

Neue Wege in der Softwareentwicklung

Individuelllösung kontra Standard- software

Die Softwareentwicklung hat sich in den letzten Jahren rapide verändert. Immer neuere Technologien und immer schnellere Produktzyklen bestimmen den Markt. Die Anforderungen an die Funktionalität von Softwarelösungen seitens der Anwender sind dabei auch ständig gestiegen. Die Lösungen müssen flexibel, kostengünstig und schnell verfügbar sein. Doch gerade diese Umstände machen es den IT-Verantwortlichen nicht gerade leicht Entscheidungen zwischen Eigenentwicklungen und Standardlösungen zu treffen. Standardlösungen sind relativ kostengünstig und schnell verfügbar, haben aber oft nicht die nötige Flexibilität und verursachen hohe Integrationskosten. Eigenentwicklungen beinhalten hohe Risiken, verursachen hohe Kosten und sind selten termingerecht verfügbar.

Neue Ansätze mit visionXMLCube

Vision Consulting AG beschäftigt sich seit 1989 mit der Thematik der Applikationsentwicklung. Wir haben unzählige Projekte im Datenbank und CRM-Umfeld durchgeführt. Dabei sind immer wieder die gleichen Aufgabenstellungen und Problematiken entstanden. Unsere Standardlösung musste individuell an die Bedürfnisse der jeweiligen Kunden und Branchen angepasst werden. Dies führte zu einer hohen Anzahl von

Schnittstellen und Applikationsteilen, welche auf die Dauer zu sehr hohen Wartungskosten und Qualitätsproblemen führten.

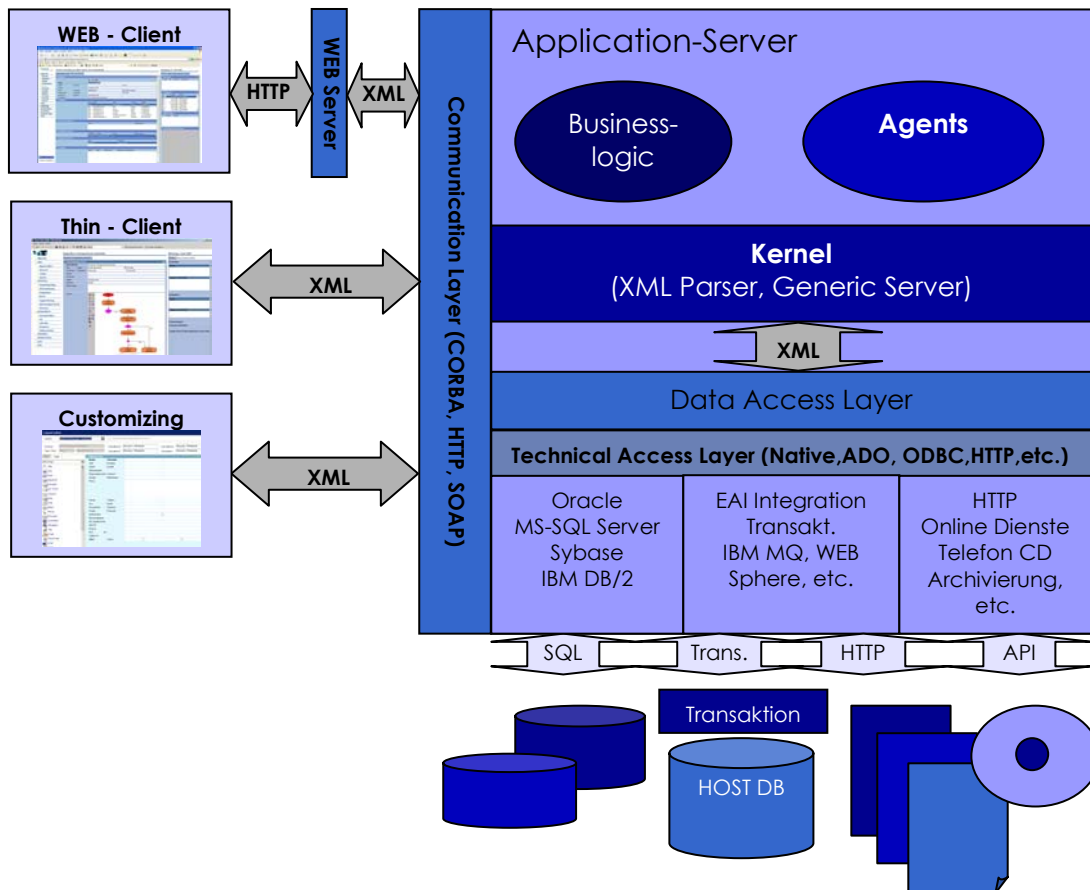
Als im Jahre 2001 wieder einmal das Ende eines Produktzyklus anstand, haben wir uns entschlossen komplett neue Wege zu gehen um der bisher beschriebenen Problematik aus dem Wege zu gehen. Bis anhin wurde für den Aufbau einer neuen Applikation jeweils eine Anforderungsanalyse durchgeführt, die hauptsächlich durch die Anforderungen der zukünftigen Anwender und deren Prozesse bestimmt wurde. Nebenbei sind dann jeweils noch einige technische Anforderungen (z.B. WEB fähig, Oracle Datenbank) eingeflossen. Dies führt zu einer sehr hohen Konzentration auf die zukünftige Anwendungsfälle und lässt dabei oft die Architektur der zukünftigen Lösung nicht im Mittelpunkt stehen.

Unser Ansatz war dabei die Spezifikation einer Applikationsarchitektur und nicht die Spezifikation einer Anwendung.

Dabei standen folgende Themen im Vordergrund:

- Internet Fähigkeit/ Thin-Client
- Verteilte Applikation / Skalierbarkeit
- Einfache Installation und Softwareverteilung
- Flexibles Datenmodell
- Freie Gestaltung von Layouts/Reports und Regeln
- Datenbankenabhängigkeit
- XML - Basiert
- Objektorientierte Technologien
- Offene Schnittstellen
- Komponentenbasiert
- Mehrsprachigkeit

Applikationsarchitektur



Technologien die heute aktuell sind, sind morgen vielleicht schon wieder überholt. Daher wurde das Schwergewicht beim Design der Architektur auf die Modularität und die Skalierbarkeit gelegt. Dabei wurden folgende grundlegende Applikationsschichten definiert:

- Web- und Thin-Client (Komponentenarchitektur)
- Kommunikation Client/Server (HTTP, CORBA, SOAP, etc.)
- Schnittstellen zu Datenbanken und externen Datenquellen
- Customizing Werkzeuge (Konfiguration)
- XML als Datenaustausch- und Objektbeschreibungssformat
- Agentenarchitektur und Businesslogik

Die Datenbeschreibungssprache **XML** hat sich in den letzten Jahren als Standard für Datenbeschreibungen durchgesetzt und etabliert. VisionXMLCube verwendet für alle Beschreibungen von Strukturen, Relationen, Layouts, Regeln und Benutzermenüs die Beschreibungssprache **XML**. Auch die Übertragung von Daten zwischen dem Server und dem Client wird mit Hilfe von **XML** durchgeführt.

Das Konzept der Komponentenarchitektur

Liest man Pflichtenhefter und Anforderungskataloge, trifft man immer wieder auf die selben Anforderungen. So soll zum Beispiel die Office-Suite von Microsoft eingebunden werden. Eine Statistik soll in Excel ausgegeben werden, die Telefonanlage soll integriert sein und Emails sollen direkt archiviert werden. Diese Liste könnte man nun beliebig weiterführen. Aber eines fällt auf, es sind immer die gleichen Themen und Anforderungen in unterschiedlichen Zusammenhängen.

Firma		Name	Präfix	Nummer
Zusatz	Suchwert	Gerodetti Marco	Tel. direkt	01 389 80 85
Abteilung	Kunden Nr.	Gerodetti Marco	Tel. privat	071 2772247
Postfach		Hässig Thomas	Handy	079 421 18 52
Strasse	Forchstrasse 239	Hässig Thomas	Tel. direkt	01 360 20 76
Lan	PLZ	Dit	Kar	CH
Briefanrede	Zuständig	Sehr geehrte Damen und Herren		
Email	Homepage	info@vision.ch		http://www.vision.ch
Telefon	Fax	01 389 80 80		01 389 80 90
Sprache	Partner seit	Deutsch		08.04.2003

Dies hat uns dazu bewogen eine ausgeklügelte Komponentenarchitektur zu entwerfen und in die Customizing-Umgebung zu integrieren. Dies garantiert, dass einmal erstellte Komponenten jederzeit **wiederverwendet** werden können. Nehmen wir als Beispiel die TAPI-Schnittstelle. Diese Komponente könnte in allen Bereichen einer Applikation zum Einsatz kommen.

Der Schlüssel zum Erfolg der Komponenten-Architektur liegt darin, dass die Komponenten möglichst allgemeingültig einsetzbar sind. Die im Beispiel aufgezeigte TAPI Komponente beinhaltet einen Mechanismus, der in beliebigen Bildschirmmasken und Relationen nach Telefonnummern suchen kann, diese in

Damit ist diese Komponente nun universell in allen Bildschirmen einsetzbar und kann jederzeit **wiederverwendet** werden.

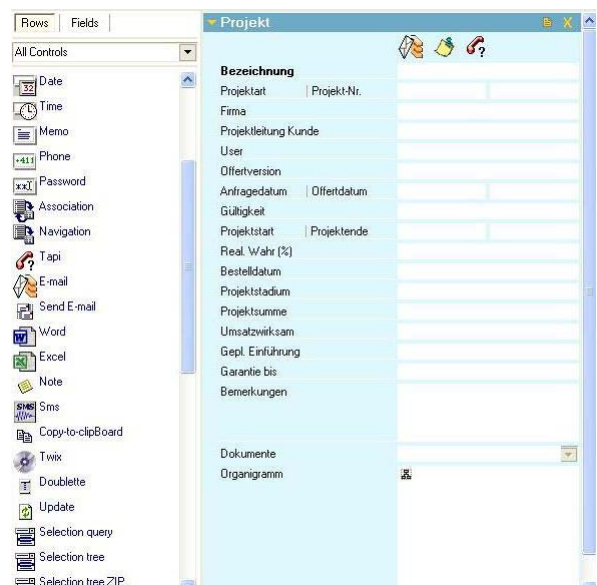
Die Komponente kann danach im visionXMLCube per „drag und drop“ verwendet werden.

Weitere Beispiele für solche Komponenten sind:

- E-Mail-, Word- und Excel integration
- Twix Integration
- Dokumentenarchivierung
- Postleitzahlen Auswahl
- Dublettencheck
- Bilderanzeige
- Datums- und Zeitfelder
- Passwortfelder, Checkboxen
- Auswahllisten, Memofelder

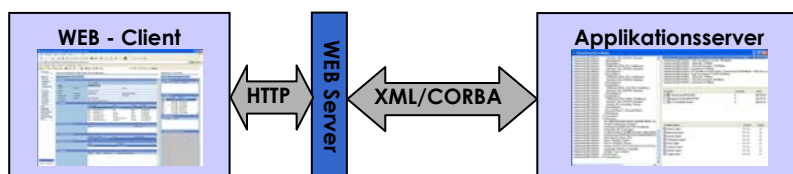
- Diverse Spezialkomponenten für Prozessdefinitionen, Prozesssteuerungen, Kampagnenmanagement, Segmentierung, Reporting, Statistik, Portfolioanalysen, Sales Funnel, Query Tools, etc.

einer Liste anzeigt und die nötigen TAPI - Funktionen in sich kapselt.



Kommunikation Client/Server

Die Applikationsschicht für die Kommunikation von Client zum Server ist sehr offen gestaltet. Da ausschliesslich XML zwischen Client und Applikationsserver übertragen wird, kann das Übertragungsprotokoll relativ frei gewählt werden. Im aktuellen Release verwenden wir zwei austauschbare Technologien. Zum einen wird „http“ oder „https“ verwendet und die XML-Dateien werden im 7Bit ASCII codierten ZIP Format übertragen. Diese Übertragungsart erfordert einen WEB Server (IIS oder Apache). Als zweite Variante wurde die CORBA Technologie implementiert. Mit CORBA können XML-Objekte transportiert werden. Die CORBA Technologie ist besser geeignet für den Einsatz in firmeneigenen Netzwerken, die http – Technologie ist für den Einsatz im WWW vorgesehen. In gemischten Umfeldern können auch beide Technologien eingesetzt werden. Da die Kommunikationssicht austauschbar ist, können in Zukunft auch beliebige andere Kommunikationstechnologien eingesetzt werden wie zum Beispiel SOAP.



Schnittstellen zu Datenbanken und externen Datenquellen

Die Applikationsschicht die für den Zugriff auf Daten zuständig ist gliedert sich in fünf grundsätzliche Teile:

- Zugriff auf die eigene SQL Datenbank (Oracle 8.x,9.x , MS-SQL Server 2000, DB/2, Sybase)
- Zugriff auf externe SQL Datenbanken (inkl. Reverse Engineering)
- Zugriff auf Read-Only Daten (z.B. Twix CD)
- Zugriff auf Dokumente / Archivierung
- Zugriff auf Host-Daten (Transaktionen, EAI - Integration)

VisionXMLCube arbeitet normalerweise mit einer der oben genannten SQL-Datenbanken. Der Struktur Editor ist in der Lage SQL-Tabellen, Indexe , Views etc. zu verwalten und automatisch zu generieren. Mit den integrierten Werkzeugen können die Strukturen auch kontrolliert, generiert und dokumentiert werden. Da alle gelesenen Daten in ein internes XML Format transferiert werden, ist es jederzeit möglich auch andere Datenformate und Datenbanken in diese Datenbankschicht zu integrieren. Es müssen lediglich die Spezialitäten der jeweiligen Datenquelle angepasst werden. Der Zugriff auf die Daten läuft normalerweise über ADO oder ODBC.

Sind in einem Unternehmen bereits Daten in externen Datenbanken oder Datenbanktabellen vorhanden können diese 1:1 in die bestehende Umgebung integriert werden. Dabei ist auch ein Konzept für die Verwaltung externer Schlüssel vorhanden. Mit dem integrierten Reverse-Engineering-Tool können die bestehenden Tabellendefinitionen eingelesen werden.

Structure editor

Structure: TTWIX: TwixTel

Table name: TTWIX
 Class: TTWIX
 Type: External
 Read-Access: Free
 Write-Access: Free
 Primary key type: String GUID

Mandatory:
 Has list:
 Searchable:
 Has type:
 Full load:
 Used by Agent:
 Can be locked:

German
English
Italian
French
Description

Structure fields

Show Systemfields Sort fields

FieldName	Type	Length	Dec. pla...	Mandatory	Primary
FEADRESSE	V	50			
FEADRESSEROBINSON	V	50			
FPHONE	V	50			
FPHONETEXT	V	50			
FTEXTBEFOREPHONE	V	50			
MAREACODE	V	50			
MCITY	V	50			

Selbst Zugriffe auf CD-Rom, WWW-Dienste oder andere Read-only Daten sind problemlos möglich. Der Applikationsserver enthält eine generische Schnittstelle, die Integration von fremden API's ermöglicht. Dabei werden die Daten in einem beliebigen Format gelesen und in dieser Schnittstelle in das interne XML Format umgewandelt. Der Anwender kann im Front-End seiner Lösung nicht mehr unterscheiden woher die Daten gelesen wurden. Sämtliche Layouteditoren, Reports, etc. können 1:1 auch für diese externen Daten verwendet werden.



Das Framework enthält auch Komponenten zur zentralen Archivierung von Dokumenten. Diese können, auch aus dem WEB-Browser, mit Up- und Download auf den zentralen Server gespeichert werden. Die Dokument werden in einer vorkonfigurierten Verzeichnisstruktur ausserhalb der Datenbank abgelegt.

Das Framework enthält auch Komponenten zur zentralen Archivierung von Dokumenten. Diese können, auch aus dem

EAI (Enterprise Application Integration) Projekte werden in den nächsten Jahren immer wichtiger. Der visionXMLCube enthält diverse Werkzeuge zur einfachen Erstellung von Connector-Objekten. Diese Objekte werden im Application-Server eingebaut und erlauben den Datenaustausch mit Fremdapplikationen über verschiedene Protokolle (CORBA, SOAP, http). In der Regel werden die Objekte auch mit XML beschrieben und der Connector wandelt die gewünschten Objekte in das interne XML Format.

Customizing Werkzeuge (Konfiguration)

Die Werkzeuge zur flexiblen Konfiguration von Applikationen werden in Zukunft immer wichtiger, da gerade in diesem Bereich enormes Potential zur Kosteneinsparung und zur Flexibilisierung von Applikationen besteht. Die Implementierungszyklen können so auf ein Minimum reduziert werden. (time-to-market).

Beim Einsatz des Frameworks visionXMLCube kann das gesamte Datenmodell frei definiert werden. Die integrierten Werkzeuge sind in der Lage die gesamte SQL Datenbank zu generieren und auf dem aktuellen Stand zu halten. Die Dokumentation der Strukturen und Layouts wird ebenfalls automatisch generiert und ist damit immer aktuell. Mit dem Layout-Editor können beliebige Layouts zu allen Strukturen definiert werden. Im späteren Betrieb können unterschiedlichen Anwendern auch unterschiedliche Layouts zugeordnet werden.

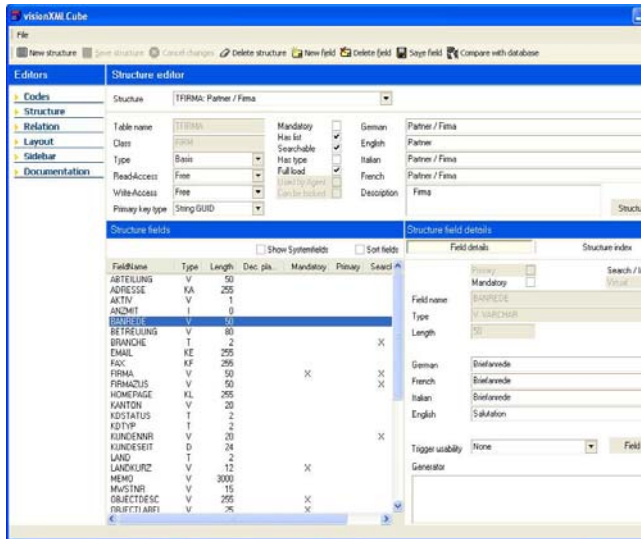
Gesamtübersicht Customizing Tools

The screenshot displays the visionXMLCube Customizing Tools interface, which is divided into several main sections:

- Strukturen/Regeln:** A panel on the left showing a tree view of database structures and a table listing various data types and their properties.
- Relationen:** A central panel showing a diagram of database relationships between tables.
- Layouts:** A panel on the right showing a visual representation of a data layout or form structure.
- Codetabellen:** A panel on the right showing a table of code definitions and their associated metadata.
- Menu-Editor (Sidebar):** A panel on the far right showing a configuration for the application's menu structure.

At the bottom left, there is a central icon representing a **SQL** database, with arrows pointing from it to the various tool panels, indicating that these tools are used to configure and generate the database structure.

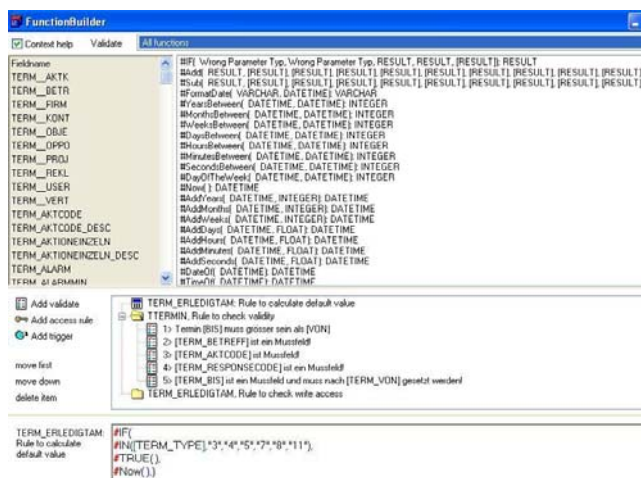
Der Struktureditor



Mit diesem Werkzeug werden die Daten-Objekte beschrieben. Dieses Werkzeug ist für folgende Aufgaben zuständig:

- Erfassen von Feldern mit Datentypen, Längen und mehrsprachigen Bezeichnern
- Definieren von Erfassungsregeln und Triggern
- Definieren von zwingenden Feldern und Index Definitionen
- Festlegung der Datenquelle und der Primär Schlüssel
- Generierung aller SQL-Skripts für die Erstellung der Datenbank (Tabellen, Index, etc.)
- Verifizierung der Datenbank, erstellen eines Differenz Skripts mit ALTER TABLE Befehlen

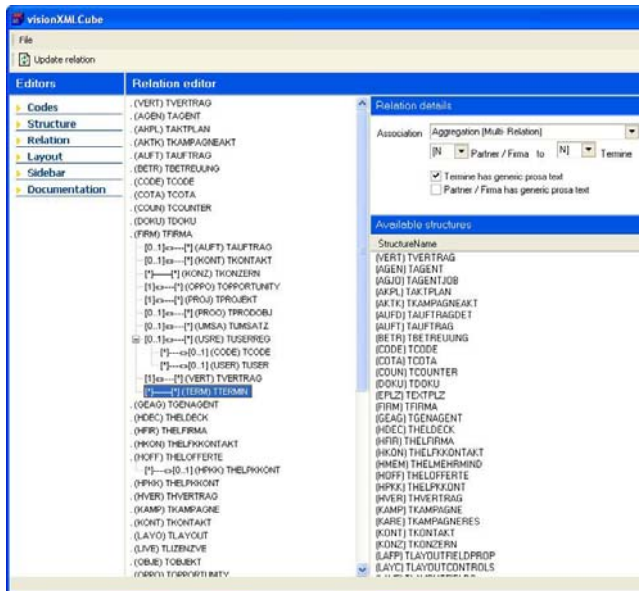
Der Struktureditor ist Datenbank unabhängig und generiert je nach Datenbanktyp die benötigten Feldtypen und Formate. Die Basis-Datentypen wurden durch Datentypen wie Email, Telefon, Dokument, Hyperlink, etc. erweitert. So ist gewährleistet, dass im ganzen Produkt die selben Feldtypen und Längen für solche Felder verwendet werden. Komponenten wie zum Beispiel die Telefonschnittstelle (TAPI) können so generisch feststellen welche Felder Telefonnummern enthalten. Eine Migration von einer Datenbank auf eine andere geht so in Minuten. Damit die Konsistenz der visionXMLCube Konfiguration jederzeit gewährleistet ist, sind einige Prüffunktionen integriert.



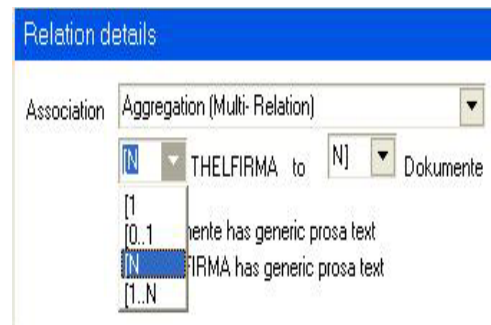
Der Regeleditor erlaubt das Definieren von Validierungsregeln, Vorgabewerten, Berechnungen und Triggeraktionen.

Es steht eine ganze Sammlung von Funktionen zur Berechnung von Feldinhalten zur Verfügung. Je nach Datentyp können beliebige Berechnungen und Formatierungen vorgenommen werden.

Der Relationen Editor

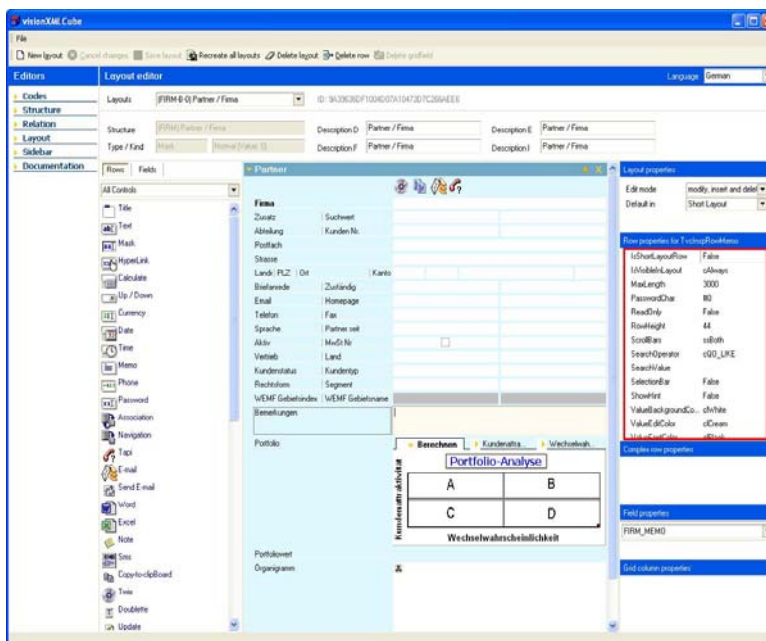


Mit diesem Werkzeug werden die Relationen der Daten-Objekte beschrieben. Das Datenmodell kann dabei frei bestimmt werden. Dabei sind alle aus Entity-Relationship Modellen bekannten Kombinationen möglich.

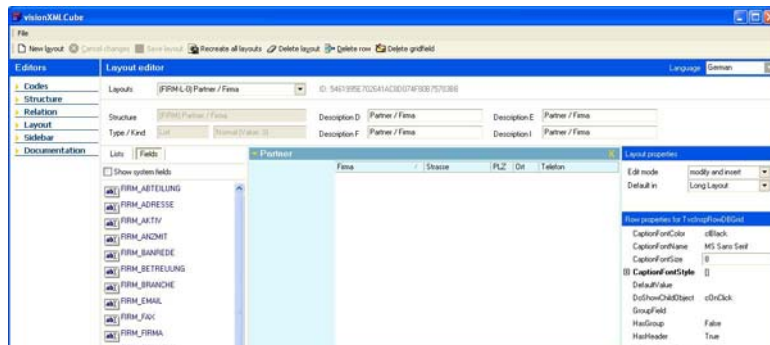


Der Layout Editor

Mit dem integrierten Layout Editor können beliebige Layouts erstellt werden. Dabei wird unterschieden ob es sich um eine Auswahlliste oder um eine Bildschirmmaske handelt. Für eine Struktur (z.B. Kunde) können beliebig viele Layouts und Listen erstellt werden. Es können auch Layouts pro Rekordart erstellt werden. So würde zum Beispiel ein Layout für einen Kunden komplett anders aussehen als für einen Lieferanten. Der Editor kann per „drag & drop“ bedient werden. Dabei stehen alle verfügbaren Komponenten und Strukturen zur Verfügung.



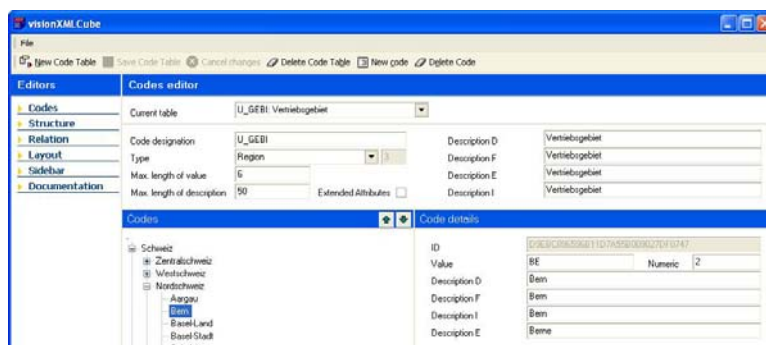
Für jedes Feld können die gewünschten Eigenschaften festgelegt werden. So kann zum Beispiel für jedes Feld festgelegt werden, ob es sich um ein Suchfeld handelt, ob es in der Kurzansicht sichtbar ist und wie es sich im Layout verhält. Die Zuordnung zu den jeweiligen Objekten und Feldnamen erfolgt ebenfalls per „drag & drop“.



Auch Auswahllisten können auf diese einfache Weise zusammengestellt werden. Dabei kann die Sortierung, die Anordnung der Spalten und die Gruppierung festgelegt werden. Die Anwender können später individuell die angezeigten Felder ein- und ausblenden.

Der Code Editor

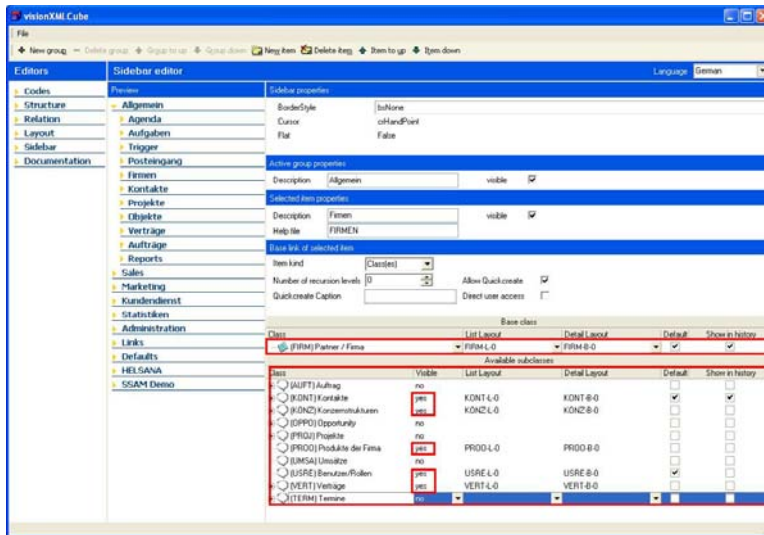
Mit dem integrierten Code Editor können Codetabellen wie Sprache, Briefanrede, Kundentyp, etc. erstellt und verwaltet werden. Diese Tabellen werden im Client zwischengespeichert. Dies erhöht die Verarbeitungsgeschwindigkeit. Änderungen, die zentral vorgenommen wurden, werden aber sofort erkannt und neu geladen. Alle Codes stehen in den Abfragemodulen auch als Selektionskriterien zur Verfügung.



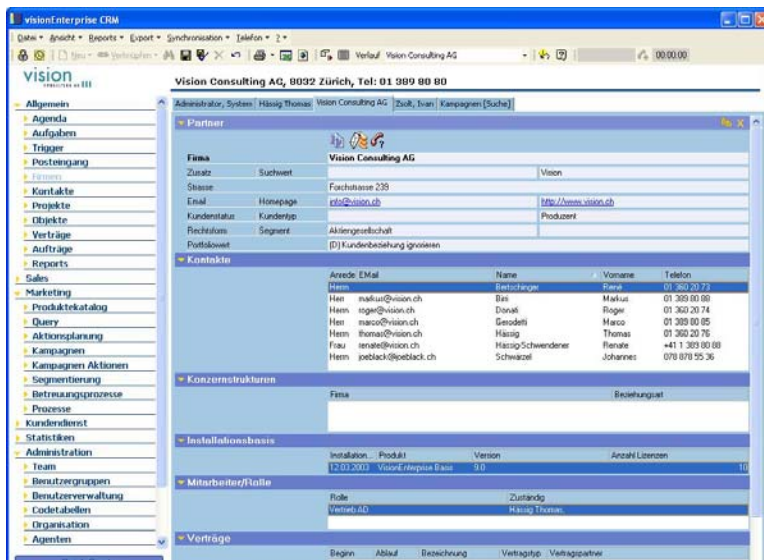
Codetabellen können auch Hierarchisch abgebildet werden. Alle Merkmale sind mehrsprachig erfassbar und werden je nach Benutzer-sprache entsprechend angezeigt.

Der Sidebar Editor

Mit dem integrierten Sidebar Editor (Menueeditor) können die vordefinierten Module und Layouts zu einer Gesamtapplikation zusammengebaut werden. Die Menustruktur muss nicht zwingend dem Datenmodell entsprechen. Objekte können beliebig ein- und ausgeblendet werden.



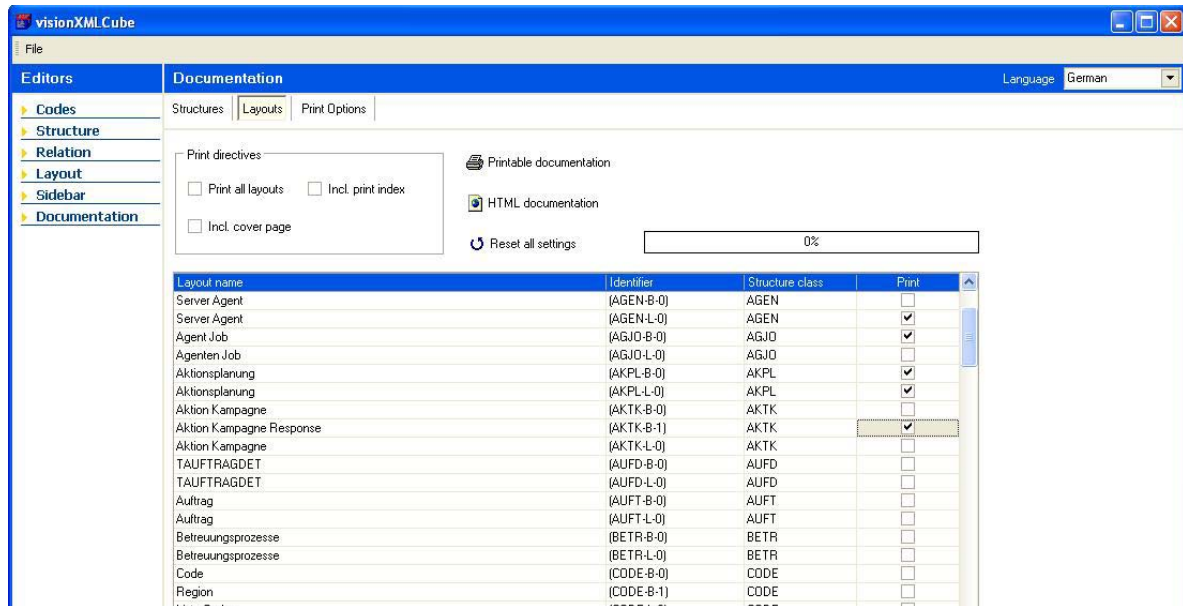
Dieses Beispiel zeigt den Menüpunkt Firma. Es werden alle gemäss Datenmodell untergeordneten Objekte angezeigt. Es kann nun frei bestimmt werden welche Objekte (z.B. Projekt, Kontakte, Umsatzzahlen, etc.) angezeigt werden und welches Layout dabei verwendet werden soll. Die so definierten Menüpunkte können später Benutzergruppen zugeordnet werden.



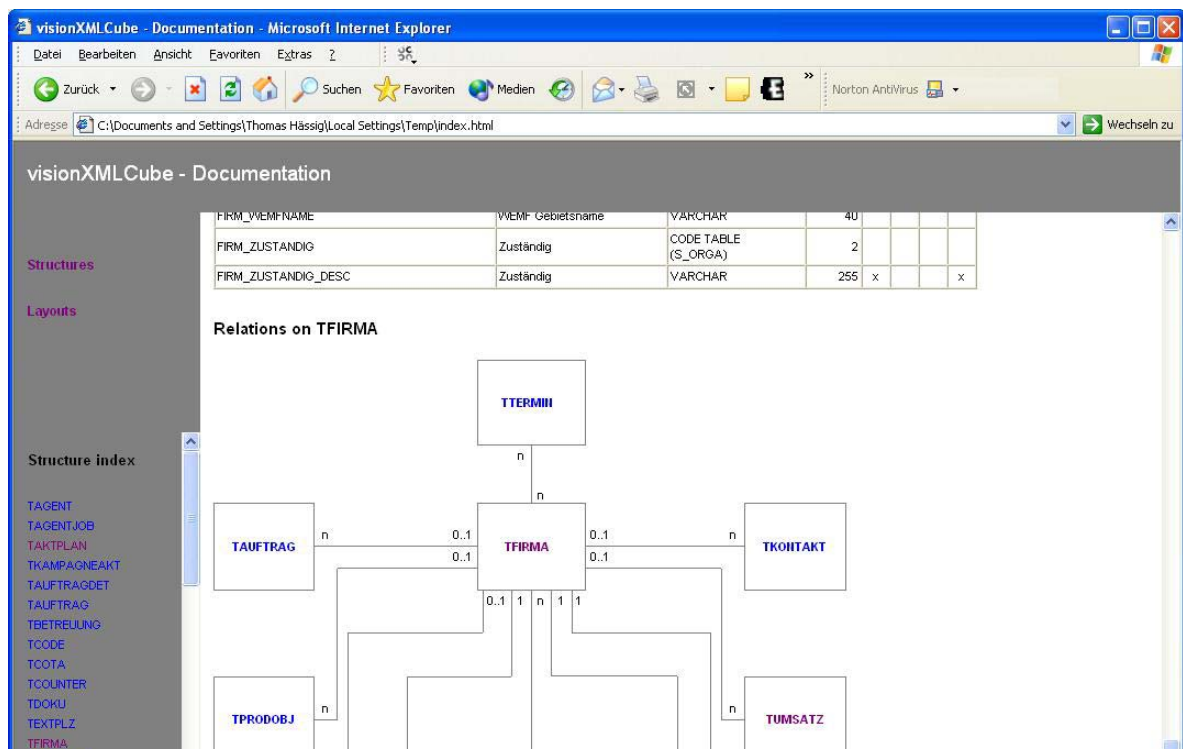
In dieser Darstellung wird das Firmenobjekt mit all seinen Beziehungen im laufenden Programm angezeigt. Dabei ist es für den Anwender nicht ersichtlich ob es sich um 1:1, 1:n oder n:n Beziehungen handelt. Der generische Client übernimmt die nötige Steuerung.

Die Dokumentation

Mit den integrierten Dokumentations-Werkzeugen kann die gesamte Konfiguration automatisch dokumentiert werden. Dies erspart viel Arbeit bei der Verwaltung und Wartung der Applikation.



Die Ausgabe erfolgt als Online-Dokumentation im HTML Format und kann auch direkt gedruckt werden.



Die Prozesssteuerung

Mittwoch, 26. März 2003

Prozesse | History | Internet | Hilfe

Prozesse

Auswahl Prozess als Tree

Prozess

Kanal: Elektronisch/EMail

Rekurrenz: Intern

Kampagne: Auswahl Kampagne

Verantwortung: Administrator System_Vision

Kunde/Kontakt: Administrator System_Vision

Internet: <http://www.vision.ch>

Kundenreklamation

```

graph TD
    Start([Kundenreklamation]) --> Step1[Entgegennahme Ticket / Eröffnung]
    Step1 --> Step2[Interne Bearbeitung der Reklamation (+3 Tage)]
    Step2 --> Decision{Fehler?}
    Decision -- beim Kunden --> Step3[Kunde besuchen persönliche Entschuldigung]
    Step3 --> Step4[Kunde informieren Entschuldigungsbrief senden (+2 Tage)]
    Decision -- intern --> Step5[Kunde informieren (+2 Tage)]
    Step4 --> End([Prozess Ende])
    Step5 --> End
  
```

Aufgabe

Reklamation erfassen

Reklamation erfassen

Kontakt: Thomas Hässig

Betreff: Entgegennahme Reklamation

Kunde/Kontakt: Huber, Mark us, Huber, Max

Verantwortung: Brunner, Karl, ...

Beginnt am: 26.03.2003

Fällig am:

Mitarbeiter/Teilnehmer: Administrator System_Vision

Aktivität: Kanal: Telefon

Erledigt: Erledigt am:

Privat: Priorität: Mittel

Erinnerung: 00:00

Bemerkung:

Internet: <http://www.vision.ch>

Links: -- 1 Unzufrieden mit der Performance

Dokumente

Moderne Softwarelösungen werden in zunehmendem Masse komplexer und überfordern oftmals die Anwender. Darum wird es immer wichtiger, dass Geschäftsprozesse durchgehend abgebildet werden können. Der Benutzer wird dabei durch vordefinierte Abläufe geführt und die Steuerung läuft vollautomatisch.

Im Prozessdiagramm wird jederzeit der aktuelle Stand des Geschäftsfalls angezeigt. Benötigt ein Prozess Zusatzinformationen wie Kontakte oder müssen Formulare ausgefüllt werden, so wird dies automatisch vorgeschlagen und gesteuert. Das System kann auch automatisch Entscheidungen fällen und die anfallenden Aufgaben, basierend auf einem Rollenkonzept, unternehmensweit verteilen.

Nehmen Sie mit uns Kontakt auf...

Wenn Sie weitere Informationen über visionXMLCube wünschen oder besuchen Sie unsere Website unter www.vision.ch
Alle Produkt- und Unternehmensnamen sind als Marken der jeweiligen Unternehmen zu betrachten.
Alle anderen Marken sind Eigentum ihrer jeweiligen Inhaber.