erforderlich. Diese Information wird nach dem Publish mit der neuen FS-Version an den Runtime-Supervisor übermittelt.

Neue Action Picture.SetImage()

Für das Picture-Control gibt es eine neue Action SetImage(). Mit dieser kann zur Laufzeit das Image verändert werden. Im Gegensatz zur Angabe eines DataSources wird so erst einmal das im Designer hinterlegte Image angezeigt. Erst bei Bedarf kann dann das Image geändert werden.

SetImage(string imageUrl) – es kann die relative Url einer Ressource oder auch eine Internet-Adresse übergeben werden. Wird null oder ein Leerstring übergeben, dann wird auf das im Designer angegebene Image zurück gewechselt.

SetImage(byte[] imageBytes) – es kann ein Byte-Array übergeben werden. Damit können zur Laufzeit erzeugte Bilder verwendet werden. Wird null oder ein leerer Byte-Array übergeben, dann wird auf das im Designer angegebene Image zurück gewechselt.

Beispiel:

```
// erzeugt ein 1-Pixel großes gelbes Bild
// (es ist ggf. sinnvoll dieses Bild zu cachen)
System.Drawing.Bitmap bitmap = new System.Drawing.Bitmap(1,1);
System.Drawing.Graphics graphics = System.Drawing.Graphics.FromImage(bitmap);
graphics.FillRectangle(
    new System.Drawing.SolidBrush(System.Drawing.Color.Yellow), 0, 0, 1, 1);
System.IO.MemoryStream ms = new System.IO.MemoryStream();
bitmap.Save(ms, System.Drawing.Imaging.ImageFormat.Png);
byte[] imageBytes = ms.ToArray();

// Das Bild durch die gelbe "Farbe" ersetzen
this.pic.SetImage(imageBytes);
```

Korrigierte Fehler

ID	Text
2016073679	Package-Manager: Wunsch - wenn man im Package Manager zu einer Packageversion den Publish-Wizard öffnet, soll automatisch der Setting Manager geöffnet werden, da in jedem Fall zunächst ein geeignetes Setting geladen werden sollte.
2016083879	IDE: Wenn man in einem customizten Report Document Type einen neuen Report anlegt, dann erhält dieser standardmäßig den Namen "report<PackageCode>_". Praktischer wäre aber "<PackageCode>_report".
2015036102	IDE: Wenn man in der Component-Query über die Wizards neue Query-Columns und Properties anlegt, dann zeigen die Properties erst nach neuem Neustart der IDE die Texte aus dem Metadatentyp bzw. der DataColumn an.
2016083936	IDE: Wenn man aus der IDE einen Development-Service starten möchte und der im Setting angegebene Port ist bereits (z.B. von einem gepublizierten Service) in Verwendung, dann startet der Development-Service nicht. In der Folge kommt es zu einer Fehlermeldung "Cannot connect to Development Service ..."
2016084008	Add-Ins: In dem Tool "Edit eNventa Update Script" funktioniert die Delete-Funktion nicht richtig. Der entsprechende Script-Block wird aus der Liste entfernt, dies wird aber nicht in die WTS-Datei übernommen.

Framework Studio 3.11.7 (29.07.2016)

Korrigierte Fehler

ID	Text
2016073384	Java-Client: Wird eine Grid-Zelle verlassen und der Fokus mit der Maus in ein anderes Control gesetzt, dann wird im EditingStopped-Event der Grid-Spalte ggf. der falsche Activator übergeben.
2016073490	Java-Client: Wird im Grid bei einem Filter-Kriterium kein Text angegeben, dann führt dies zu Exceptions. <i>Kein Text bedeutet jetzt kein Filter. Ohne Text wird für die entsprechende Spalte der Filter zurückgesetzt.</i>
2016063211	Package-Manager: Beim Import einer Packageversion in ein Oracle Repository kann es zu folgender Exception kommen, wodurch der Import abgebrochen wird: <i>Exception during import Die angegebene Umwandlung ist ungültig. System.Exception: Exception during import --> System.InvalidCastException: Die angegebene Umwandlung ist ungültig. bei Oracle.DataAccess.Client.OracleDataReader.GetOracleDate(Int32 i) bei FrameworkSystems.FrameworkDataClient.OracleODPFactory .GetValueFromDatareader(IDataReader reader, Int32 ordinal) bei FrameworkSystems.FrameworkStudio.General.DevObjects.Version.Label .LabelRecord.FCLockLabel(FCLockLevel lockLevel, FCLockLevel& sperrLock)</i>
2016073449	Package-Manager: Wird ein Service-Release kompiliert oder importiert, dann kann es bei neuen Element-Versionen dazu führen, dass in den Context-Versionen der SavedCode des Workspaces die alten Element-Versionen behält. In der Folge werden ggf. die XSD-Dateien von Reports nicht mehr rausgeschrieben oder beim Compile kommt es zu Fehlermeldungen.
2016063123	Package-Manager: Wird ein Package zusammen mit Basis-Packages importiert und bei einem Basis-Package handelt es sich um ein Service-Release, dann erfolgt keine Prüfung, ob das Service-Release zum bereits importierten Compile-Stand des Packages passt.
2016073341	IDE: Bei Ressourcen mit einem leeren Inhalt kann es passieren, dass sie nicht rausgeschrieben werden. Ggf. tritt dieser Effekt auch erst ein, nachdem ein Package in ein anderes Repository importiert wurde.
2016073371	IDE: Bei DBTables ist die View-Definition von Basis-Packages und im Service-Release-Modus nicht zugänglich.
2016063002	IDE: Wird im Code-Editor ein Replace All ausgeführt und die Component kann dabei nicht ausgecheckt werden, dann wird die Fehlermeldung in einer Endlos-Schleife angezeigt. Framework Studio kann dann nur noch über den Task-Manager beendet werden. <i>Der Suchdialog wurde durch einen eigenen ersetzt</i>
2016073685	Oracle: Wenn im FSROWID-Converter ein Fehler auftritt, dann hängt sich die Routine ohne Fehlermeldung auf.

Framework Studio 3.11.6 (24.06.2016)

Korrigierte Fehler

ID	Text
2016062742	<p>Java-Client: Werden in einem Request gleichzeitig SetEnabled-/SetVisible- und SetFocus-Actions an den Client geschickt, dann kann es passieren, dass die SetFocus-Action nicht ausgeführt wird. Ein SetFocus kann nur dann ausgeführt werden, wenn ein Control visible und enabled ist. Das trifft auch dann zu, wenn in diesem Request die SetFocus-Action vor dem Enablen des Controls ausgeführt werden soll.</p> <p><i>Jetzt sorgt der Java-Client dafür, dass die SetEnabled- und SetVisible-Actions vor den SetFocus-Actions ausgeführt werden.</i></p>
2016031581	<p>Java-Client: Ein FocusGained-Event oder eine SetCell-Action kann in Java 8 dazu führen, dass eine Grid-Zelle in den Editmodus versetzt wird, obwohl sich das Grid nicht im vordersten modalen Dialog befindet. Dies hat zur Folge, dass Werte vom Broker nicht korrekt übernommen werden, was weiter zu Fehlern beim Speichern von Datensätzen führen kann.</p>
2016062697	<p>Java-Client: Wenn beim Öffnen von Forms gleichzeitig MessageBoxen angezeigt werden, können diese unter Umständen an nicht intuitiven Positionen auf dem Bildschirm erscheinen.</p>
2016052674	<p>Java-Client: Wird im Grid für eine Spalte ein Format Pattern angegeben, das für negative Zahlen eine separate Definition enthält (z.B. "#,##0.00; #,##0.00"), so werden negative Werte nicht angezeigt.</p>
2016052676	<p>Java-Client: Wird an einem Grid eine dynamische Spalte mit "visible = false" hinzugefügt, so ist die Spalte trotzdem sichtbar.</p>
2016063001	<p>Package-Manager: Wenn man eine Package-Version löscht, von der man der Owner ist, dann werden dadurch auch Element-Versionen gelöscht, die in keiner anderen Package-Version benutzt werden. Diese Element-Versionen werden dadurch wieder "frei". Sie können ggf. bei einem anschließenden CheckOut in einer anderen Version des Packages erneut verwendet werden. Wenn nun das gelöschte Package bereits mit diesen Element-Versionen in einem anderen Repository importiert wurde, dann kollidieren diese später mit den neuen Elementen.</p> <p><i>Die Ursache liegt darin, dass die Element-Versionen übergreifend für alle Versionen eines Packages erzeugt werden (MAX + 1).</i></p> <p><i>Jetzt wird beim Löschen einer Package-Version eine Routine aufgerufen, welche in den verbleibenden Versionen des Packages alle bestehenden Element-Versionen um 5000 erhöht. Der Package-Manager zeigt eine entsprechende Hinweis-Meldung an.</i></p> <p><i>So werden bei einem anschließenden CheckOut in jedem Fall neue Versions-Nummern erzeugt, die in noch keinem Ziel-Repository existieren können.</i></p>
2016062961	<p>Package-Manager: Beim automatischen Löschen der ungenutzten Context-Versionen kann es passieren, dass Workspaces nicht korrekt gelöscht werden und abhängige Workspaces mit einer ungültigen Hierarchie hängen bleiben.</p>
2016052485	<p>IDE: Einige Elemente zeigen MessageBoxen beim Speichern an. Validierungen sollten beim CheckIn vorgenommen werden.</p>
2016062740	<p>IDE: Wenn man in einer Form-Methode auf einer GenerateLoadCondition-Methode Goto Definition aufruft (Strg+Doppelclick bzw. F12), dann hängt sich</p>

	die IDE auf und es kann nichts mehr geklickt werden. Dieses Verhalten ist ein Folge der Intellisense-Erweiterungen zum letzten Patchday (Bug 2016052471).
2016063077	IDE: Wenn in einem Formular sehr viele Workflowevents existieren, ist es nicht möglich einem Control ein vorhandenes Workflowevent zuzuweisen, wenn dieses außerhalb der sichtbaren Liste liegt.
2016062774	IDE: Bei Overlapping Controls werden keine Treffer aus Basis-Packages gefunden. Dies wäre aber bei Context-Versionen sehr hilfreich. <i>In Overlapping Controls gibt es jetzt neben dem Button "Refresh" einen neuen Button "Refresh including base packages".</i>
2016063004	IDE: Bei einigen Comboboxen gibt es Aktualisierungs-Probleme, wenn z.B. im Hintergrund die Änderung eines Wertes verhindert wird. Die Combobox zeigt dann den vermeintlich neuen Wert an, statt auf den alten Wert zurück zu wechseln. z.B. der Property-Type von Component-Properties.
2016062820	Doku-System: In gecustomizten Dokumentationen kann nicht auf Image-Ressourcen aus den Basis-Packages zugegriffen werden.
2016062914	Doku-System: Bei gerenderten Properties fehlt im HTML das <meta charset="utf-8" /> Tag, was zu Anzeigefehlern führt.
2016062737	Oracle: Werden bei einer SQL-Query Zahlenwerte ausgelesen, kann dies in seltenen Fällen ggf. zu ArgumentOutOfRangeExceptions oder InvalidCastExceptions führen.
2016062823	Add-Ins: Das Package Merge Add-In bricht unter Oracle mit einer Exception ab, wenn im Repository versteckte Spalten existieren.
2016062734	Azure: seit dem letzten Patchday gibt es beim Start der Anwendung eine Exception, wenn für die Application die Authentication aktiviert ist.

Framework Studio 3.11.5 (27.05.2016)

Neuer Development Broker ohne IIS

Um den IIS auf Entwicklungsrechnern nicht mehr zu benötigen, wurde ein neuer Development Broker auf Basis von Owin und Web API 2 entwickelt.

ACHTUNG:

Alle hier beschriebenen Änderung gelten nur für den Development Broker und die Development Service Hosts! Wird eine Anwendung über den Publish Wizard oder Publish 2 Go bereitgestellt, wird weiterhin der IIS verwendet.

Der bisherige Development Broker bestand im wesentlichen aus zwei Teilen. Der erste Teil war das auf dem IIS gehostete virtuelle Verzeichnis (Development Broker Directory), welches beim erstmaligen Ausführen der Anwendung erstellt wurde. Der IIS nahm die Requests des Java Clients entgegen, hostete die Client Start Page und fungierte als Fileserver für Bilder und Reports.

Der zweite Teil des Development Brokers lief als FSDevBrokerHost.exe (Tray Icon) und arbeitete auf Basis des Compile Directories, um die Requests des IIS zu verarbeiten.

All diese Funktionalitäten wurden in Framework Studio 3.11.5 nun in einen einzigen Prozess (FSBrokerHost.exe) zusammengefasst. Der größte Vorteil, der sich daraus ergibt ist, dass der IIS auf einem Entwicklungsrechner nicht mehr benötigt wird.

Wird nun eine Anwendung aus Framework Studio heraus gestartet, wird nur noch die FSBrokerHost.exe (als Tray Icon wie bisher auch) für den Broker oder Service Host gestartet. Der Java Client kommuniziert direkt mit diesem Prozess.

Das Development Broker Verzeichnis für den IIS gibt es somit nicht mehr und der entsprechende Menüeintrag zum öffnen dieses Verzeichnisses aus dem Menü „Compile“ entfernt. Alle für den Development Broker nötigen Dateien befinden sich nun im Compile Directory.

Hinweis zum Debugging von Development Broker und Service Hosts

Der Development Broker und die Development Service Hosts wurden bisher über verschiedene Executables realisiert, die FSDevBrokerHost.exe für den Broker und die FSDevServiceHost.exe für die Service Hosts. Beide werden nun über die FSBrokerHost.exe verarbeitet.

Trace Information und Icons im Context-Menü

Die Tray Icons des Development Brokers/Hosts wurde um den Menüpunkt „Show Trace Log“ erweitert. Zusätzlich wurden die wichtigsten Optionen mit Icons versehen.

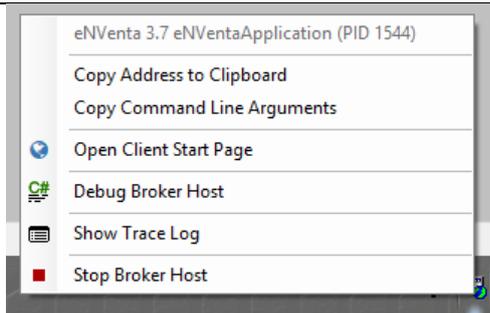


Abbildung 1: Development Broker

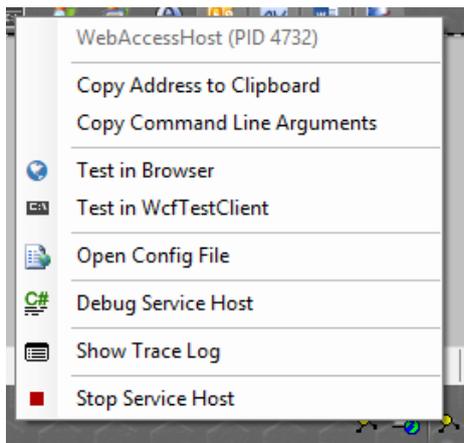


Abbildung 2: Development Service Host

„Show Trace Log“ öffnet ein Ausgabefenster, in das der Development Broker/Host Fehler protokolliert. Sollte es z.B. technische Schwierigkeiten beim Verarbeiten eines Requests geben, ist es ratsam, erst einmal einen Blick in das Trace Log zu werfen.

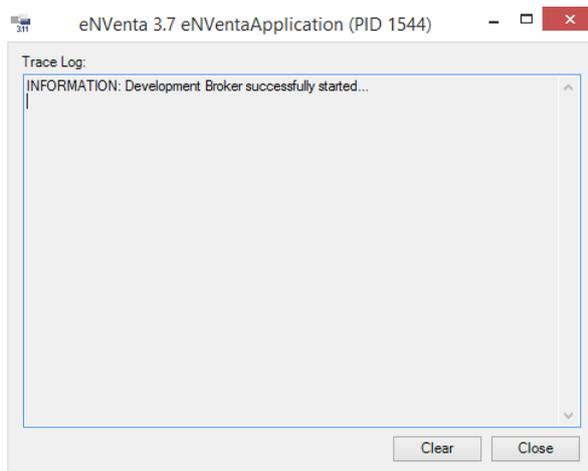


Abbildung 3: Trace Log

Client Start Page

Die Client Start Page im Development Broker wurde umgestaltet und optisch besser aufbereitet.

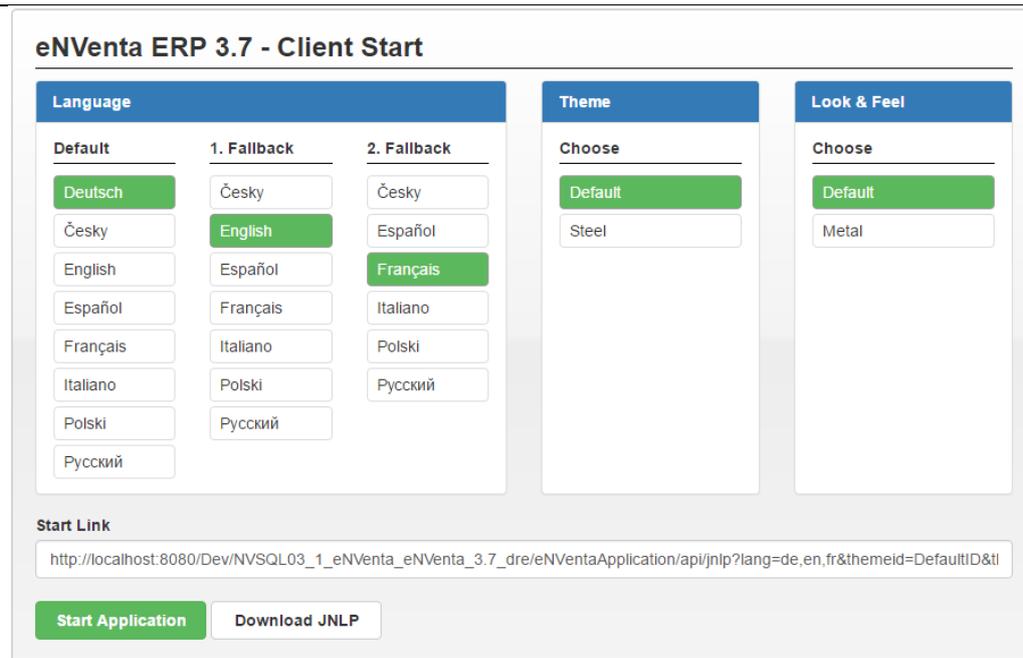
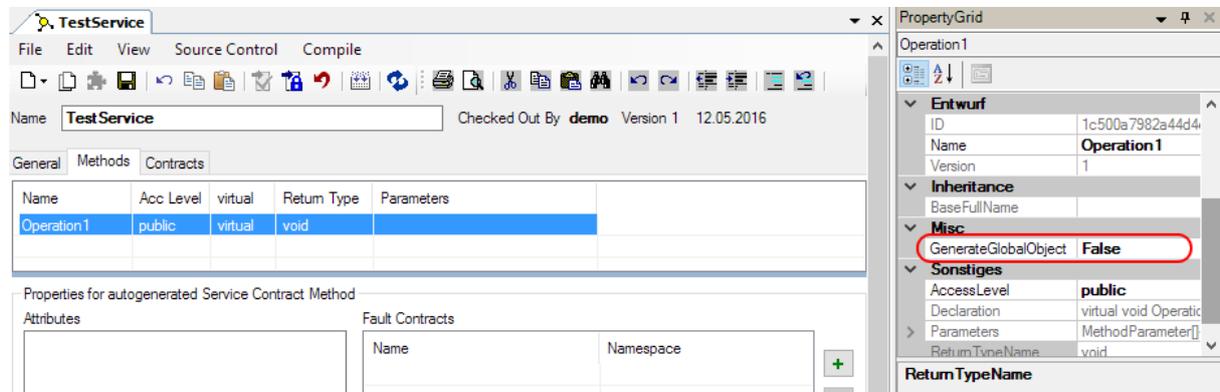


Abbildung 4: Neue Client Start Page

GlobalObject / Performance in Service-Methoden

Bei Service-Methoden kann man im Property-Grid die Eigenschaft „GenerateGlobalObject“ angeben. Wenn diese den Wert „True“ hat, dann wird im generierten Code für diese Methode ein GlobalObject erzeugt, welches über „Global“ angesprochen werden kann.



Bisher wurde diese Eigenschaft beim Anlegen einer neuen Methode mit „True“ vorbelegt.

Oftmals wird aber individuelle Logik implementiert, bei der ein selbst erzeugtes GlobalObject aus einem Cache oder einem Sitzungs-Kontext verwendet wird. Steht die Eigenschaft auf „True“ wird bei jedem Methoden-Aufruf zusätzlich ein GlobalObject erzeugt. Dies ist unnötig und verschlechtert zudem sehr stark die Performance.

Aus diesem Grund wird ab jetzt **bei neuen Methoden** die Eigenschaft mit „False“ vorbelegt. Wenn das generierte „Global“ benötigt wird, muss die Eigenschaft manuell auf „True“ gesetzt werden.

HINWEIS: Bestehende Methoden werden nicht geändert. Es ist sehr sinnvoll bei eigenen Service-Methoden bzw. im Customizing überschriebenen Methoden diese Einstellung zu prüfen und ggf. auf „False“ zu setzen. Dies kann die Performance dieser Services stark verbessern.

Korrigierte Fehler

ID	Text
2016052298	<p>Broker / Reporting: Beim Erzeugen von Druck-Belegen mit der Crystal Reports Runtime gibt es bei Exceptions einen Mechanismus, der die Erzeugung bis zu 10 mal wiederholt. Wenn die Erzeugung des Reports aber extrem lange dauert und wiederholt in eine Exception läuft, dann hat dies extrem lange Laufzeiten zur Folge.</p> <p><i>Es gibt jetzt zusätzlich einen Timeout von 1 Minute. Die bis zu 10 Wiederholungs-Versuche werden nur noch maximal innerhalb dieser Zeit gestartet.</i></p>
2016042083	<p>Broker: Eine Collection wird in einem TreeView angezeigt. Nun wird dieser Collection per Insert() ein neuer Record am Anfang oder in der Mitte der Collection hinzugefügt. Wenn man jetzt den Wert des Properties Ändert, welches als DisplayMember im TreeNode angezeigt wird, dann wird die Anzeige im Client nicht aktualisiert.</p>
2016052328	<p>Broker: Wird in der Form-Methode IsApplicationQuitAllowed() ein Workflow-Event ausgelöst, welches ein anderes Form öffnet, dann führt dies zu einer Exception: "Die Auflistung wurde geändert..."</p>
2016052476	<p>JavaClient: Die Werte für das OnCellValidate-Event von GridSpalten werden falsch kodiert übertragen. In Folge dessen werden die Properties an den EventArgs falsch konvertiert.</p>
2016010764	<p>JavaClient: Wenn beim Ausdocken eines Forms ein OnLeave-Event gefeuert wird, kann es zu folgender Exception kommen:</p> <p><i>java.lang.ArrayIndexOutOfBoundsException: No such child: 0</i></p>
2016052332	<p>IDE: Wenn man nach einem Get Latest CheckIns die IDE neu startet, schlägt der Login-Dialog "Get DLLs" vor, obwohl das Compile-Directory noch aktuell ist. In der Folge müssen alle Änderungen noch einmal neu kompiliert werden.</p>
2016052402	<p>IDE: In sehr seltenen Fällen kann es leider immer noch passieren, dass beim Complete eines FrameworkCompilers solche Workspaces gelöscht werden, die noch von anderen Package-Versionen benötigt werden.</p> <p><i>Es wurde noch eine weitere Absicherung eingebaut, die das Löschen von benötigten Workspaces unterbindet.</i></p>
2015036111	<p>IDE: Wenn man den Focus im Comment Window hat und Strg + S drückt, dann öffnet sich ein Speichern-Dialog um eine XML-Datei zu speichern. Es sollte aber das aktuelle Element gespeichert werden.</p>
2015109600	<p>IDE: Durch einen unnötigen Refresh Workspace kann man einen invaliden Workspace erzeugen. Das tritt dann auf, wenn vom Benutzer Elemente oder Namespaces nach dem letzten FrameworkCompiler eingecheckt wurden. Wird dann durch einen erneuten Refresh erneut der FrameworkCompiler-Workspace gezogen, sind diese Elemente nicht mehr sichtbar. In der Folge werden Änderungen vermisst oder es kommt sogar zu Exceptions.</p> <p><i>Im Login-Dialog ist die Checkbox "Get latest FrameworkCompiler workspace" jetzt deaktiviert, wenn der Benutzer-Workspace bereits auf den aktuellsten FrameworkCompiler-Workspace aufsetzt. Zudem sind die Workspace-Informationen im Login-Dialog jetzt klarer formuliert.</i></p>
2016042254	<p>IDE: Intellisense wird für einige Unter-Elemente nicht angeboten.</p>
2015036096	<p>IDE: Im Intellisense von Form-Methoden werden die TreeNodes nicht korrekt angeboten.</p>
2016052471	<p>IDE: Im Intellisense gibt es folgende Fehler:</p>

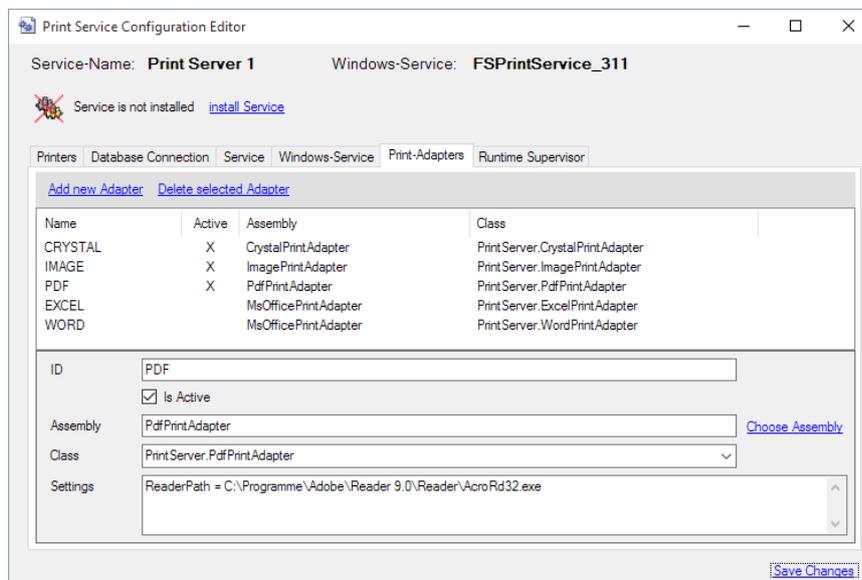
	<ol style="list-style-type: none">1.) DataContracts werden unter Umständen mit fast gleichnamigen Components verwechselt (dcPrice / cPrice)2.) Go to Definition funktioniert manchmal nicht bei tieferen Hierarchien - insbes. unterhalb von this.Global.ocGlobal.3.) Go to Definition geht nur bei Methoden aus demselben Element.4.) Bei einem Component-Property springt Go to Definition nicht zum Property, sondern zur Datentyp-Component des Properties.5.) Bei Variablen, die in der ersten Code-Zeile einer Methode deklariert werden, werden evtl. keine Unter-Objekte aufgelöst.
2016052259	Publish2Go: Beim Exportieren eines Service-Release-Paketes werden auch Ordner für die Publish2Go-Routine erzeugt. Dabei werden die Dlls statt in den Ordner "assembly" in den Ordner "bin" abgelegt. In der Folge werden beim Publish einer Application mit der Publish2Go.exe die Service-Release-Dlls nicht berücksichtigt.
2016042169	Publish2Go: Beim Starten der Anwendung wird das Startbild nicht angezeigt.
2016052455	FSConsole: Publish über FSConsole.exe prüft Authentication-Einstellungen an Applications und Service Hosts auf Duplikate.
2016052517	Azure: Dauert eine Anfrage an den Broker länger als 2 Minuten, dann hängt sich diese Anfrage und in der Folge der Java-Client auf. Der Client kann dann nur noch über den Task-Manager beendet werden.
2016052265	Add-Ins: Wunsch: Das Package Merge Add-In soll die Methoden OnBeforeSaveCustom und OnAfterSaveCustom (wie im Ziel vorhandene Methoden) umbenennen, inhaltlich anpassen und eine Warning erzeugen.
2016052425	FS-AddIns / UpdateScriptManager: Es muss abgesichert werden, dass am Ende eines Scriptes ein Semikolon steht. Ansonsten wird das Script von der eNventa Update Routine nicht korrekt verarbeitet. <i>Das Semikolon wird jetzt angehängt, wenn der Editor geschlossen und das Script in die Ressource zurück geschrieben wird.</i>

Framework Studio 3.11.4 (29.04.2016)

Print-Service

Es gibt einen neuen Print-Service.

Im Print Service Configuration Editor ist es jetzt möglich die Print-Adapter zu konfigurieren.



Weitere Details dazu finden finden sich im Dokument „FS 3.11 Installation PrintService.pdf“, welches im Installations-Paket des aktuellen Print-Service enthalten ist.

Außerdem können Print-Adapter jetzt deaktiviert werden. Bisher war es erforderlich, ungewünschte Adapter in der Config-Datei zu löschen.

Darüber hinaus wurden die folgenden Bugs behoben:

Bug –ID	Beschreibung
2016042009	Print-Service: Obwohl im PrintServiceConfigEditor die Einstellung "Delete print jobs after processing" nicht gestetzt ist, löscht der Print-Service die Jobs nach dem Druck. Dies tritt auf, wenn nach der Installation diese Einstellung nicht geändert wird.
2016042133	Print-Service: Der Service wirft eine SecurityException, wenn er ohne Administratorrechte bei aktiviertem UAC gestartet wird.

Korrigierte Fehler

ID	Text
2016042133	Auth-Service: Der Service wirft eine SecurityException, wenn er ohne Administratorrechte bei aktiviertem UAC gestartet wird.
2016031780	Broker: Wenn man an einem Grid dynamische Spalten hinzufügt und anschließend mit HidePermanent bestehende Spalten ausblendet, dann wird das XML für den Client nicht mehr korrekt aufgebaut. In der Folge kommt es zu Exceptions im Java-Client (z.B. Java-Exception: 38 >= 38).
2016041988	Broker: An FrameworkOn(Can)DropEventArgs ist die Eigenschaft TargetControl null, wenn das Ziel ein Form (genauer die Default Variant des Forms, "Form.Variant") ist.

2016042134	<p>Broker: Damit nicht mehr laufende oder nicht korrekt beendete Sitzungen nicht ewig hängen bleiben und dadurch Ressourcen und Lizenzen blockieren, sind Timeout-Mechanismen vorhanden. Ein Mechanismus erkennt einen abgestürzten Broker und wirft Lizenz-blockierende Sessions weg. Dies ist auch der Fall, wenn die Broker-Routine zum Aktualisieren des Timeouts nicht mehr funktioniert. In diesem Fall werden laufende Sitzungen dieses Brokers von einem anderen Broker gelöscht und die Benutzer erhalten die Meldung: "Authorisation error while checking current sessions: Session was lost"</p> <p><i>Neben einer zusätzlichen Absicherung wurde die Protokollierung der Timeout-Routine ausgeweitet. Auftretende Fehler werden jetzt im Runtime-Supervisor bzw. im Event-Log besser protokolliert.</i></p>
2016031613	<p>Doku-System: Das Überschreiben von Sections funktioniert nur mit der direkten Basis. Sind ein oder mehrere Packages zwischen der zu überschreibenden Section und dem aktuellen Package, so funktioniert der "Insert Section"-Dialog im Documentation-Editor nicht korrekt und fügt fälschlicherweise Sections mit neuen IDs in die Dokumentation ein.</p>
2016041848	<p>IDE: Es kann vorkommen, dass Framework Studio beim Öffnen des Kontextmenüs der Methoden- oder Propertiesübersicht abstürzt.</p> <p><i>Der Objektverweis wurde nicht auf eine Objektinstanz festgelegt. System.Exception: An unhandled exception occurred and the application is shutting down. bei FrameworkSystems.FrameworkStudio.General.DevObjects.Methods. MethodCollection.IsPastItemFromXmlEnabled(String clipBoardContent)</i></p>
2016041920	<p>IDE: Bei Tabellen-Namen wird die Länge nicht auf 30 Zeichen geprüft. Ein entsprechende Fehler tritt unter Oracle somit erst zur Laufzeit auf.</p>
2016042056	<p>IDE: Wenn man an einem Property die Checkbox "External Type" gesetzt hat, in der Textbox "Datatype" mit dem Tippen beginnt und anschließend einen Eintrag aus den Vorschlägen per Maus auswählt, beendet sich FS mit einer unbehandelten Ausnahme.</p>
2016042131	<p>IDE: Wird ein Index an einer DBTable auf "Inactive" geschaltet, wird dieser bei der Prüfung auf FSWarn[1056], FSErr[1056] und FSErr[1057] trotzdem beachtet.</p>
2016041977	<p>IDE: Die Intellisens bietet in den CodeEditoren am Form fälschlicherweise die Membervariable 'variant' an.</p>
2016031676	<p>Add-Ins: Das Package Merge Add-In führt am Metadatatype angegebene Wertebereiche nicht korrekt zusammen.</p>
2016042046	<p>Add-Ins: Das Package Merge Add-In zeigt beim Start den Hinweis, dass 'Correct workflow link parameters' ausgeführt werden muss, obwohl dies bereits geschehen ist.</p>

Framework Studio 3.11.3 (31.03.2016)

MLColumns benutzen FSROWID

An einer DBColumn kann schon länger die Eigenschaft **IsMLColumn** gesetzt werden. Dies aktiviert für diese Spalte und die darauf aufsetzenden Properties die Mehrsprachigkeit. Mit der Einführung der FSROWID in Framework Studio 3.11 wurde es nötig, die Speicherung der mehrsprachigen Texte zu überarbeiten.

Jetzt gibt es eine neue Tabelle dbRun_MLStrings. Diese benutzt direkt den Wert der FSROWID um einen Datensatz zu identifizieren. Der neue Mechanismus bringt zusätzlich einige Vorteile mit sich:

- 1.) Die FSROWID ist dauerhaft stabil und so können die Datensätze dauerhaft referenziert und ggf. auch in anderen Datenbanken transportiert werden.
- 2.) Die direkte Referenz auf die FSROWID ermöglicht es, die MLString-Daten per SQL zu referenzieren. Dies ebnet auch den Weg für zukünftige Weiterentwicklungen.
- 3.) Das Datenvolumen wird deutlich reduziert, weil die FSROWID deutlich kompakter ist als der alte Hash-Code.



ACHTUNG: Die Umstellung führt zu Verhaltensänderungen. Wenn Sie MLColumns einsetzen, dann müssen Sie die folgenden Punkte unbedingt beachten:

- 1.) Der neue Mechanismus funktioniert **ohne weitere Anpassungen**. In der eigenen Programmierung muss daher nichts geändert werden.
- 2.) **Bestehende Daten müssen** in die neue Tabelle **überführt werden**. Dazu gibt es, wie für die FSROWID, eine Routine, welche weiter unten beschrieben wird.
- 3.) Weil der neue **MLColumn-Mechanismus auf die FSROWID aufsetzt**, kann er nur noch in DBTables benutzt werden, die die Spalte FSROWID (mit der Eigenschaft IsROWID) besitzen. Beim Compile der DBTables werden entsprechende Warnungen ausgegeben.

In Framework Studio 3.11 gibt es noch einen Fallback für einen PK ohne ROWID. Diese Daten werden weiter in der alten Tabelle gespeichert. Ab Framework Studio 4.0 wird es diesen Fallback nicht mehr geben. Diese Tabellen werden dann einen Compile-Error produzieren.

Bisher wurden die Texte in einer Tabelle ml_Strings gespeichert. Zur Identifikation eines Datensatzes wurde ein Hash-Code berechnet, welcher auf der ROWID und wenn diese nicht vorhanden war auf den PK-Eigenschaften aufsetzt. Ab FS 3.11 arbeiten die Components nicht mehr mit der ROWID sondern mit der FSROWID. Dadurch ergeben sich andere Hash-Codes und die bestehenden Daten können nicht mehr verwendet werden. Zudem bestand bei Oracle-Datenbanken das Problem, dass beim Import eines Dump-Files ggf. neue ROWIDs erzeugt und somit die MLString-Daten komplett unbrauchbar werden konnten. Die Datensätze werden nun in einer automatisch neu angelegten Tabelle mit dem Namen dbRun_MLStrings abgelegt.

Bestehende Daten überführen

Wenn der Mechanismus, fremdsprachige Texte über die **IsMLColumn**-Eigenschaft zu verwalten, genutzt wird, müssen die Mehrsprachigen Texte in der Daten-Datenbanken mithilfe einer Routine in die neue Tabelle überführt werden. Diese steht in dem Dialog zur Verfügung, der im Publish-Wizard unter dem Button „FSROWID-Update“ geöffnet werden kann. Im Dialog gibt es dann ein Button „ML-Strings übertragen“, der den eigentlichen Vorgang startet.

Der Vorgang überträgt alle mehrsprachigen Texte in die neue Tabelle. In der neuen Tabelle vorhandene Datensätze werden dabei überschrieben; die Einträge in der alten Tabelle bleiben erhalten. Grundsätzlich kann die Routine mehrfach ausgeführt werden. Die einzelnen Schritte der Routine werden in Protokollform ausgegeben. Sollte ein Problem auftreten, wird dieses ebenfalls protokolliert.

ACHTUNG! Registerkarten komplett disabled

Im letzten Service-Release gab es eine Korrektur im Java-Client:

2016010769	Java-Client: Durch das enablen der TabPage oder des TabbedWindows geht die IsEditable-Eigenschaft der Unter-Controls verloren.
------------	--

Diese Korrektur hat auch zur Folge, dass jetzt am TabbedWindow-Control die Eigenschaft „IsEditable“ korrekt ausgewertet wird.

Wenn man in der Vergangenheit an einem TabbedWindow im Property-Grid die Eigenschaft „IsEditable“ (ggf. aus Versehen) auf „false“ gesetzt hat, dann ist dies bisher nicht aufgefallen. Jetzt hat diese Eigenschaft aber zur Folge, dass das komplette TabbedWindow inaktiv wird und nicht mehr bedient werden kann. Das ist insbesondere daran zu erkennen, dass im Java-Client die Beschriftungen der Registerkarten kursiv dargestellt werden.

Diese Änderung betrifft die aktuellen Service-Releases von Framework Studio 3.8 bis 3.11.

Eigene Packages sollten auf diese Problematik hin überprüft und korrigiert werden. Um die Recherche zu vereinfachen, haben wir ein SQL-Script erstellt, welches die betroffenen Forms identifiziert.

Das Script muss auf dem Repository ausgeführt werden. Es liefert eine Liste mit den folgenden Informationen: FS-Version, Name des Packages, Name der Package-Version, Name des Forms und die interne ID des Forms. Analysiert werden jeweils die aktuellen FrameworkCompiler-Stände. Nach einer Korrektur muss erst der FrameworkCompiler ausgeführt werden, damit das Script diese nicht mehr findet.

In dem entsprechenden Form muss am TabbedWindow im Property-Grid die Eigenschaft „IsEditable“ zurückgesetzt werden, sodass dort der Wert „true“ steht.

SQL-Server - Script:

```
select lbl_Version as [FS-Version], P_Name as [Package], lbl_Name as [Version],
COALESCE(F_Name, (select MAX(F_Name) from tblRep_Form F2 where F2.F_ID = F1.F_ID)) AS
[Form], F_ID from tblRep_Control
inner join subRep_Control on subCtrl_ID = Ctrl_ID and subCtrl_Version = Ctrl_Version and
subCtrl_PackageID = Ctrl_PackageID
inner join tblRep_Form F1 on subCtrl_F_ID = F_ID and subCtrl_F_Version = F_Version and
subCtrl_PackageID = F_PackageID
inner join weRep_Form on OID = F_ID and OVersion = F_Version and PackageID = F_PackageID
inner join (select P_Name, lbl_Name, lbl_Version, (select top 1 WS_workspaceID from
tblRep_Workspace where WS_LogonLbl_ID = Lbl_ID and WS_UserID = '101' and WS_CompleteDate
is not null order by WS_CompleteDate desc) as WS_workspaceID from tblRep_Label
inner join tblRep_Package on P_ID = Lbl_P_ID and lbl_Version in ('3.8.0', '3.9.0',
'3.10.0', '3.11.0')) as ws on ws.WS_workspaceID = workspaceID
where ctrl_dataxml like '<Ctrl_Dataxml><Type v="DevControlDataTabbedwindow"%>' and
ctrl_dataxml like '%<Editable v="false" />%>' and COALESCE(F_IsVersionDeleted, '0') = '0'
order by lbl_Version, P_Name, lbl_Name, F_Name;
```

Oracle - Script:

```
select lbl_Version AS "FS-Version", P_Name AS "Package", lbl_Name AS "Version",
COALESCE(F_Name, (select MAX(F_Name) from tblRep_Form F2 where F2.F_ID = F1.F_ID)) AS
"Form", F_ID from tblRep_Control
inner join subRep_Control on subCtrl_ID = Ctrl_ID and subCtrl_Version = Ctrl_Version and
subCtrl_PackageID = Ctrl_PackageID
inner join tblRep_Form F1 on subCtrl_F_ID = F_ID and subCtrl_F_Version = F_Version and
subCtrl_PackageID = F_PackageID
inner join weRep_Form on OID = F_ID and OVersion = F_Version and PackageID = F_PackageID
inner join (select P_Name, lbl_Name, lbl_Version, (select MAX(WS_workspaceID) over (order
by WS_CompleteDate) from tblRep_workspace where WS_LogonLbl_ID = Lbl_ID and WS_UserID =
'101' and WS_CompleteDate is not null and rownum = 1) as WS_workspaceID from
tblRep_Label
inner join tblRep_Package on P_ID = Lbl_P_ID and lbl_Version in ('3.8.0', '3.9.0',
'3.10.0', '3.11.0')) ws on ws.WS_workspaceID = WorkspaceID
where ctrl_DataXml like '<Ctrl_DataXml><Type v="DevControlDataTabbedWindow%"' and
ctrl_DataXml like '%<Editable v="false" />%' AND COALESCE(F_IsVersionDeleted, '0') = '0'
order by lbl_Version, P_Name, lbl_Name, F_Name;
```

Je nach Größe des Repositories kann es sein, dass dieses Script mehrere Minuten benötigt.

Print-Service

Es gibt einen neuen Print-Service.

Bug -ID	Beschreibung
2015088681	<p>Print-Service: Wenn im PrintserviceConfigEditor ein Export Folder angegeben, gespeichert und danach wieder geleert wird, wird beim Drucken eine Exception geworfen.</p> <p><i>PrintService.exe Warning: 0 : System.ArgumentException: Illegales Zeichen im Pfad.</i> <i>bei System.IO.Path.Combine(String path1, String path2)...</i> <i>PrintService.exe Information: 0 : Printserver continues after 1 errors.</i></p>

Korrigierte Fehler

ID	Text
2016031496	<p>Java-Client: Je nach Umgebung kann es ggf. vorkommen, dass während einem extrem langen Request die Verbindung zum Broker verloren geht. In diesem Fall bekommt der Client das Ende des Requests nicht. Es wird dauerhaft der Bitte-Warten-Dialog angezeigt und der Request kommt nie zu einem Ende.</p> <p><i>Über den parallel zum normalen Request laufenden Bitte-Warten-Mechanismus wird jetzt so eine Situation erkannt und entsprechend behandelt. Wenn so ein Fall eintritt, dann übermittelt der Broker das Ergebnis des Requests über die Bitte-Warte-Anfrage an den Client. Dieser reagiert darauf und behandelt das Ergebnis dann wie das normale Request-Ergebnis.</i></p>
2016031584	<p>Java-Client: Bei FreeTextInput-Comboboxen im Grid werden + Zeichen zu Leerzeichen, % Zeichen machen auch Probleme.</p>
2016031570	<p>Broker: Die Status NewButDeleted und UnchangedButDeleted eines DevFrameworkObjects bleiben bei einem erneuten Delete-Aufruf nicht erhalten sonder werden in Deleted überführt.</p>
2016031636	<p>Broker: Bei der Anzeige von Reports kann es ggf. zu erheblichen Verzögerungen kommen, wenn die entsprechende Repository-Tabelle sehr viele Datensätze enthält. Die Aufräum-Routine läuft dann in einen Timeout und diese vielen Sätze bleiben dadurch stehen. Diese Situation kann sich dadurch nicht mehr von selbst lösen.</p> <p><i>Es wurden 2 Verbesserungen vorgenommen:</i></p>

	<p>1.) Das Aufräumen abgelaufener Reports erfolgt jetzt in kleinen 100er Portionen, damit es keinen Timeout mehr gibt die Tabelle nicht unnötig lange blockiert wird.</p> <p>2.) Die Aufräum-Routine wird jetzt asynchron ausgeführt. Somit verzögert sie nicht mehr die Anzeige des Reports im Browser.</p>
2016031678	Publish/Publish2Go: Beim Publish von Service Hosts kann es zu SecurityExceptions kommen, weil auf Registry-Schlüssel nicht zugegriffen werden kann.
2016031576	Die FSROWID-Routine findet bei einer Oracle-Datenbank keine Tabellen, wenn die Datenbank einen so alten Stand hat, dass noch keine FSROWVERSION-Spalten angelegt wurden. Somit ist es nicht möglich eine SQL-Business-Datenbank auf eNventa 3.6 upzudaten.
2016021041	IDE: Wird eine als override definierte Methode kopiert, wird diese im Ziel als virtual eingefügt.
2016031409	IDE: Die Reihenfolge von JoinTables kann im ServiceRelease geändert werden.
2016031534	<p>IDE: Wenn in der Ableitung eines Forms Controls existieren, deren Namen bereits im Basis-Form verwendet werden, dann gibt es weder einen Compile-Error noch eine Warning. In der Folge kann es zu Fehlern führen, die ggf. erst später auffallen.</p> <p><i>Jetzt wird eine Warning in dem Form ausgegeben, in dem ein Control hinzugefügt wurde, dessen Name bereits im Basis-Form verwendet wird. Das greift sowohl bei Ableitungen als auch bei Customizings.</i></p>
2016031682	IDE: Nach dem Löschen eines kompletten Namespaces kann es beim Anmelden an FrameworkStudio zu einer unhandled Exception kommen.
2016031709	IDE: Wird am Ende des Compiles eines Service-Release der Workspace completed, dann werden abhängige Workspace-Mappings angepasst. Somit benutzen die abhängigen Packages anschließend den neuen Stand ohne dass diese kompiliert werden müssen. Wenn diese Workspace-Mappings aber eine Context-Version des kompilierten Packages verwenden, dann wird dabei das Workspace-Mapping so verändert, dass diese Context-Version entfernt wird. In der Folge werden die falschen DLLs verwendet und es kann zu Laufzeit-Fehlern kommen.
2016031435	<p>IDE: Beim Ausführen von 'Sync From AccessUnits' kann es zu einem Ungültiger threadübergreifender Vorgang kommen der eine Exception zur Folge hat.</p> <p><i>System.Exception: Ungültiger threadübergreifender Vorgang: Der Zugriff auf das Steuerelement [...] erfolgte von einem anderen Thread als dem Thread, für den es erstellt wurde.</i></p> <p><i>--> FrameworkSystems.FrameworkExceptions</i></p>
2016031705	<p>IDE: Beim Kompilieren von Components tritt eine unklare Warning (2035) mit Hinweis auf eine NullReferenceException auf, wenn an einer Component mehreren Virtual Columns die gleiche SortOrder zugeordnet ist und die Component nicht ausgecheckt ist.</p> <p><i>#warning: 'FSWarn[2035] SQL-Parser-Error in ComponentQuery (details see following comment) '</i></p> <p><i>System.NullReferenceException: Der Objektverweis wurde nicht auf eine Objektinstanz festgelegt.</i></p> <p><i>bei FrameworkSystems.FrameworkStudio.General.DevObjects.Component.ComponentRecord.CreateCodeQueryInfo(CodeBlock code, Int32 tab)</i></p>
2016021342	Doku-System: Der HTML-Export für Component-Properties liefert im Custom

Package eine SQL Exception.

2016031397 Doku-System: In der HTML-Preview des Dokumentationssystems werden Bilder teilweise nicht angezeigt.

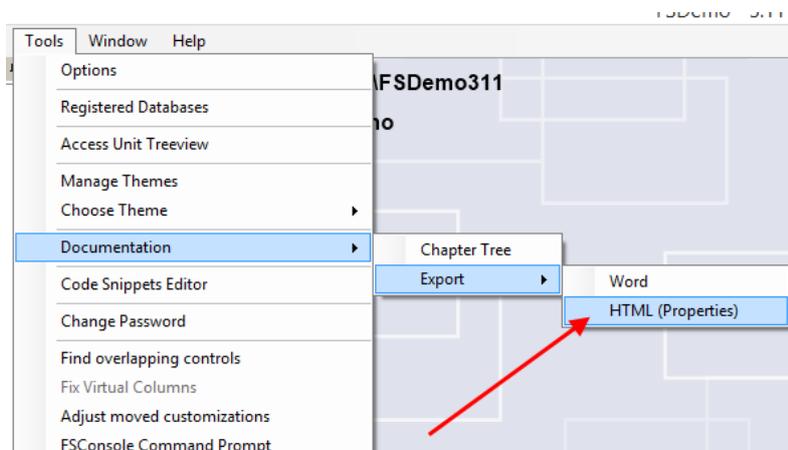
Framework Studio 3.11.2 (26.02.2016)

HTML-Export für Component-Properties

Für das Dokumentationssystem wurde ein neuer Export-Modus implementiert. Dieser ist in der Lage, Dokumentationen von Component-Properties in einzelne HTML-Dateien zu rendern.

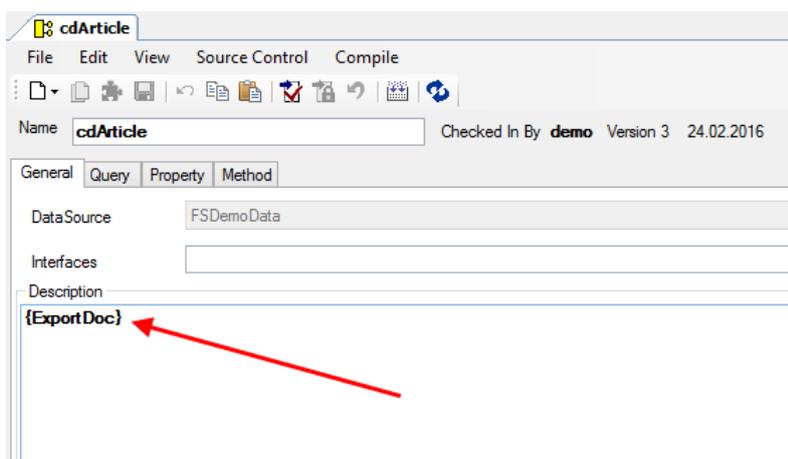
Soll die Dokumentation eines Properties z.B. an der Oberfläche der Endanwendung angezeigt werden, ist dies nun mit Hilfe des Browser-Controls möglich.

Der neue Export-Modus befindet sich wie nun auch der bisherige Word-Export-Modus im Menü **Tools** → **Documentation** → **Export** → **HTML (Properties)**.



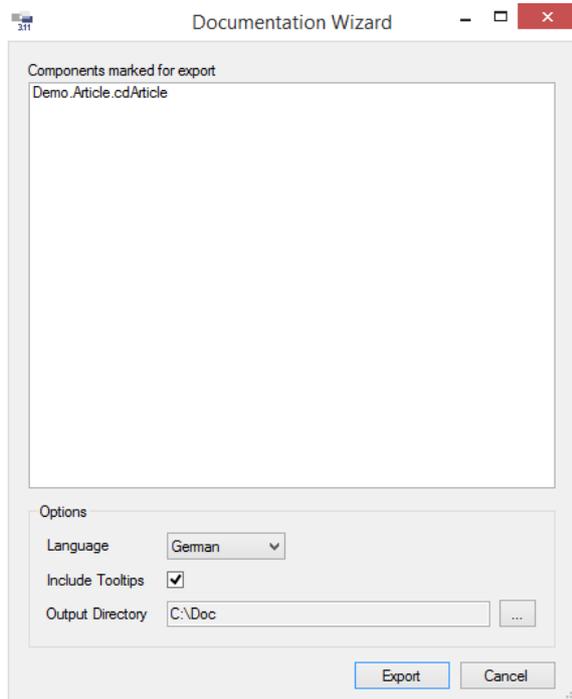
Markieren der Components

Eine Endanwendung besteht im Normalfall aus extrem vielen Components und Component-Properties. Der neue Export-Modus wird aber meist nur für einen gewissen Bereich der Anwendung benötigt. Deshalb muss auf Component-Ebene definiert werden, welche Components beim neuen Export-Modus beachtet werden sollen.



Um eine Component für den neuen Modus zu markieren, muss die Zeichenfolge **"{ExportDoc}"** in das Description-Feld der Component eingetragen werden. Nun werden beim Export alle dokumentierten Properties der markierten Component exportiert.

Export-Dialog



Wird der neue Export-Dialog aufgerufen, wird eine Übersicht mit allen markierten Components angezeigt.

Language:

Definiert die Sprache, in der die Component-Properties gerendert werden sollen

Include Tooltips:

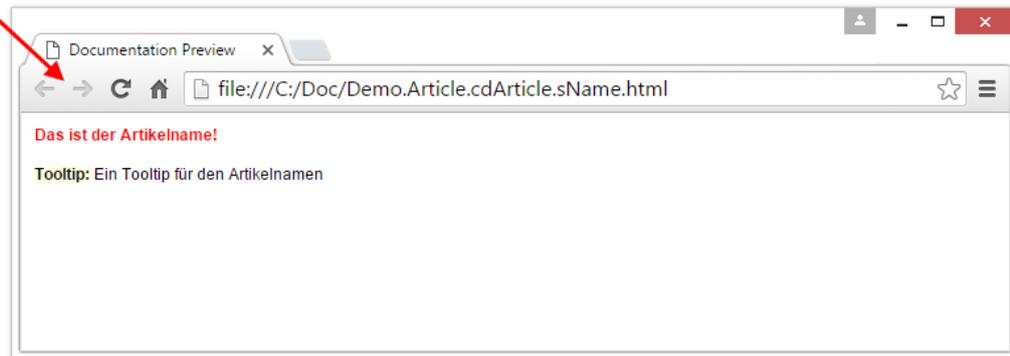
Definiert, ob an den Component-Properties definierte Tooltips zuzüglich zur eigentlichen Dokumentation gerendert werden sollen.

Output Directory:

Über den Button "...“ muss ein Zielverzeichnis für den HTML-Export definiert werden. Dieses Verzeichnis muss leer sein.

Ausgabe

Name	Änderungsdatum	Typ	Größe
styles	18.02.2016 15:27	Dateiordner	
Demo.Article.cdArticle.decPrice.html	18.02.2016 15:27	Chrome HTML Do...	1 KB
Demo.Article.cdArticle.sArticleNr.html	18.02.2016 15:27	Chrome HTML Do...	1 KB
Demo.Article.cdArticle.sColor.html	18.02.2016 15:27	Chrome HTML Do...	1 KB
Demo.Article.cdArticle.shtRadioTest.html	18.02.2016 15:27	Chrome HTML Do...	1 KB
Demo.Article.cdArticle.sID.html	18.02.2016 15:27	Chrome HTML Do...	1 KB
Demo.Article.cdArticle.sName.html	18.02.2016 15:27	Chrome HTML Do...	1 KB



Beim Export wird jedes Property einer markierten Component durchlaufen und in eine separate .html-Datei geschrieben. Der Name dieser Dateien setzt sich allein aus dem Fullname der Component und dem Namen des Properties zusammen.

Wird der Export-Ordner nun an einem im Netzwerk zugänglichen Pfad abgelegt (vorzugsweise auf dem Webserver, auf dem auch schon die bisherige Dokumentation liegt), können die Property-Dokumentationen in der Endanwendung mittels eines Browser-Controls eingebunden werden.

Korrigierte Fehler

ID	Text
2016010769	Java-Client: Durch das enablen derTabPage oder des TabbedWindows geht die IsEditable-Eigenschaft der Unter-Controls verloren.
2016020986	Java-Client: Der "Bitte warten..."-Dialog kann im JavaClient nach einem Verbindungsabbruch zum Server offen stehenbleiben.
2016021174	<p>Oracle: Die von der Datenbank-Struktur-Analyse erzeugten Statements übergeben bei VARCHAR2-Spalten die die Anzahl der Zeichen. Diese werden von Oracle standardmäßig als Bytes interpretiert, was beim Einsatz von Unicode-Codepages dazu führt, dass sich die tatsächliche Anzahl der Zeichen halbiert. In der Folge werden diese Spalten auch immer wieder von der Struktur-Analyse gemeldet.</p> <p><i>Jetzt wird bei CHAR- und VARCHAR2-Spalten der Zusatz "CHAR" übergeben. z.B. SpalteAbc VARCHAR2(80 CHAR)</i></p>
2016021052	Publish2Go: Wenn man aus einem Publish2Go-Paket einen ServiceHost published, und Service-Releases berücksichtigt werden sollen dann führt dies zu einer Exception.
2016021269	<p>FS IDE: Wenn es in einer DBTable mehrere Indizes oder mehrere Spalten mit ähnlichen Namen gibt, die sich nur durch ihre Groß-Klein-Schreibung unterscheiden, dann führt dies zur Laufzeit zu Fehlern in der Datenbank. Die Datenbanken unterscheiden bei den Namen keine Groß-Klein-Schreibung.</p> <p><i>Jetzt werden in solchen Fällen Compiler-Warnungen erzeugt. In einer nächsten FS-Version führen diese Fälle zu einem Compile-Error.</i></p>
2015110143	Add-Ins: Das Package Merge Add-In führt unter Umständen Workflowlink-Parameter nicht korrekt zusammen, wenn im Quellpackage einige Parameter eines Links geerbt wurden, jedoch nicht alle Parameter.
2016021192	Dokumentationssystem: Wenn ein Panel eine ButtonGroupDataSource definiert hat, welche eine Dokumentation enthält, wird diese nicht in die gerenderte Dokumentation exportiert.

Framework Studio 3.11.1 (29.01.2016)

Authentication Service 3.11.1 (Update optional)

Es wurde lediglich eine unkritische Korrektur am Authentication Service Configuration Editor durchgeführt, wodurch eine unnötige Hinweismeldung unterdrückt wird, die unter Umständen nach der Installation eines Authentication Services im Authentication Service Configuration Editor erscheint, siehe Bug 2015120434.

Bug-ID	Beschreibung
2015120434	Authentication-Service: Beim Beenden des Config-Editors, kann es vorkommen, dass folgende Meldung angezeigt wird, obwohl alle Änderungen vor der Installation bereits gespeichert wurden: „Configuration Changed. Service need to be restarted. Restart Service?“

Dynamische Gridspalten

Bisher konnten für vorhandene Properties in einem Grid zur Laufzeit dynamisch Spalten hinzugefügt werden. Dazu konnte an einem Grid-Control (Typ *GridColumnAction*) die Methode *AddColumn(GridColumnDefinition columnDefinition)* aufgerufen werden. Das Entfernen dynamischer Spalten zur Laufzeit hingegen war bisher nicht möglich.

Nun können dynamisch zur Laufzeit hinzugefügte Spalten auch wieder entfernt werden. Dazu stehen an der *GridColumnAction* folgende Methoden zur Verfügung:

- **RemoveColumn(GridColumnDefinition columnDefinition):** Entfernt die dynamische Gridspalte zum in der *GridColumnDefinition* angegebenen Property.
- **RemoveColumn(string property):** Entfernt die dynamische Gridspalte zum angegebenen Property.
- **RemoveColumn(GridColumnControlAction gridColumnControlAction):** Entfernt die dynamische Gridspalte. Die *GridColumnControlAction* wird von den Methoden *AddColumn* und *GetDynamicColumnAction(string property)* zurückgegeben.
- **RemoveAllDynamicColumns():** Entfernt alle dynamischen Gridspalten.

Ein Property muss nicht eine Eigenschaft direkt an der Component sein, die eine Gridzeile darstellt, sondern kann auch ein Property einer Unterkomponente darstellen. In diesem Fall wird als *property* der Pfad zum Property durch Punkte separiert angegeben.

Beispiel:

Einem Grid *grdDynObjColl* ist die Collection *this.oDynObjColl* als Datasource zugewiesen. Die Collection enthält Elemente vom Typ *cDynObj*. Am *cDynObj* gibt es ein Property *oDevObj* vom Typ *DevFrameworkObject*. *oDevObj* werden Instanzen von *cdCompany* zugeordnet. An *cdCompany* ist ein Property *sName* definiert.

Das Hinzufügen der Gridspalte für das Property *sName* könnte wie folgt implementiert werden:

```
this.grdDynObjColl.Add( new GridColumnDefinition() {
    ComponentType = this.oDynObjColl[0].oDevObj.GetType(),
    PropertyMdt = this.oDynObjColl[0].oDevObj.GetPropertyMdt("sName"),
    Property = "oDevObj.sName"
});
```

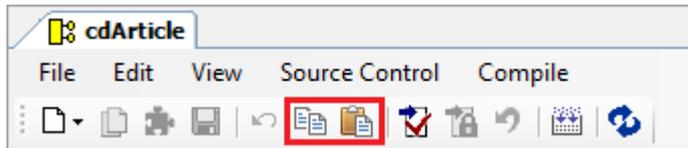
Das Entfernen könnte beispielsweise so implementiert werden:

```
this.grdoDynObjColl.RemoveColumn(„oDevObj.sName“);
```

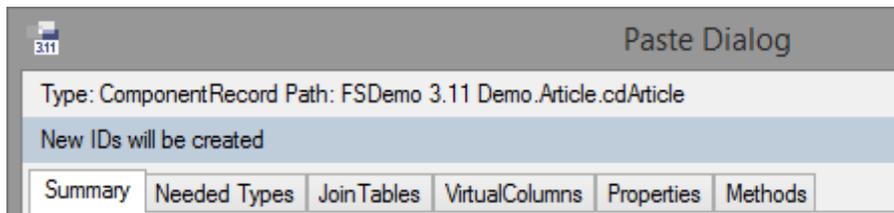
Überarbeitung von Copy & Paste

In dieser Version gibt es einige Copy & Paste Überarbeitungen

1. Neben den Menüeinträgen existieren nun Buttons mit denen ein Element kopiert oder eingefügt werden kann.



2. Es ist nun möglich einzelne Methoden, Controls, Form- und Component-Properties mit der internen ID zu kopieren. Die ID wird bei Unterelementen nur dann übernommen wenn in demselben Package aber unterschiedlichen Versionen kopiert wird und die ID der Parent-Elemente übereinstimmt. Das Kopieren mit ID ist ab FS 3.11.1 möglich und wird für ältere Versionen nicht unterstützt.
3. Beim Einfügen von Methoden, Controls und Properties in ein anderes Parent-Element wird der PasteDialog angezeigt. Beim Einfügen von Unterelementen aus einer Version < 3.11.1 wird der Dialog nicht angezeigt.
4. Bei Forms und Components kommt nun der neue Paste-Dialog zum Einsatz
5. Die Elemente des PasteDialogs sind nun nach MergeAction (Insert, Replace ...) und Namen sortiert.
6. Von dem Element welches eingefügt werden soll wird im PasteDialog der Type und der Pfad angezeigt. Diese Informationen sind ab FS 3.11 verfügbar und werden bei Elementen die aus älteren Versionen stammen nicht angezeigt.



Korrigierte Fehler

ID	Text
2015120475	<p>JavaClient: Wenn in einem Grid die Tab-Taste gedrückt wird, dann wird der Fokus in die nächste editierbare Zelle gesetzt. Dabei werden die Zellen mit dem Event OnEditingStarting auf ihre Editierbarkeit getestet. Wird in einem sehr großen Grid in sehr vielen oder gar allen Zellen dabei die Bearbeitung verhindert (e.Cancel = true), dann sucht der JavaClient sehr lange vergeblich nach einer editierbaren Zelle.</p> <p><i>Es wird jetzt maximal 2 Sekunden nach der nächsten Zelle gesucht. Das begrenzt die Wartezeit und die Anzahl der OnEditingStarting-Events.</i></p>
2015120353	<p>Java-Client: Wenn im SelectionChanged Event einer Combobox eine MessageBox ausgegeben wird, dann führt dies zu verschiedenen Fehlern.</p> <ol style="list-style-type: none"> 1.) Die MessageBox reagiert erst beim 2. Click, weil beim ersten Click der Combobox DropDown geschlossen wird. 2.) Es werden Java-Exceptions ausgelöst (Cannot format given Object as a

	Number)
2015120658	Java-Client: Wenn in einem OnValidate-Event in einem TabbedWindow per DataSource die aktive Registerkarte gewechselt wird, dann kann es passieren, dass das OnValidate-Event mehrfach ausgelöst wird. Im 2. Event kann es dabei zu falschen Werten kommen.
2015067542	<p>Java-Client: Wird an einem Feld eine Änderung vorgenommen, dann kann es passieren, dass das OnValidate-Event nicht oder das OnLeave-Event ohne HasChanged gefeuert wird. Dies tritt dann auf, wenn nach der Bearbeitung ein Event ausgelöst wird, das keinen Focus-Wechsel zur Folge hat - z.B. Aufruf eines Menü-Items oder Click in ein Grid, an dem OnDrag aktiv ist.</p> <p><i>Jetzt wird in solchen Fällen, in denen in einem TextFeld eine Änderung vorgenommen wurde, das OnValidate- bzw. OnLeave-Event vorher ausgeführt.</i></p> <p><i>Achtung! Wenn ein solcher Validate z.B. das Grid aktualisiert, in das gerade geklickt wird, dann kann es passieren, dass die RowID in den EventArgs nicht mehr stimmt. Dasselbe kann auch auftreten, wenn ein Grid über eine durch das Validate ausgelöste OnAfterRequest-Methode aktualisiert wird. In solchen Fällen muss die Implementierung des Events entsprechend abgesichert werden.</i></p>
2015120579	Broker / Printing: Wird ein Crystal Report mit dem Format HTML als Byte-Array exportiert, dann produziert die Crystal-Reports-Runtime eine FileNotFoundException.
2015120576	Broker / Printing: Crystal Reports Runtime kann ab Version 13.0.10 nicht verwendet werden. Beim Druck eines Reports kommt es im Broker zu einer NullRefernceException.
2015120583	IDE: Bei doppelten Gridspalten beim Ableiten oder Customizen von Forms, wird bisher eine Warning (FSWarn[2032]) zur Compilezeit ausgegeben. Da in einem solchen Fall zur Laufzeit eine Exception geworfen wird, muss zur Compilezeit schon ein Fehler (FSErr[1076]) ausgegeben werden.
2015120504	IDE: Vererbte Dokumentationen von Properties werden an GridColumns nicht beachtet.
2015120513	IDE: Im Run bzw. Publish Wizard wird die Darstellung des Start- und Backgroundimage bei einer Änderung des Bilds nicht aktualisiert.
2015120625	IDE: Die Überprüfung der Method Observer beim Import einer Package-Version führt zu einer Exception.
2015120629	IDE: Im Theme-Manager kann ein Theme nicht auf deprecated gesetzt werden, weil die CheckBox disabled ist.
2015120610	IDE: Im Customzing-Package kann an einem Component-Property mit dem Typ Relation das Individual Load Command nicht überschrieben werden. Der Fehler tritt auf, wenn im Basis-Package bei der letzten Änderung der Component das entsprechende Property nicht angefasst wurde.
2016010728	IDE: Labels innerhalb einer Registerkarte, die einen Tooltip aber keine Dokumentation besitzen werden nicht in der gerenderten Dokumentation angezeigt.

Framework Studio 3.11 Neuheiten

Änderungen bei Systemanforderungen

Es wird jetzt zusätzlich Windows 10 unterstützt.

Update-Hinweise

Beim Update von **eNVenta**-Packages auf Framework Studio 3.11 / **eNVenta** 3.6 muss unbedingt die Maintenance-Routine **Update / eNVenta 3.6 / Move MainButtons into StackPanel** ausgeführt werden. Ansonsten kann es u.a. dazu führen, dass z.B. Button-Events nicht mehr gefeuert werden. Es ist wichtig, dass diese Routine in JEDEM Customizing-Package vollständig ausgeführt wird. Genauere Details zu dieser Routine finden Sie im **eNVenta** Update-Dokument.

FSROWID

Die ROWID-ROWVERSION-Logik wurde überarbeitet. In jeder Datenbank-Tabelle gibt es jetzt eine neue Spalte FSROWID, die aus einer GUID besteht.

Diese FSROWID bietet im Gegensatz zur bisherigen ROWID folgende Vorteile:

- 1.) Die FSROWID identifiziert mit der GUID einen Datensatz dauerhaft und absolut eindeutig. Damit kann die FSROWID auch in Relationen verwendet werden.
- 2.) Auf SQL-Server wurde die bisherige ROWID mithilfe einer Identity-Spalte abgebildet, die ausschließlich von der Datenbank erzeugt wurde. Diese ist lediglich pro Datenbank-Tabelle eindeutig und kann nicht manuell bestückt werden.
- 3.) Auf Oracle wurde die interne ROWID-Spalte verwendet. Diese hat jedoch einige Einschränkungen, welche eine Verwendung in einer Relation verhindern.
- 4.) Durch die Umstellung konnte der Historie-Modus komplett fertiggestellt werden (siehe auch → Auslagerung historischer Daten) und weitere Entwicklungen können in Zukunft einfacher realisiert werden.
- 5.) Durch die Verwendung eines Default-Constraints wird die neue FSROWID-Spalte automatisch von der Datenbank befüllt, wenn beim Insert kein Wert übergeben wird. So können auch Daten außerhalb von Framework Studio oder mit einer älteren Version problemlos eingefügt werden.

Update der Datenbank

Die Daten-Datenbanken müssen mithilfe einer Routine um die Spalte FSROWID erweitert werden. Diese steht im Publish-Wizard unter dem Button „FSROWID-Update“ zur Verfügung. Der Update-Assistent kann diese Erweiterung nicht vornehmen – nicht umgestellte Datenbank-Tabellen zeigt er als fehlerhaft an.

Die Umstellung kann schon im Vorfeld des Updates erfolgen. Die älteren Programm-Versionen können ohne Einschränkungen mit einer umgestellten Datenbank arbeiten. Sofern die Programm-Versionen dies zulassen, ist auch ein Parallel-Betrieb möglich.

SQL-Server

Die Datenbank wird für die Erweiterung in den Einzelbenutzer-Modus versetzt und kann in dieser Zeit nicht verwendet werden. Bestehende Verbindungen werden getrennt. Die Routine setzt während der Umstellung das Wiederherstellungs-Modell der Datenbank auf „Einfach“. Dadurch wird verhindert, dass die Transaktions-Logs überlaufen. Stellen Sie sicher, dass vor der Umstellung die Sicherung der Datenbank auf den aktuellen Stand ist.

Bei der Umstellung selber werden einmal alle Tabellen umhergeschaufelt. Es wird eine neue Tabelle mit der Spalte FSROWID angelegt, dann werden die Daten in die neue Tabelle kopiert und anschließend wird die alte Tabelle gelöscht. Der Primary-Key der Tabelle wird dabei von ROWID aus FSROWID geändert. Für die ROWID-Spalte wird ein zusätzlicher Index erzeugt. Durch das Verschieben der Daten werden die Datenseiten neu organisiert, was in der Regel dazu führt, dass in der Datenbank Platz frei wird. Die Routine beginnt mit der kleinsten Tabelle, damit der frei werdende Platz später bei den großen Tabellen zur Verfügung steht und sich so das Wachstum der Datenbank in Grenzen hält.

Dennoch sollte ein Wachstum der Datenbank einkalkuliert werden. Pro 1 Mio. Rows werden ca. 70 - 100 MB zusätzlicher Platz benötigt. Insgesamt wächst so die Datenbank um etwa 10 - 20%.

Die Routine kann, wenn sie z.B. zu lange dauert, jederzeit abgebrochen werden. Die aktuelle Tabelle wird dann noch fertig umgestellt und anschließend werden die Einstellungen in der Datenbank wieder auf die ursprünglichen Werte zurückgesetzt, sodass sie anschließend wieder mit der älteren Programm-Version verwendet werden kann. Zu einem späteren Zeitpunkt kann diese Routine für die übrigen Tabellen erneut gestartet werden.

Oracle

Die Umstellung sollte zu einem Zeitpunkt erfolgen, an dem möglichst nicht produktiv gearbeitet wird. Insbesondere bei den großen Tabellen kann die Umstellung dazu führen, dass SQL-Anfragen anderer Anwendungen sich erheblich verzögern.

Die Routine erweitert die Tabellen um die neue Spalte FSROWID und legt für diese Spalte einen neuen Index an.

Dabei werden sehr viele Rollback-Informationen erzeugt. Im Grunde wird einmal die komplette Datenbank angefasst. Es muss sichergestellt werden, dass die Rollback-Segmente genügend Platz haben. Ggf. kann es sinnvoll sein, den Parameter UNDO_RETENTION herunterzusetzen, damit die Rollback-Segmente schneller wieder freigegeben werden und nicht überlaufen. Der Datenbank-Administrator muss dies ggf. mit den Backup-Einstellungen abstimmen. Es müssen auch die ArchiveLogs beachtet werden. Aufgrund der großen Masse an Änderungen ist es sogar sinnvoll, die ArchiveLogs während der Umstellung komplett zu deaktivieren.

Kalkulieren Sie auch ein Wachstum der Datenbank ein. Pro 1 Mio. Rows werden ca. 30 MB für die neuen Indizes (Tablespace INDEX1) und ca. 10 – 20 MB für die neue Spalte (entsprechende DATA-Tablespaces) benötigt.

Die Routine benötigt je nach Performance des Systems ca. 10-30 Sekunden pro 1 Mio. Rows. Sie kann jederzeit abgebrochen werden. Die aktuelle Tabelle wird noch fertig umgestellt. Die Routine kann zu jeder Zeit neu gestartet werden, um die verbleibenden Tabellen umzustellen.

Neuer Typ FSSystemGuid

Für die FSROWID gibt es einen neuen Typ **FSSystemGuid**. Dieser verpackt einen Wert vom Typ System.Guid.

Framework Studio bietet schon lange einen eigenen Typ „guid“, welcher z.B. in SelectedRowsCollections verwendet wird. Dieser hat mit der System.Guid nicht viel zu tun. Um dies besser voneinander abzugrenzen, hat der neue Typ den Namen **FSSystemGuid** erhalten.

Änderungen in der Programmierung

Das Component-Property hat seinen Namen ROWID behalten. Dadurch arbeitet in den allermeisten Fällen der Bestands-Code ohne Probleme.

Es hat sich aber der Typ geändert. Bisher hatte das Property den Typ FSString, jetzt ist es FSSystemGuid. Dies hat zur Folge, dass entsprechender Code angepasst werden muss – z.B. wenn das ROWID-Property im eigenen Code verarbeitet oder als Parameter in einer Funktion übergeben wird.

Beispiele (FSString → FSSystemGuid):

```
HashSet<FSSystemGuid> rowIds = new HashSet<FSSystemGuid>(
    oCollPropCollP.Cast<cdCollectionProposal>()
        .Select(oCollPropDel => oCollPropDel.ROWID));
List<FSSystemGuid> oROWIDStoExclude = new List<FSSystemGuid>();
```

Datenbank-Trigger

Wenn Datenbank-Trigger die ROWID benutzen, dann funktionieren sie nach der Umstellung weiter. Die ROWID-Spalte bleibt erhalten.

Es sollte jedoch geprüft werden, ob ein Umbau auf die neue FSROWID-Spalte sinnvoll ist. Insbesondere unter SQL-Server ist das interessant, weil nach der Umstellung die neue FSROWID-Spalte den Primary-Key bildet.

Fremd-Datenbanken / importierte Strukturen

Wenn bei der Einbindung von Fremd-Datenbanken auf den ROWID-Mechanismus gesetzt wurde, dann muss dies jetzt angepasst werden. Mit dem Update auf Framework Studio 3.11 erhalten die bisherigen ROWID-Spalten den neuen Namen FSROWID und Framework Studio wird diese Spalten beim Insert mit einem Wert bestücken.

Wenn die Fremd-Datenbank nicht mit dem Umstellungs-Werkzeug angepasst werden kann, dann stehen folgende Lösungs-Wege zur Verfügung:

- 1.) Die ROWID-Spalte aus der DBTable entfernen und stattdessen andere PK-Spalten kennzeichnen. ACHTUNG: Dadurch wird aber ein Daten-Update in diesen Spalten verhindert.
- 2.) Wenn die Routine nicht laufen darf und 1.) keine Lösung ist, dann können in der Fremd-Datenbank die FSROWID-Spalten manuell ergänzt werden. Diese Spalten beeinträchtigen nicht die Verhaltensweise der Tabelle (solange unter SQL-Server keine Anpassung des Primary-Key-Constraints erfolgt). Die Spalte befüllt sich dank dem Default-Constraint eigenständig, wenn beim Insert kein Wert angegeben wird.

Anlegen von Spalte und Index unter SQL-Server:

```
ALTER TABLE <Tabelle> ADD FSROWID uniqueidentifier NOT NULL DEFAULT NEWSEQUENTIALID();
CREATE UNIQUE INDEX <Tabelle>_FSROWID ON <Tabelle> (FSROWID);
```

Anlegen von Spalte und Index unter Oracle:

```
ALTER TABLE <Tabelle> ADD FSROWID RAW(16) DEFAULT SYS_GUID() NOT NULL;  
CREATE UNIQUE INDEX IXROWID_<Tabelle> ON <Tabelle> (FSROWID) TABLESPACE INDEX1;
```

Java-Client

Vereinheitlichung des Kommunikationsformats zwischen Broker und Client

Der Java-Client wirft Exceptions, wenn spezielle Zeichen per Xml kommuniziert werden, die im Xml per Definition nicht erlaubt sind (z.B. vertikaler Tabstopp). Daher wurden von Framework Studio bisher Strings meist URL-kodiert übertragen. Große Teile wurden jedoch nicht kodiert. So mussten beispielsweise Custom Control Entwickler manuell kodieren und dekodieren, um Properties, Event-Parameter und Action-Parameter sicher übertragen zu können. Diese Arbeit übernimmt nun Framework Studio.

Außerdem wurden bisher Null-Werte nicht einheitlich sondern abhängig vom Datentyp als "null" oder "" übertragen.



Achtung: Die Umstellung führt zu Verhaltensänderungen. Daher ist unbedingt der Abschnitt *Nötige Anpassungen* zu beachten.

Schnittstellendefinition

1. Strings werden URL-kodiert übertragen, andere Datentypen nicht.
2. Als Null-Wert wird bei allen Datentypen der Leerstring "" verwendet.
3. Spezielle Datasources, welche zwar technisch gesehen vom Typ *String* sind, jedoch ein von Framework Studio vorgegebenes Datenformat darstellen, werden nicht URL-kodiert. Beispiel: *SelectedTreeNodePath*.

Am Broker steht für die Umwandlung in Xml und umgekehrt nach neuem Schema die Singleton-Instanz der Klasse *FrameworkSystems.FrameworkBase.Converter.XmlConverter* zur Verfügung. Die Instanz kann über *XmlConverter.Get* abgerufen werden.

Die Klasse bietet beispielsweise die generischen Methoden *string ToXml<T>(T value)* und *T FromXml<T>(string xml)* sowie die Methode *bool IsConvertible(string typeName)*.

Am Client wird die Umwandlung in Xml und umgekehrt von den statischen Methoden *Object StringToDotNetType(String xml, int type)* und *String DotNetTypeToString(Object o, int type)* der Klasse *FrameworkSystems.FSJavaClient.UIWrapper.DotNetTypes* übernommen.

Custom Controls

Nach den obigen Regeln werden nun auch im Bereich der Custom Controls Action-Parameter, Event-Parameter und Eigenschaften kodiert. Dazu wurden die auch bisher zu verwendenden Schnittstellenklassen im Java-Client angepasst. Diese sind für

- **Eigenschaften:** *FrameworkSystems.FSJavaClient.JavaClientAPI.FSProperties*
- **Action-Parameter:** *FrameworkSystems.FSJavaClient.JavaClientAPI.ActionParams*
- **Event-Parameter:**
FrameworkSystems.FSJavaClient.EventAndEventArgs.DevEventParameters

Datasources konnten nicht umgestellt werden, denn hier gibt es bisher keine Klasse am Client, welche die Kodierung (für Standard-Datasources) übernimmt. Eine Umstellung hätte

zur Folge, dass alle Custom Controls, welche einen Standard-Datasource verwenden, angepasst werden müssten.

Nötige Anpassungen

Im Wesentlichen gibt es drei Bereiche, in denen unter Umständen Anpassungsbedarf besteht:

1. Eigene Custom Controls:

Nur wenn eigene Custom Controls entwickelt wurden und diese nicht(!) bereits die Klassen *FSPProperties*, *ActionParams* und *DevEventParameters* verwenden, besteht Anpassungsbedarf. In diesem Fall sollten die genannten Klassen zum Auslesen von Properties, Auslesen von Action-Parametern und zum Setzen von Event-Parametern verwendet werden.

Eine zusätzliche URL-Kodierung ist grundsätzlich nicht mehr nötig, muss aber nicht zwingend ausgebaut werden, wenn sie bisher implementiert war.

2. Datasources vom Typ Object / Unsauberes Databinding:

Bisher hat Framework Studio keine Fehlermeldung erzeugt, wenn einem Datasource ein Property zugewiesen wurde, dessen Typ nicht unterstützt wird.

Beispiel

Einem Editfield *edtoValue* wurde als *DataSource* ein Property *oValue* vom Typ *Object* (external Type) oder *SelectedRowsCollection* zugewiesen.

In diesem Fall würde Framework Studio nun eine Fehlermeldung in folgender Form beim Kompilieren des betroffenen Forms erzeugen:

```
#error: 'FSErr[1074] Cannot bind property 'this.form.oValue' to control 'edtoValue' because property type 'object' is not supported.'
```

Diese Kompilierfehler müssen nachgearbeitet werden, z.B. indem als *DataSource* ein individual String Property zugewiesen wird, welches eine geeignete Konvertierung durchführt.

3. Eigener Client:

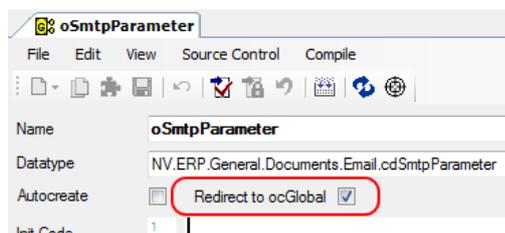
Wenn ein eigener Client implementiert wurde, muss ggf. die XML-Verarbeitung am Client entsprechend der Schnittstellendefinition angepasst werden (URL-Encoding von Strings, einheitliche Darstellung von Null-Werten).

Usability in der IDE

Lazy Global Objects und Safe Initialize

Um die Performance beim Start der Anwendung zu erhöhen, wurde eine Möglichkeit geschaffen, die Global-Objects erst bei Bedarf einzulesen.

Am GlobalObject gibt es eine neue Checkbox **Redirect to ocGlobal**.

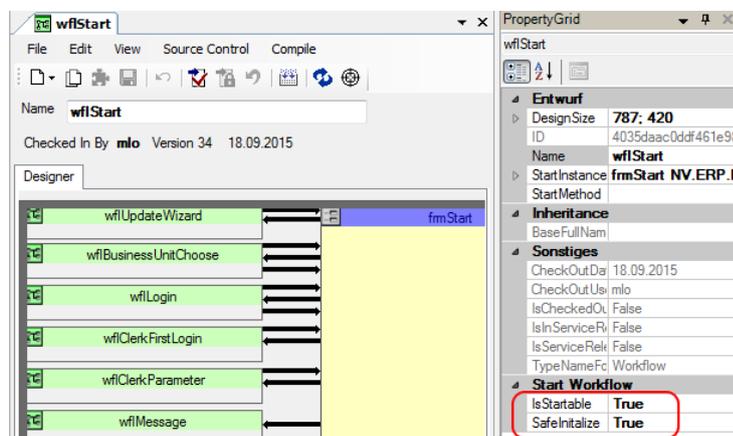


Wenn sie aktiviert ist, dann werden alle Zugriffe auf dieses GlobalObject in das gleichnamige Property in der Component FSGeneral.cGlobal umgeleitet. Dies betrifft Lese- und ggf. auch Schreibvorgänge. Wenn es sich bei dem cGlobal-Property um ein Readonly Property handelt, dann erkennt Framework Studio dies und wird das GlobalObject selber auch entsprechend readonly erzeugen.

Das cGlobal-Property kann wie gewohnt als Individual Property mit Membervariable gestaltet werden, welches die Daten erst beim ersten Zugriff aufbereitet.

Die Programmlogik greift z.T. schon sehr früh auf GlobalObjects zu. Wenn ein GlobalObject Daten aus der Datenbank einliest, dann kann es während einer Update-Situation zu Exceptions kommen. Diese müssen unterbunden werden. Wenn das Update abgeschlossen ist und das Programm normal arbeitet, dann dürfen derartige Exceptions hingegen nicht unterbunden werden.

Für diesen Zweck gibt es den sog. **Safe Initialize**. Dieser wird in den Properties des Start-Workflow eingestellt, denn der Start-Workflow gibt vor, wie der Einstieg in die Anwendung aussieht und ob es überhaupt eine Update-Logik gibt.



Zur Laufzeit kann dieser Modus über **this.Global.SafeInitializing** abgefragt werden.

Beispiel:

```
try
{
    [Aktion]
}
catch (Exception ex)
{
    if (this.Global.SafeInitializing)
    {
        [Fehler behandeln]
    }
    else
    {
        // Exception weiter werfen.
        throw;
    }
}
```

Der Modus muss mithilfe der Funktion **this.Global.EndSafeInitialize()** nach dem Start der Anwendung zurückgesetzt werden.

Copy & Paste

- Das Kopieren und Einfügen wurde um verschiedene Elemente erweitert und ausgebaut.
- Um Parallelentwicklungen zu realisieren, ohne mit den IDs in verschiedenen Versionen auseinander zu laufen, wurde das Kopieren mit ID realisiert.

Neue Elemente

Copy & Paste ist nun für folgende Elemente möglich:

- | | | |
|-------------------|------------------------|----------------------------|
| • Namespace2 | • Service ¹ | • TextCollection |
| • Resource | • ServiceHost1 | • GlobalEvent ² |
| • MDT | • Service Proxy1 | • Form |
| • DBTable | • Report | • Workflow |
| • DataContract | • Component | |
| • ServiceContract | • Collection | |

Allgemein

In Framework Studio haben die internen IDs aller Elemente eine große Bedeutung. Über diese IDs werden fast alle Beziehungen zwischen den Elementen abgebildet (z.B. Metadatentyp → DBColumn → Query-Column in der Component → Component-Property → Datenbing im Form-Control). Diese ID macht die Beziehungen unabhängig vom Namen der Elemente, wodurch sich z.B. eine Umbenennung automatisch in den generierten Code durchzieht.

Auch Customizing-Packages setzen auf diese ID. Es werden ganz normale Beziehungen zu den Elementen aus dem Basis-Package aufgebaut. Beim Customizing-Vorgang selber spielt die ID ebenfalls eine wichtige Rolle: Sie ist die einzige Beziehung, die so ein Element zum Basis-Package besitzt.

Diese ID kann aber auch zu unschönen Situationen führen. Wenn eine ID bzw. das entsprechende Element gelöscht wurde oder aus anderen Gründen nicht mehr verfügbar ist, dann können auch diese Beziehungen nicht mehr aufgelöst werden. Dadurch stehen auch gecustomizte Elemente nicht mehr zur Verfügung.

¹ Für die Ableitung des Elements ist Copy & Paste nicht verfügbar.

² Elemente desselben Typs können nicht zusammengeführt werden.

Beim Anlegen eines neuen Elements vergibt Framework Studio diese absolut eindeutige ID. Einmal vergeben kann sie vom Entwickler nicht mehr geändert werden.

Eine besondere Situation ergibt sich dadurch bei der parallelen Entwicklung in zwei unterschiedlichen Versionen oder auch in unterschiedlichen Packages. In beiden Versionen erhalten die Elemente jeweils unterschiedliche IDs. Solange innerhalb dieser Package-Version gearbeitet wird, ist dies kein Problem. Wenn aber Customizing-Packages auf diese Programmierungen aufsetzen (neue Elemente verwenden / oder neue Elemente customizen), dann treten Probleme später beim Update auf die neue Version auf. In der neuen Version haben die Elemente jetzt andere IDs und können so nicht mehr gefunden werden.

Um diese Schwierigkeiten zu reduzieren, gibt es ab Framework Studio 3.11 die Möglichkeit, beim Copy & Paste die IDs zu übernehmen.

Copy

Über das Kontextmenü eines Elements oder über den Menüpunkt **Copy** im **Edit**-Menü kann ein Element kopiert werden.

Paste

Wird ein Element ohne Basis und ohne ID kopiert, kann dieses überall eingefügt werden. Wird ein Element mit ID kopiert, kann dies nur eingefügt werden, wenn ein Element mit derselben ID noch nicht existiert. Eine Customization kann nur erstellt werden, wenn im Package noch keine Customization vorhanden ist und die passende Basis existiert.

Grundsätzlich wird zwischen zwei verschiedene Arten unterschieden, um ein Element einzufügen:

1. Element am Namespace eingefügt

Über das Kontext-Menü am Namespace kann das kopierte Element in den Namespace eingefügt werden. Dabei wird ein Neues Element erstellt.

2. Element in ein bestehendes Element des gleichen Typs einfügen

Über das Kontextmenü eines Elements oder über den Menüpunkt **Paste** im **Edit**-Menü kann das kopierte Element in ein bestehendes Element desselben Typs eingefügt werden. Dieser Vorgang stand in älteren Versionen schon unter Merge bei Components zur Verfügung.

Kompatibilität

Elemente bei denen Copy & Paste bereits vorhanden war (Components, Forms) können zwischen der alten und neuen Version kopiert werden. Das Kopieren mit ID wird jedoch in alten Versionen nicht unterstützt.

PasteDialog

Über den **Paste**-Dialog kann beim Kopieren in ein bestehendes Element ausgewählt werden, welche Unterelemente wie kopiert werden.

Zu beachten ist, dass der **Paste**-Dialog sehr viele Freiheiten bietet. Dies ermöglicht natürlich auch, fehlerhafte Stände zu erstellen.

Die zu kopierenden Unterelemente sind in Gruppen auf verschiedene Registerkarten aufgeteilt. Jeder **Paste**-Dialog enthält eine ‚Summary‘ und die ‚NeededTypes‘.

Summary

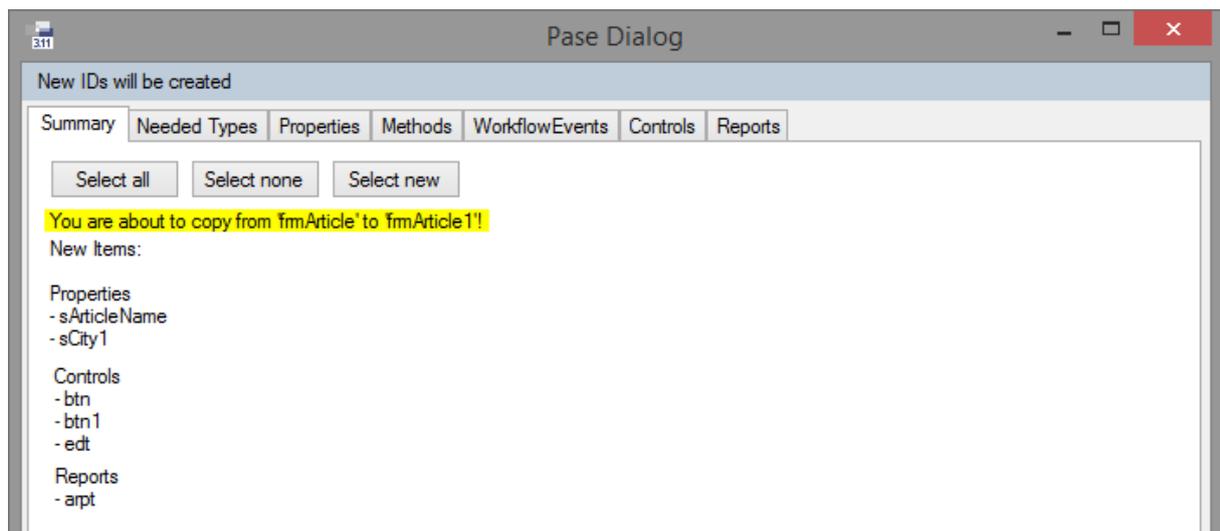
Auf der Summary werden fehlende Typen sowie neue Elemente angezeigt. Außerdem wird eine Warnung angezeigt, falls die ID und der Name des kopierten Elements nicht mit dem Ziel übereinstimmen.

Über verschiedene Buttons kann ausgewählt werden, welche Unterelemente der Registerkarte selektiert und beim Einfügen berücksichtigt werden.

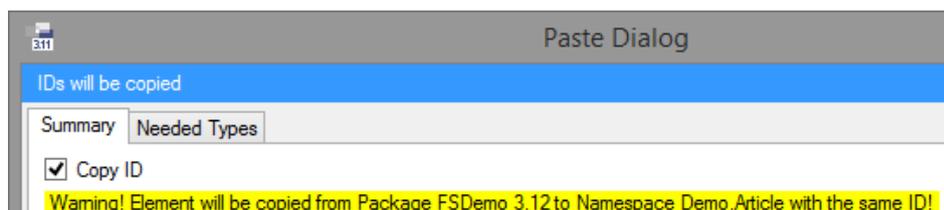
Select all: Alle Unterelemente werden selektiert.

Select none: Alle Unterelemente werden deselektiert.

Select new: Es werden nur neu hinzugefügte Unterelemente selektiert.



Beim Einfügen eines Elements am Namespace kann außerdem ausgewählt werden, ob die ID des Elements kopiert werden soll.



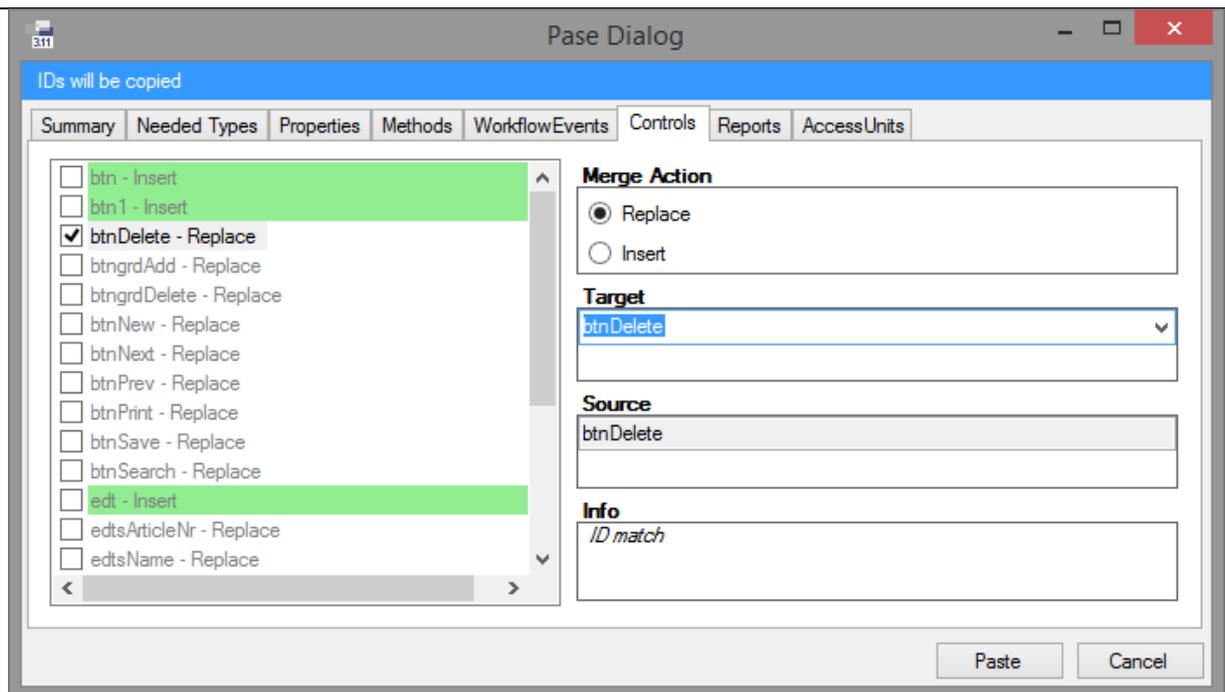
Needed Types

Bei den ‚Needed Types‘ werden alle Typen aufgelistet, die nicht zum Standard gehören. Diese können beliebig ausgetauscht werden. Das kann z.B. dann sinnvoll sein, wenn ein benötigtes Element vorher auch schon kopiert wurde und jetzt ggf. unter einem anderen Namen in einem anderen Namespace zu finden ist.

Zusammenführen von Elementen

Beim Zusammenführen von Elementen kann erreicht werden, dass Änderungen in einer Kopie nachgezogen werden. Was nicht möglich ist, sind gelöschte Unterelemente durch den Merge zu entfernen. Es werden nur neu hinzugekommene oder geänderte beachtet.

Über die einzelnen Registerkarten kann für jedes Unterelement separat entschieden werden, ob und wie es im Ziel eingefügt wird.



Unterelemente

- Unterelemente können einzelne Werte oder komplexere Elemente wie Methoden oder Properties darstellen.
- Alle Elemente werden gruppiert und in verschiedenen Registerkarten dargestellt.
- Die einzelnen Elemente können aus einer Liste ausgewählt werden.
- Für jedes Element kann separat entschieden werden, ob und wie es im Ziel eingefügt wird.
- Elemente, die im Ziel noch nicht vorhanden sind (Insert), werden in der Liste Grün dargestellt.

Merge Action

Es stehen verschiedene Merge-Operationen zur Verfügung:

- Insert
 - Das Element wird im Ziel neu eingefügt.
 - Der Eintrag wird in der Liste Grün dargestellt.
- Replace
 - Das ausgewählte Element wird im Ziel ersetzt.
 - Unter „Target“ kann das Element, das im Ziel ersetzt werden soll, ausgewählt werden.
- Replace with
 - Das ausgewählte „Source“-Element ersetzt das „Target“-Element.

Target

- Dieses Element wird beim Mergen ersetzt.

Source

- Das „Source“-Element ersetzt das „Target“-Element beim Merge.

Info

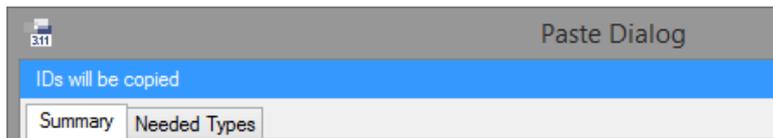
- Bietet Informationen über das Merge Element z.B. die Art der Übereinstimmung im Ziel.

Kopieren von ID's

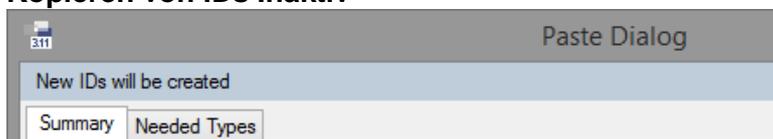
Um Parallelentwicklungen zu realisieren, ohne dass die internen IDs von Elementen in verschiedenen Versionen auseinanderlaufen, besteht nun die Möglichkeit die ID von Elementen zu kopieren.

In der obersten Leiste des **Paste**-Dialogs ist farblich gekennzeichnet, ob die IDs kopiert oder neu angelegt werden.

- **Kopieren von IDs Aktiv**

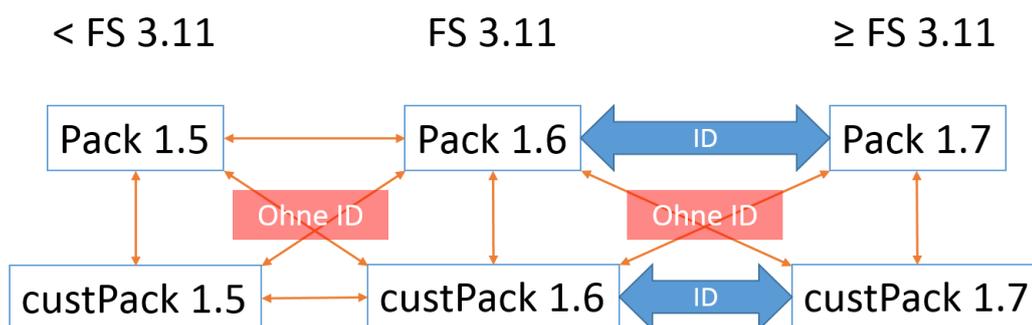


- **Kopieren von IDs Inaktiv**



Für das Kopieren von IDs gelten folgende Regeln:

- Wenn das zu kopierende Element am Namespace eingefügt wird und aus demselben Package, aber unterschiedlicher Version stammt, ist das Kopieren der ID's optional.
- Beim Zusammenführen von Elementen im selben Package, aber unterschiedlicher Version, wird die ID automatisch kopiert, wenn die IDs der beiden Elemente übereinstimmen, ansonsten nicht.
- Wenn das zu kopierende Element aus einem anderen Package stammt, ist das Kopieren der ID's nicht möglich.
- Ein Element kann nicht eingefügt werden, wenn ein anderes Element bereits dieselbe ID besitzt.



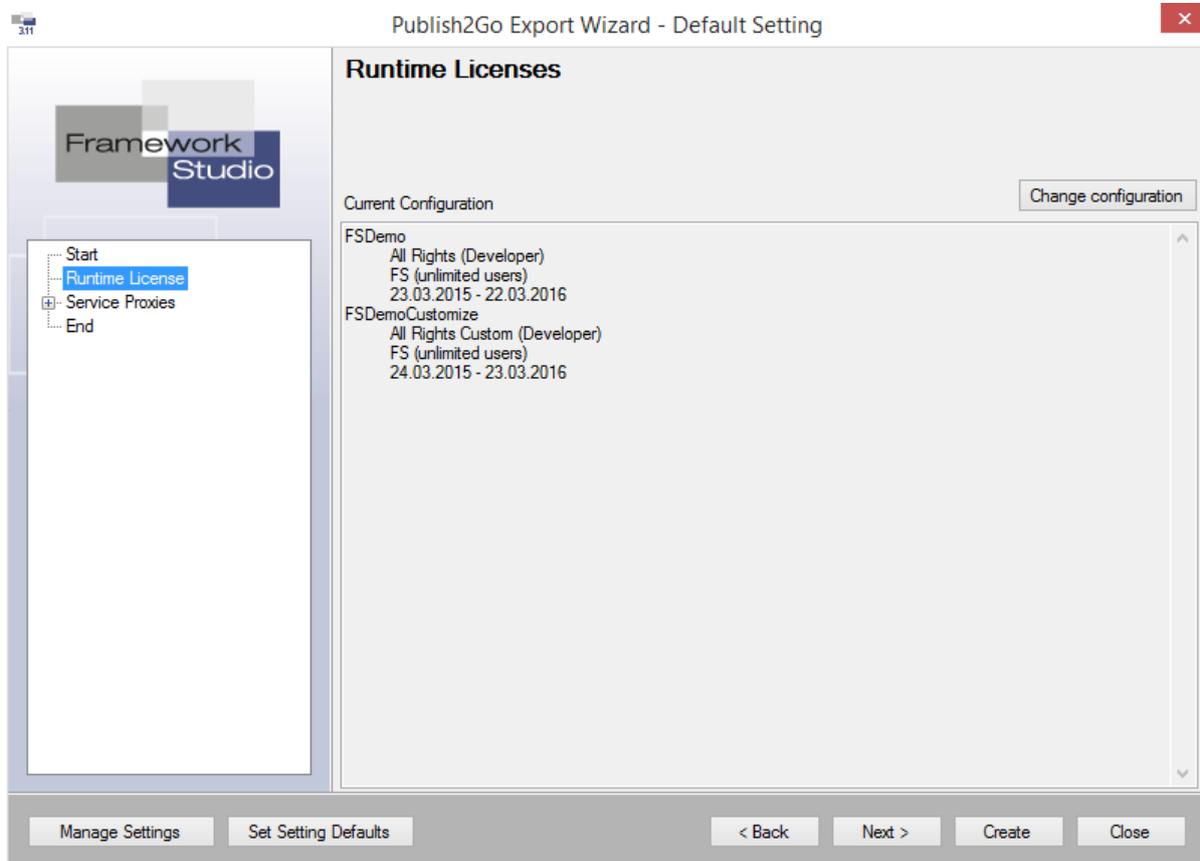
Ausnahmen

Das Kopieren von einzelnen Unterelementen wie Methoden und Properties mit ID ist bisher NICHT möglich! Das Kopieren mit ID bezieht sich bisher nur auf den kompletten LabeledRecord.

Um einzelne Elemente dennoch mit ID zu kopieren, können diese im **Paste**-Dialog mit dem Button **Select new** ausgewählt werden.

Beim Erstellen von Publish2Go Lizenzen exportieren

Beim Erstellen eines Publish2Go-Pakets ist es nun möglich, die zugehörigen Runtime Lizenzen gleich mit zu exportieren. Dazu steht im Publish2Go Wizard der aus dem normalen Publish Wizard bekannte Runtime License Manager zur Verfügung.



Nachdem das Paket erfolgreich erstellt wurde, befinden sich die Runtime Lizenzen im angegebenen Ordner.

Name	Änderungsdatum	Typ
32Bit	24.03.2015 13:58	Dateiordner
64Bit	24.03.2015 13:58	Dateiordner
All Rights Custom.RuntimeLicense.xml	24.03.2015 13:58	XML-Datei
All Rights.RuntimeLicense.xml	24.03.2015 13:58	XML-Datei
sqlitedb.p2go	24.03.2015 13:58	P2GO-Datei

Beim Starten des Publish2Go werden alle im Ordner befindlichen Lizenzen automatisch eingelesen.

Verbesserung beim Sortieren von Gridspalten im Formdesigner

Folgende Punkte wurden umgesetzt:

1. Drag & Drop im Grid sowie die Move-Funktionen aus dem Kontextmenü laufen bei Grids mit vielen Spalten deutlich performanter.

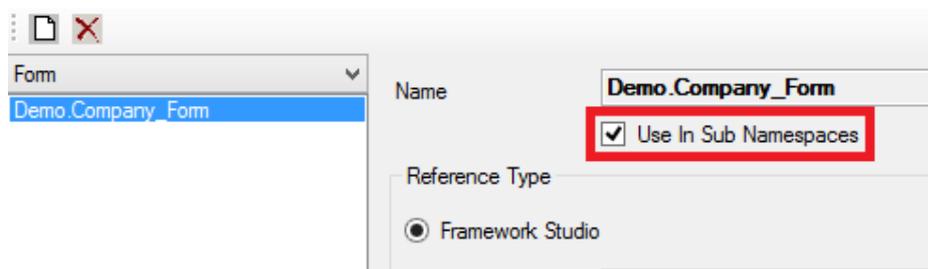
2. Drag & Drop einer Quellspalte auf eine Zielspalte verschiebt die Zielspalte und alle Spalten zwischen der Quellspalte und der Zielspalte in Richtung Quellspalte. (Vorher war das Verhalten nicht exakt vorhersehbar.)
3. Das Setzen der OrderID im PropertyGrid führt eine Renummerierung nach dem in 2. beschriebenen Prinzip durch.
4. Im ClassView werden Gridspalten nach OrderID, nicht nach Name sortiert angezeigt.
5. Im ClassView können Gridspalten per Drag&Drop sortiert werden.

Usability im Service Release

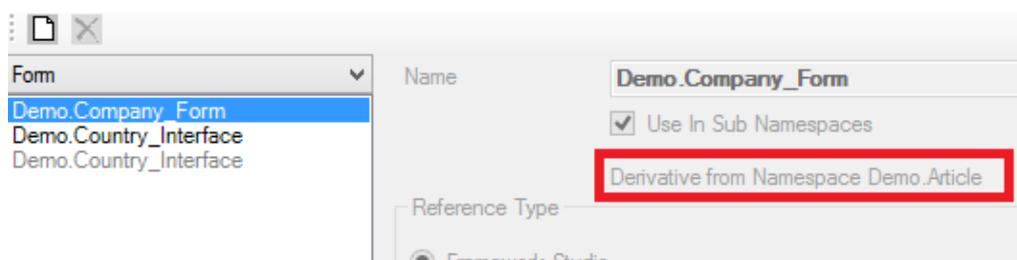
Im ServiceRelease wurden einige Textfelder bisher komplett disabled. Dies hatte zur Folge, dass der Text an diesen Stellen nicht aus den Feldern kopiert werden konnte. Um das Kopieren der Texte zu ermöglichen, wurden mit der Änderung die entsprechenden Textfelder auf ReadOnly gestellt.

Vererbung von Namespace-Referenzen

Es ist nun möglich Referenzen an Unter-Namespace zu Vererben. Hierzu muss in der Referenz-Verwaltung, bei den entsprechenden Referenzen, die Checkbox **Use in Sub-Namespace** gesetzt werden.



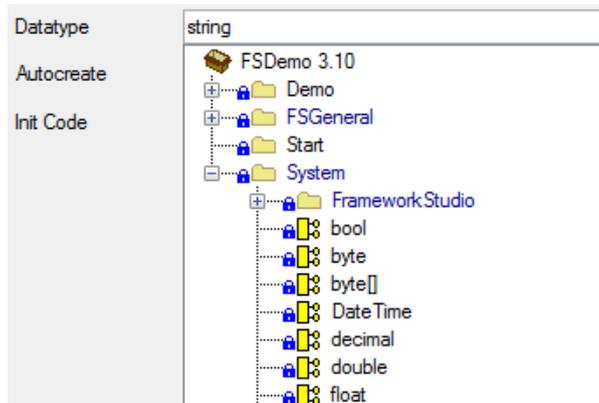
Vererbte Referenzen werden in der ListView ausgegraut und als ReadOnly dargestellt. Der Namespace aus dem die Referenz stammt wird zusätzlich angezeigt.



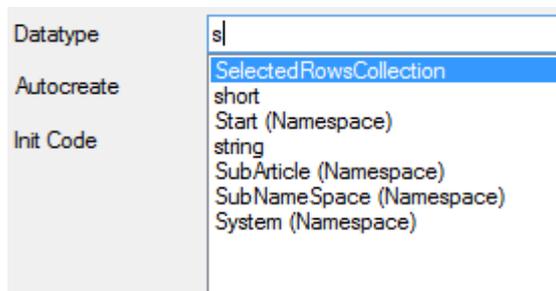
Datatype-Auswahl

Datentypen wurden in der vorherigen Version in einigen Bereichen über einen TreeView ausgewählt. Um die Auswahl zu erleichtern, wurde an diesen Stellen der TreeView entfernt und durch eine TextBox ersetzt. In dieser TextBox können während der Eingabe verfügbare Datentypen aus einer Dropdownliste ausgewählt werden.

- Auswahl über TreeView (alt)



- Auswahl über TextBox (neu)



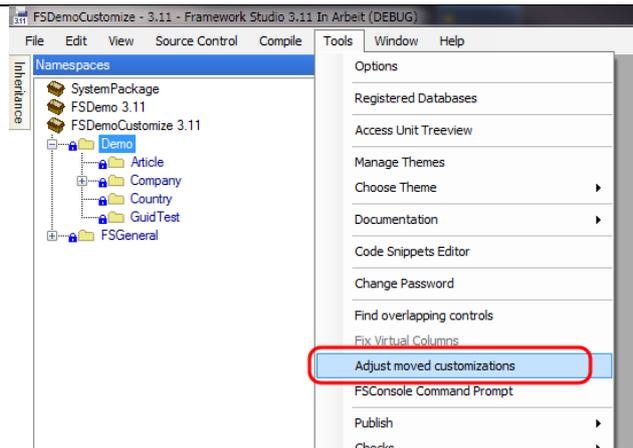
Betroffene Stellen:

- GlobalObjects
- References
- Report Document Types

Adjust moved Customizations

Wenn ein Element gecustomized wird, dann befindet sich die Customization in demselben Namespace wie das Basis-Element. Aus Performance-Gründen wird dieser Namespace an der Customization gespeichert. Wenn im Basis-Package ein Element in einen anderen Namespace verschoben wird, dann muss diese Information auch im Customizing-Package nachgezogen werden. Wird dies nicht gemacht, dann führt dies teilweise zu irreführenden Fehlermeldungen. Bisher gab es für die Korrektur im Kontextmenü des Namespaces einen Befehl „Adjust moved customizations...“

- 1.) Dieser Befehl steht jetzt im „Tools“-Menü zur Verfügung. Es werden alle Namespaces analysiert und ggf. korrigiert.
- 2.) Außerdem wird jetzt beim Compile geprüft, ob der Namespace korrekt ist. Bei Abweichungen wird ein entsprechender Compile-Error erzeugt. So kann die Ausführung dieser Routine nicht mehr vergessen werden.



Warnings

Folgende Warnings werden beim Compile nun angezeigt:

CS0105 (Stufe 3)

Fehlermeldung:

- Die using-Direktive für "Namespace" wurde bereits zuvor in diesem Namespace verwendet.
- The using directive for 'namespace' appeared previously in this namespace.

CS0162 (Stufe 2)

Fehlermeldung:

- Unerreichbarer Code wurde entdeckt.
- Unreachable code detected.

CS0168 (Stufe 3)

Fehlermeldung:

- Die Variable "Var" ist zugewiesen, ihr Wert wird aber nie verwendet.
- The variable 'var' is assigned but its value is never used.

CS0169 (Stufe 3)

Fehlermeldung:

- Das private Feld "Klassenmember" wird nie verwendet.
- The private field 'class member' is never used.

CS0219 (Stufe 3)

Fehlermeldung:

- Die Variable "Variable" ist zugewiesen, ihr Wert wird aber nie verwendet.
- The variable 'variable' is assigned but its value is never used.

CS0414 (Stufe 3)

Fehlermeldung:

- Das private Feld "Feld" wurde zugewiesen, aber der Wert wird nie verwendet.
- The private field 'field' is assigned but its value is never used.

Steuerelemente

Grid – Neues Event OnRowSelectionChanged

Im Grid-Control wird ein neues Event OnRowSelectionChanged angeboten. Dieses reagiert immer dann, wenn sich die Selektion im Grid ändert - auch bei Deselektion von Datensätzen. Mit dem Event wird auch der Datasource SelectedRows aktualisiert.

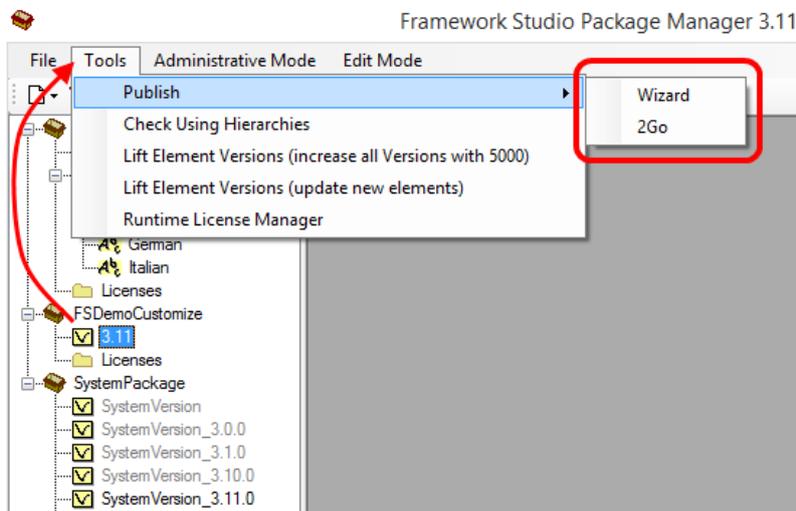
Browser – Erstellen von Snapshots

- GenerateSnapshot: Mit der Action GenerateSnapshot() am Browser-Control ist es möglich, einen Snapshot der aktuellen Ansicht zu erstellen.
- OnSnapshotGenerated: Wenn der Snapshot über die Action GenerateSnapshot() erstellt wurde, wird er über das Event OnSnapShotGenerated als byte-Array zurückgegeben.

Package Manager

Publish & Publish2Go aus dem Package Manager heraus

Der Publish bzw. Publish2Go Wizard kann jetzt auch aus dem Package Manager heraus aufgerufen werden. Dazu muss die Package Version ausgewählt werden, die gepublished werden soll. Danach kann über das Menü **Tools** -> **Publish** -> **Wizard/2Go** der entsprechende Dialog geöffnet werden.



Die Funktionalität der Wizards ist deckungsgleich mit denen der Framework Studio IDE.

Method Observer beim Import eines Service Release prüfen

Beim Import eines Service Release ist es nicht notwendig, von der importierten Package-Version abhängige Package-Versionen zu kompilieren. Dennoch ist es wünschenswert, wenn beim Import alle gesetzten Method Observer in den abhängigen Package Versionen überprüft werden, um Änderungen an Methoden im Service Release mitzubekommen und ggf. darauf reagieren zu können.

Bei der Überprüfung werden ausschließlich diejenigen Package Versionen betrachtet, die einen Owner Key besitzen bzw. **NICHT** importiert wurden.

Beispiel anhand eines Repositories mit 4 Packages:

1. FSDemo 3.11 (importiert)
2. FSDemoCustomize 3.11 (importiert)
3. One 3.11 (Owner)
4. Two 3.11 (Owner)

Wird nun im Package Manager für die Package Version „FSDemo 3.11“ ein Service Release importiert, werden alle Method Observer der Packages „One 3.11“ und „Two 3.11“ geprüft. „FSDemoCustomize 3.11“ wird nicht geprüft, da es sich um eine importierte Package-Version handelt und somit an dieser Package-Version sowieso keine Fehler korrigiert werden können (außer durch den Import eines Service Release).

Schlägt eine Prüfung eines Method Observers fehl, wird eine entsprechende Warnung in das Output-Fenster geschrieben:

```
Framework Studio - Import Service Release

Package File C:\2015-04-10_FSDemo_3.11.srbb
Start Import Close

Update Table tblRep_SVCHostEndPointBinding 0 records
Update Table tblRep_Service 1 records
Update Table tblRep_ServiceHost 1 records
Update Table tblRep_ServiceHostEndPoint 1 records
Update Table tblRep_ServiceHostMethod 1 records
Update Table tblRep_ServiceImplContract 0 records
Update Table tblRep_ServiceMethod 1 records
Update Table tblRep_ServiceMethodFault 0 records
Update Table tblRep_ServiceProxy 1 records
Update Table tblRep_ServiceProxyMethod 1 records
Update Table tblRep_VirtualColumn 0 records
Update Table tblRep_WorkFlowElement 1 records
Update Table tblRep_WorkFlowEvent 0 records
Update Table weRep_Assembly 14 records
Update Table weRep_SavedCode 8 records

Enable trigger and constraints... done.
Transaction committed.
Refresh compile stamps... done.

Checking Method Observers...
Dependent package versions:
- One - 3.11
- Two - 3.11

Checking package version 'One - 3.11'...
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.cdTest.CompTestMethod' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.cdTest.sTestProp.get_sTestProp' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.cdTest.sTestProp.set_sTestProp' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.frmTest.FormTestMethod' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.dcTest.sContractTestProp.get_sContractTestProp' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.dcTest.sContractTestProp.set_sContractTestProp' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.hostTest.HostTestMethod' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.proxyTest.ProxyTestMethod' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.trTest.Method' have been changed.

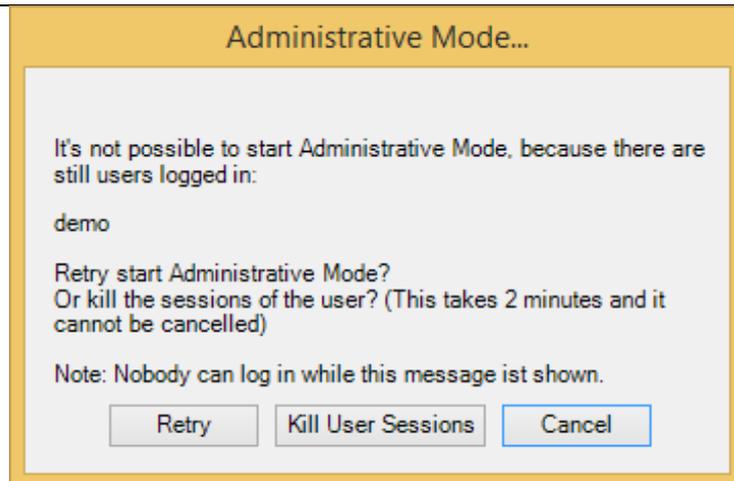
Checking package version 'Two - 3.11'...
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.cdTest.CompTestMethod' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.cdTest.sTestProp.get_sTestProp' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.cdTest.sTestProp.set_sTestProp' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.frmTest.FormTestMethod' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.dcTest.sContractTestProp.get_sContractTestProp' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.dcTest.sContractTestProp.set_sContractTestProp' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.hostTest.HostTestMethod' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.proxyTest.ProxyTestMethod' have been changed.
#warning FSWarn[2029] Method Observer: One or more base methods of 'Demo.Observer.trTest.Method' have been changed.

***** Import completed *****
```

Administrative Mode – Benutzersitzungen beenden

Wenn der Admin-Mode gestartet wird, während noch Benutzer am Repository angemeldet sind, können diese Benutzersitzungen beendet werden.

Über einen Dialog, analog wie beim Locken eines Packages, können die Sitzungen beendet werden. Die angemeldeten Benutzer haben beim Beenden der Sitzungen zwei Minuten Zeit alle Änderungen zu sichern. Nach Ablauf dieser Zeit wird FS mit einer unhandled Exception beendet. Danach befindet sich der PackageManager im Admin-Mode.



Auslagerung historischer Daten

Bereits mit **Framework Studio 3.8** wurde ein Mechanismus zur Auslagerung historischer Daten eingeführt. Das ursprüngliche Konzept wurde zu **Framework Studio 3.10** abgewandelt, da unter Oracle mit damaligen Mitteln nicht zu lösende Probleme aufgetreten sind. Wesentlicher Unterschied: Im Historienmodus werden nur alte (also historisierte) Datensätze angezeigt, keine aktuellen Datensätze.

Durch eine grundlegende Neuerung in **Framework Studio 3.11** (FSRowID) ist es nun möglich, das ursprünglich gewünschte Verhalten zu erreichen:

Im Historienmodus werden sowohl historisierte Datensätze als auch aktuelle Datensätze angezeigt.

Dazu wird auf der Datenbank (wie im ursprünglichen Konzept vorgesehen) eine Tabelle für die historisierten Datensätze und ein View, welcher die Datensätze aus der Originaltabelle mit denen aus der Historientabelle vereinigt, verwendet.

Weitere Details sind der Technischen Dokumentation zu entnehmen.

Systemvoraussetzungen

Entwicklungs-Rechner

Ein Rechner auf dem mit der Framework Studio IDE entwickelt wird bzw. ein Rechner oder Server auf dem der Framework-Compiler ausgeführt wird.

Betriebssystem

- Windows 10 64 Bit (Pro / Enterprise) ^[2]
- Windows 10 (Pro / Enterprise) ^{[1][2]}
- Windows 8.1 64Bit (Pro / Enterprise) ^[2]
- Windows 8.1 (Pro / Enterprise) ^{[1][2]}
- Windows 8 64Bit (Pro / Enterprise) ^[2]
- Windows 8 (Pro / Enterprise) ^{[1][2]}
- Windows 7 SP1 64Bit (Professional / Enterprise / Ultimate)
- Windows 7 SP1 (Professional / Enterprise / Ultimate) ^[1]
- Windows Vista SP2 64Bit (Business / Enterprise / Ultimate)
- Windows Vista SP2 (Business / Enterprise / Ultimate) ^[1]
- Windows Server 2012 R2 64Bit (Essentials / Enterprise / Datacenter)

- Windows Server 2012 64Bit (Essentials / Enterprise / Datacenter)
- Windows Server 2008 R2 SP1 64Bit (Standard / Enterprise / Datacenter)
- Windows Server 2008 SP2 64Bit (Standard / Enterprise / Datacenter)
- Windows Server 2008 SP2 (Standard / Enterprise / Datacenter) ^[1]

.NET Framework

- Microsoft .NET Framework 4.5 / 4.5.1 / 4.5.2 / 4.6 ^[2]

^[1] Es wird ein 64Bit-Betriebssystem empfohlen.

^[2] Ab Windows 8 / Windows Server 2012 ist das .NET Framework 4.5 ab Windows 8.1 / Windows Server 2012 R2 ist das .NET Framework 4.5.1 und ab Windows 10 ist das .NET Framework 4.6 standardmäßig installiert.

Weitere Software:

- Java-Runtime (siehe unten)
- Crystal Reports Runtime (siehe unten)
- Crystal Reports Designer (siehe unten)
- Oracle-Client (siehe unten)
- Visual Studio: empfohlen zum Debuggen von Anwendungen
 - Visual Studio 2012
 - Visual Studio 2010
 - Andere Versionen und Editionen können auf eigene Verantwortung in den IDE-Options konfiguriert werden.
- Windows Azure: Für die Funktion Publish2Azure wird folgende Software benötigt:
 - Windows Azure SDK Version 2.4

Anwendungs-Server

Rechner, auf dem Broker oder mit Framework Studio veröffentlichte Services laufen.

Betriebssystem

- Windows Server 2012 64Bit (Essentials / Enterprise / Datacenter)
- Windows Server 2012 R2 64Bit (Essentials / Enterprise / Datacenter)
- Windows Server 2008 R2 SP1 64Bit (Standard / Enterprise / Datacenter)
- Windows Server 2008 SP2 64Bit (Standard / Enterprise / Datacenter)
- Windows Server 2008 SP2 (Standard / Enterprise / Datacenter) ^[1]
- Client-Betriebssysteme: Ein Broker oder Service kann prinzipiell auf einem Client-Betriebssystem (Windows 7, Windows 8, Windows 8.1, Windows 10) installiert werden. In einer Produktiv-Umgebung wird aber dringend der Einsatz eines o.g. Server-Betriebssystems empfohlen. Auf Client-Betriebssystemen können nur sehr wenige Benutzer parallel arbeiten.

.NET Framework

- Microsoft .NET Framework 4.5 / 4.5.1 / 4.5.2 / 4.6

^[1] Es wird ein 64Bit-Betriebssystem empfohlen.

Weitere Software:

- Crystal Reports Runtime (siehe unten)
- IIS
- Oracle-Client (siehe unten)

Print-Server

Der Print-Service benötigt eine Verbindung zur Repository-Datenbank und er muss die gewünschten Drucker ansprechen können. Er wird nicht von der Anwendung direkt kontaktiert, deshalb kann auch ein Client-Betriebssystem verwendet werden.

Print-Service

- Framework Studio Print-Service 3.11
- Framework Studio Print-Service 3.10.3
- Framework Studio Print-Service 3.9.10
- Framework Studio Print-Service 3.8.14

Betriebssystem

- (siehe Entwicklungs-Rechner)

Weitere Software

- Crystal Reports Runtime (siehe unten)
- .NET Framework 4.5 / 4.5.1 / 4.5.2 / 4.6
- Oracle-Client (siehe unten)

Authentication-Server

Authentication-Service

- Framework Studio Authentication-Service 3.9
- Framework Studio Authentication-Service 3.9.1 (unterstützt Runtime Supervisor)
- Framework Studio Authentication-Service 3.8 oder 3.8.5

Betriebssystem

- (siehe Anwendungs-Server)

Weitere Software

- .NET Framework 4.5 / 4.5.1 / 4.5.2 / 4.6
- Oracle-Client (siehe unten)

Client-Rechner

Rechner, auf dem die Anwendung mit dem Java-Client ausgeführt wird.

Betriebssystem

An das Betriebssystem werden grundsätzlich keine besonderen Anforderungen gestellt. Entscheidend ist, dass die Java-Runtime verfügbar ist. Je nach Framework-Studio-Anwendung kann es ggf. weitere Einschränkungen geben.

- Windows 10
- Windows 8.1
- Windows 8
- Windows 7 SP1
- Windows Vista SP2
- Windows Server 2012
- Windows Server 2008
- Ubuntu Linux

- Mac OS X 10.9

Weitere Software:

- Java-Runtime (siehe unten)
- Crystal Reports Designer (siehe unten) - sofern Reports bearbeitet werden sollen
- Je nach Framework-Studio-Anwendung ggf. weitere Software

Java-Runtime

Die Java Laufzeit-Umgebung wird benötigt am Client-Rechner und auf dem Entwicklungs-Rechner zur Ausführung der Anwendung.

- Java 8
- Java 7

Crystal Reports Runtime

Diese Runtime wird benötigt auf dem Anwendungs-Server, dem Print-Service und dem Entwicklungs-Rechner.

- SAP Crystal Reports Runtime engine for .NET Framework 13.0.9 oder höher
- Bei 64Bit-Betriebssystemen muss die 64Bit-Version der Runtime verwendet werden

Crystal Reports Designer

Der Crystal Reports Designer wird für die Bearbeitung der Crystal Reports auf dem Entwicklungs-Rechner benötigt und sofern erforderlich auch auf dem Client-Rechner.

- Crystal Reports 2008

Datenbank-Server

- Microsoft SQL Server 2014 (Standard / Enterprise / BI)
- Microsoft SQL Server 2012 SP1 (Standard / Enterprise / BI)
- Microsoft SQL Server 2008 R2 SP2 (Standard / Enterprise / Datacenter)
- Microsoft SQL Server 2008 SP3 (Standard / Enterprise / Datacenter)
Die Express- und Developer-Editionen können zu Entwicklungs- und Test-Zwecken eingesetzt werden. Dabei sind aber entsprechende funktionale und lizenzrechtliche Einschränkungen seitens Microsoft zu beachten. Der Einsatz erfolgt auf eigene Verantwortung.
- Oracle 12c (Standard / Enterprise)
- Oracle 11gR2 (Standard / Enterprise)
- Oracle 11g (Standard / Enterprise)

Oracle-Client

Wenn eine Oracle-Datenbank als Repository, Runtime-Repository oder Daten-Datenbank eingesetzt wird, dann muss ein Oracle-Client installiert werden.

- Oracle Client 12.1.0.1+ mit Oracle Data Provider for .NET 4
- Oracle Client 11.2.0.3+ mit Oracle Data Provider for .NET 4
Die Version des Oracle Client sollte möglichst zur Version des Datenbank-Servers passen, ansonsten kann es zu Performanceeinbußen kommen.

Korrigierte Fehler

Neben den Fehlern, die bereits bis Framework Studio 3.10.5 korrigiert wurden, wurden in Framework Studio 3.11 die folgenden Fehler korrigiert.

ID	Text
2015109400	JavaClient: Wenn einer Combobox als ValueMamber ein Property zugewiesen wird, welches einen anderen Typ als der Datasource besitzt, sollte im Java-Client eine aussagekräftigere Fehlermeldung angezeigt werden.
2015077836	JavaClient: Ein Wert eines Controls wird nicht an den Broker gesendet, wenn dieser Wert geändert und danach wieder auf den Originalzustand gesetzt wird.
2015098790	Java-Client: Controls in einem Stack-Panel lassen sich nicht per Action sichtbar schalten, wenn sie im Designer unsichtbar geschaltet wurden.
2015078055	Broker: Wunsch: Die AccessUnitHierarchy soll für Form-AccessUnits den FullName des Forms enthalten.
2015099023	Broker: HasDbChanged() funktioniert bei ReadOnly-Components nicht korrekt und gibt immer "false" zurück.
2015110242	Broker: Wird die Anwendung mit FrameworkCloseMode.KeepUserLogin neu gestartet, dann werden die Benutzerdaten nicht in die neue Session übernommen (es wird z.B. bei einem Mandanten-Wechsel im eNventa ERP der Login-Dialog angezeigt).
2015120382	Framework: Wird ein Control mit der Methode HidePermanent im Form ausgeblendet, dann bleibt das zugehörige Label sichtbar. <i>Jetzt wird das Label mit ausgeblendet - es muss sich dafür aber in demselben Container/Panel befinden wie das Control.</i>
2015057117	Framework - Wunsch: Am "IFrameworkControlActionBase" sollte es eine Eigenschaft "Form" geben, damit z.B. beim Drag&Drop vom Target- oder SourceControl aus das zugehörige Form bzw. die FormVariant ermittelt werden kann.
2015120375	Package-Manager: Bevor der Compile-Wizard aufgerufen wird, müssen alle Änderungen - insbesondere an der Hierarchie - gespeichert werden. Bei Aufruf über den Compile-Wizard-Button im Versions-Dialog erfolgt diese Prüfung nicht. In der Folge berücksichtigt der Compile-Wizard die Änderungen nicht.
2015046605	IDE: Bei der Suche nach Overlapping Controls in einem Custompackage kann es zu Exceptions und falschen Suchergebnissen kommen.
2015088424	IDE: Namespace-Window synchronisiert sich, obwohl die Option "Track active item" deaktiviert ist.
2015088591	IDE: Ist Intellisense geöffnet, wird in seltenen Fällen eine Exception geworfen, wenn der Anwender mit [STRG] + [Z] die Änderung rückgängig macht.

2015088687	IDE: Methodenüberladungen im Intellisense werden in der falschen Richtung durchgeblättert.
2015088690	IDE: Methoden mit mehreren Überladungen werden im Intellisense nicht nach der Parameteranzahl sortiert.
2015088726	IDE: Das Intellisense-Fenster für Methodenüberladungen wird von der Größe her falsch berechnet. Der Inhalt wird immer umgebrochen.
2015098794	IDE: Die Intellisense ListBox, der Tooltip rechts daneben und der Tooltip für Methodenüberladungen verschwinden unter anderen Forms/Panels und sind durch einen Mausklick fokussierbar.
2015099097	IDE: Nach dem Schließen des Documentation Chapter Trees wird nicht das zuletzt im Chapter Tree geöffnete Element angezeigt.
2015110261	IDE: Intellisenseliste wird geschlossen, sobald mit KeyUp/KeyDown oder der Maus ein anderer Eintrag ausgewählt wird.
2015120276	IDE: Wenn FS nicht auf der primären Anzeige offen ist und es wird der Intellisense-Tooltip für Methodenüberladungen geöffnet und mit Up/Down ein anderer Eintrag ausgewählt, erscheint der Tooltip plötzlich auf der primären Anzeige anstatt auf derjenigen, auf der FS geöffnet ist.
2015120299	IDE: Das Öffnen der Element History für eine grafische Transformation führt zu einer System.NullReferenceException.
2015109511	IDE: Weil für das SystemPackage im Repository keine Assembly-Infos abgelegt sind, findet die Referenzermittlung über das Laden der Assembly in einer AppDomain statt. Dieser Mechanismus versagt aber manchmal, weil z.B. die gerade kompilierten Interfaces nicht zur mehr System-Package Assembly passen.
2015109478	IDE: Wenn man in einem Custom Package nach dem Text "override" sucht, kann es zu folgender Fehlermeldung kommen: Search.findRegExpInGeneratedCode: Exception occured. A search result will be dropped.
2015110035	Customizing: Werden im Basis-Package an einer Component private oder internal Properties angelegt und wird diese Component dann gecustomized, dann führt dies in der Application zur Laufzeit zu Fehlermeldungen.
2015110273	Print-Server: Im Crystal-Adapter kommt es in seltenen Fällen zu einer InvalidPrinterException, wenn der Print-Service nicht als Console sondern als Dienst ausgeführt wird.
2015109285	AuthenticationService: Wird die AuthenticationService.exe gestartet, so werden keine "Admin-Rechte" verlangt. Dies kann dazu führen, dass der Service nicht erreichbar ist.