

40 Jahre Oracle Forms

Frank Hoffmann, Cologne Data GmbH



Oracle feiert in diesem Jahr ein beeindruckendes Doppel-Jubiläum: Vierzig Jahre Datenbanken und Forms/ Reports. Es war Bill Friend, der nach Bruce Scott fünfte Oracle-Mitarbeiter (#1004), den Larry Ellison im Jahr 1979 mit der Entwicklung des berühmten Software-Tools beauftragte. Im selben Jahr brachte Oracle dann noch vor Weihnachten neben der ersten kommerziellen Datenbank (Version 2.3) auch die erste Forms- und Reports-Version unter der Bezeichnung „IAF/RPT“ heraus.

Mercedes wirbt für seine G-Klasse aus dem Jahre 1979 mit dem Claim „Stronger than time“ und das ist auch für Oracle Forms passend. Die Software-Evolution läuft bereits seit vierzig Jahren; der Modulcode von ASCII-Text bis XML und der Support für Java-Versionen von 1 bis 11, der Datenbank-Support für die Versionen 2.3 bis 19, der Windows-Support von 3.1 bis 2016 sowie der Linux-, Solaris- und AIX-Support ebenfalls. Immer ist das Tool mitgegangen, ohne dass der Anwender seinen Code umschreiben musste – ein starkes Plus. Darüber hinaus schaffen 4GL-Standardfunktionen wie Row Locking, Binding sowie die PL/SQL-Nutzung der aktuellen Datenbank enorme Vorteile gegenüber allen anderen PL/SQL-Werkzeugen (siehe Abbildung 1).

Es ist an der Zeit, einmal zu den Forms-Ursprüngen zu gehen, um die Gründe zur Entstehung dieses Tools verstehen zu können. Über Hinweise von Mike Ferrante und soziale Medien ist der Autor auf Bill Friend gestoßen, den Entwickler der Ursprungsversion, und durfte ihn intensiv dazu befragen. Mit seinen sechzig Jahren ist er immer noch sehr vital und kann sich an viele Details aus den Anfängen von

Forms erinnern. Die folgende Geschichte und das Interview stammen direkt aus den Informationen von Bill Friend und einem Gespräch Ende Dezember 2018. Er hat damit nach vierzig Jahren sein erstes Interview zu Oracle Forms gegeben.

Forms erblickt das Licht der Welt

Das Jahr 1979 hat viele wirklich interessante und spannende Neuerscheinungen hervorgebracht. Quincy Jones verhalf mit „Off the Wall“ Michael Jackson zu einem Durchbruch und Mercedes entwickelte das erste Modell der G-Klasse. Namco brachte in Japan mit „Galaxy“ die erste farbige Arcade-Fassung heraus sowie danach die berühmten Nachfolger „Galaga“ und „Pac Man“ – noch lange bevor Heimcomputer ihren Siegeslauf antraten. Aber auch wichtige technische Neuerungen fielen in dieses Jahr wie zum Beispiel „TCP/IP“, die Programmiersprache „C“, der Verkauf der Prozessoren Intel 8088 und Motorola 68000, alles Voraussetzungen für zukünftige Entwicklungen von Microsoft, Apple und Oracle sowie der Verbreitung des Internets.

In dieser Zeit suchte ein junger Entwickler einen neuen Job. Sein Name ist Bill Friend und er machte das so, wie man in dieser Zeit üblicherweise einen neuen Job suchte: Man nahm die Gelben Seiten zur Hand und suchte unter „Data Processing“ nach geeigneten Firmen, um dann unter „S“ auf „Software Development Labs“ zu stoßen, einer Firma mit drei Gründern und einem talentierten angestellten Entwickler namens „Bruce Scott“. Beim ersten Anruf ging Bob Miner an das Telefon. Bei der Frage nach einem Job kam folgende Antwort: „We’re creating the first commercial relational database management system, we’re doing it on DEC PDP minicomputers in assembly language, but we want to rewrite it in C for portability. We need someone who can write some tools for the thing.“ Sie suchten also nach einem Entwickler, der für die neue relationale Datenbank Werkzeuge in der neuen Programmiersprache C entwickeln konnte.

Bill fuhr zum Vorstellungsgespräch nach Sand Hill Circle in Menlo Park und wurde zum Mittagessen mit Bob Minor und Larry Ellison eingeladen. Nachdem Bob Minor mit UFI (User Friendly Interface, einem SQL-Plus-Vorläufer) ein paar

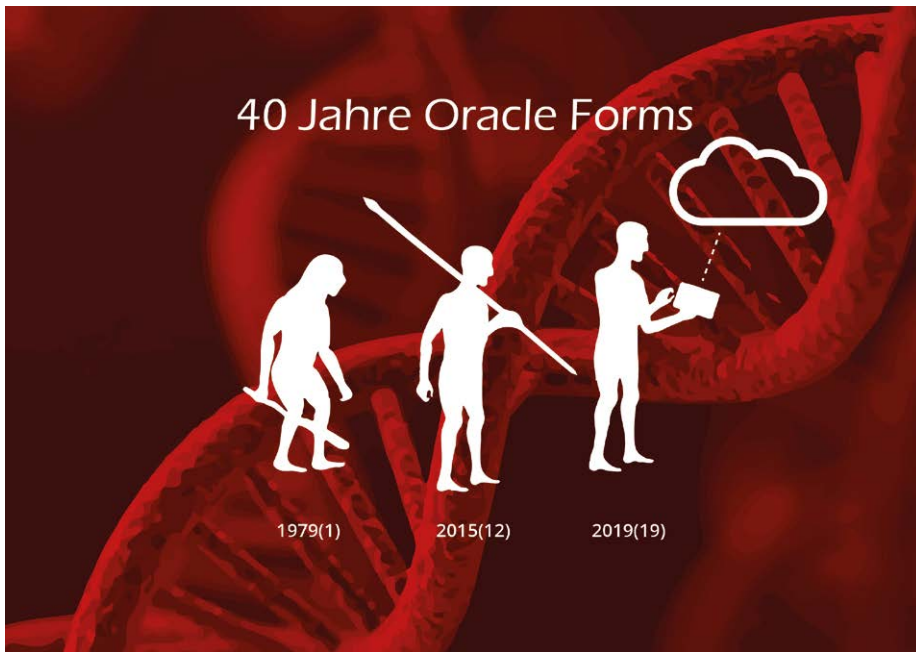


Abbildung 1: 40 Jahre Oracle Forms

Datenbank-Abfragen präsentiert und Bill über seine Programmier-Erfahrungen gesprochen hatte, wurde man sich schnell einig. Dass Bill weder einen Universitätsabschluss vorweisen noch jemals zuvor mit C entwickelt hatte, störte dabei nicht – das könne man ja noch lernen. Für Larry und Bob schien Bill genau der Richtige zu sein, also smart genug, um die Aufgabe erledigen zu können. Der Instinkt sollte sie nicht trügen. So wurde Bill Friend die Nummer fünf bei Oracle, die zweite Person, die man einstellte, und eine starke Unterstützung für den Aufstieg von fünf auf fünfhundert Mitarbeiter in den nächsten Jahren. Er kaufte sich sofort das Buch „The C Programming Language“ von Brian Kernighan and Dennis Ritchie (K&R book), das 1978 erschienen war, lernte autodidaktisch und mit großer Freude C und hatte genau den Job gefunden, den er immer gesucht hatte.

Ein einfacher Auftrag von Larry Ellison brachte Oracle Forms

Neben der Einarbeitung in C beschäftigte sich Bill Friend mit dem neuen relationalen Datenbank-System und arbeitete eng mit Bruce Scott zusammen. Nach ein paar Monaten beherrschte er C und hatte auch einen guten Überblick über die Datenbank-Technologie sowie deren APIs.

Dann musste er selbst herausfinden, womit er sich genau beschäftigen sollte. In dieser Zeit wurde niemand überwacht, bevormundet oder kontrolliert – jeder machte das, was er am besten konnte, mit großer Leidenschaft.

Bill fehlte ein konkretes Projekt und so ging er mit dieser Bitte zu Larry Ellison. Der sagte: „Wir brauchen einen komfortableren Weg, um Daten in die Datenbank zu schreiben, als über einen „INSERT“-Befehl. Vielleicht zeigst du für jede Datenbank-Spalte („column“) ein „Prompt“ an – damit ging es los. Aus diesen sehr wagen Vorgaben erstellte Bill Friend ein detailliertes Konzept, das deutlich mehr bot, als nur den Wunsch von Larry zu erfüllen. Das Ziel von Bill bestand darin, das Leben für den Masken-Entwickler und -Anwender zu erleichtern und eine hohe Produktivität bei der Masken-Entwicklung zu schaffen. Passend zur 4GL-Theorie schreibt Wikipedia dazu: „Als fourth generation language oder kurz „4GL“ bezeichnet man Programmiersprachen beziehungsweise Umgebungen der vierten Generation. Diese sind darauf ausgerichtet, rasch – mit möglichst wenigen Codezeilen – für einen bestimmten Anwendungsbereich Funktionen oder komplette Anwendungen schreiben zu können.“ Analog dazu die Ziele, die sich Bill Friend stellte:

- Der Design-Prozess einer Eingabemaske sollte interaktiv und einfach änderbar erfolgen.

- Der Anwender sollte zur Laufzeit dynamisch durch die Eingabemaske geführt werden.
- Die Runtime-Engine soll die Transaktion zur Datenbank dynamisch erzeugen. Der Entwickler der Maske sollte nicht irgendetwas programmieren müssen, um Standard Insert-, Update-, -Locking-, Delete- und Query-Transaktionen durchführen zu müssen, und ein Rollback sollte bei Fehlern möglich sein. Alle SQL-Befehle sollten dazu noch in einer korrekten Reihenfolge ablaufen.
- Der End User sollte in der Lage sein, auch komplexe Abfragen selbst erstellen zu können, mit relationalen Operatoren wie „<“, „>“, „LIKE“ und in Verbindungen mit anderen Variablen.
- Das relationale Modell sollte logisch mit Transaktionsintegrität umsetzbar sein.
- Dateneingabe von Beziehungen zu anderen Tabellen sollten auch einfach über Wertelisten („List of Values“) möglich sein.
- Für automatisierte SQL-Transaktionen wurden Pre- und Post-Trigger entwickelt, um erweiterte SQL-Logik in einer prozeduralen Abwicklung zu ermöglichen.
- Das Beste aus zwei Welten sollte vereint werden. Dynamisches SQL mit Transaktions-Management und Möglichkeiten der erweiterten Logik mit Laufzeit-Trigger. Diese „SYSTEM-R“-Trigger waren eine Oracle-Vision und noch nicht in der Datenbank implementiert, zuerst in Forms.
- Das Tool sollte auf allen Betriebssystemen lauffähig sein.
- Mit einer „CRT“-Maske sollte das System auf verschiedene Bildschirme anpassbar sein (zum Beispiel 24x80 character grids auf einer „cathode-ray-tube“).
- Die Eingabe musste auch ohne Bildschirm via „teletypes“ möglich sein, weil nicht jeder Arbeitsplatz zu dieser Zeit einen Bildschirm hatte.

Mit allen Möglichkeiten, die die Programmiersprache C bot, hatte Bill Friend nach etwas mehr als vier Wochen die erste Version von Oracle Forms entwickelt. Larry Ellison nannte das Produkt „IAF“ (Interactive Application Facility). Später änderte sich der Name noch ein paar Mal von „Fastforms“ über „SQL-Forms“, bis schließlich „Oracle Forms und Reports“ daraus wurde (ab Version 3).

Weil Larry einem Kunden ein Reporting-Tool versprochen hatte, wurde Bill gefragt, ob er nicht noch schnell eines erstellen könne. Wenn er dafür seinen Urlaub verschieben würde, wäre ein Upgrade (Paris oder Hawaii) möglich. So nahm Bill eine ältere Dokumentations-Lösung von Bruce Scott (FMT) und erweiterte das System um ähnliche Funktionen wie in Forms. Dazu kam die Spezifikation von zusätzlichen Variablen, Select-Statements mit Variablen, Schleifen und IF-Statements. Nach etwa zwei Wochen war RPT fertig und wurde gemeinsam mit der Datenbank ausgeliefert. Bill konnte daraufhin einen schönen Urlaub mit seiner Freundin auf Hawaii machen.

Die ersten Erfolge

IAF und RPT waren ein Volltreffer für Larry Ellison und halfen, die Kundenliste zu vergrößern. Jeder, von der CIA bis zur Bank of America, große Öl- und Gas-Unternehmen, aber auch kleine IT-Consulting-Firmen, entwickelten nun „Forms und Reports“.

In dieser Zeit waren Kunden noch komplett an den Datenbank-Hersteller gebunden; sie konnten selbst keine Software-Applikationen erstellen. Dateneingaben mit „UFI“ (später SQL*PLUS) oder über das neue C-API waren nicht benutzerfreundlich.

Nun konnten sie eine eigene Datenbank modellieren, diese physisch mit UFI erstellen und dazu noch komfortable Eingabemasken und Berichte mit IAF/RPT entwickeln. Mit IMP/EXP kamen weitere Tools von Bill Friend dazu, um Datenbanken zu exportieren und migrieren zu können. Die Datentypen „DATE“, „TIME“ und „TIMESTAMP“ wurden von Bill auch noch in die Datenbank integriert, weil Zeitangaben bei der Datenerfassung oft sehr wichtig sind. Langsam wurde das Ganze rund.

Forms heute

Nach Jahren der Migrations-Abwerbversuche in andere Welten bildete sich in der DOAG eine solide Community unter der Leitung von Dr. Jürgen Menge, die auf der DOAG Konferenz in Nürnberg und einmal im Jahr (dieses Jahr zur Erscheinung

dieser Ausgabe am 21. Februar 2019) zu einem „Forms-Day“ zusammenkommen. Waren es im Jahr 2015 drei Oracle-Forms-Vorträge auf der DOAG Konferenz in Nürnberg, so gab es letztes Jahr schon sechzehn Vorträge. Forms lebt und wird weiterentwickelt – nicht so schnell und agil wie andere Werkzeuge, aber stetig und kontinuierlich. Zudem wächst auch die LinkedIn-Gruppe „Forms und Reports Developer“ wieder. Nach knapp 9.600 Mitgliedern im letzten Jahr sind es aktuell schon 10.051 Mitglieder in dieser Gruppe – Tendenz steigend.

Viele Forms-Kunden sind inzwischen auf die aktuelle Plattform 12.2.1.3 gewechselt, beginnen mit nativen Modernisierungen und schauen auf die bereits angekündigten neuen Versionen Forms 19 und Forms 20. Mit der Version 19 wird primär ein Maintenance-Release erscheinen. In der Version 20 sollen weitere Änderungen eingeführt werden. Wie bei Oracle-Datenbanken heute üblich, orientiert sich die neue Versionsnummer ab sofort an der Jahreszahl der Produktveröffentlichung.

Sehr wahrscheinlich ist nach aktuellem Stand auch, dass die Version 19 noch einmal mit Reports ausgeliefert wird. Das wird allen Kunden noch etwas mehr Zeit geben, Alternativen zu evaluieren.

Laufzeiten von zwanzig oder dreißig Jahren sind bei Oracle-Forms-Projekten durchaus üblich. Kunden planen aktuell, wie Oracle auch mit seiner Enterprise Business Suite (EBS), die in Teilen immer noch auf Oracle Forms basiert, Support und Weiterentwicklung ihrer Applikationen bis zum Jahr 2030. Das sind Laufzeiten, wie sie mit anderen Entwicklungswerkzeugen eher unüblich sind. Mike Ferrante, Produktchef von Oracle Forms, hat folgende Features für die weiteren Versionen (19/20) in Aussicht gestellt:

- REST-Aufruf-Funktionen für externe Services
- Support für SSO with FSAL
- Support mit dem Identity Cloud Service
- Support mit OAuth
- UI-Verbesserungen (Rahmen, Farben, Custom-Colorschema)
- Konfigurierbare Java-Versionen for FSAL
- Support für Java11 FSAL (also Java 11+)
- Forms-Builder-Integration mit FSAL (aktuell nur HTTP-Plug-in)

Zusammen mit den bereits in Forms12 publizierten Neuerungen haben Forms-Entwickler mit Forms19 dann eine Reihe von Features zur Verfügung, um den Masken ein Facelift zu geben. Forms kann und soll auch im Layout professioneller werden, ohne Grautöne, mit hohen Kontrasten und Farbpaletten, die den Anwendern zugutekommen. Eine weitere neue Option wird die Auslagerung der Forms-Entwicklung und des Deployments in die Oracle-Cloud sein. Ganz aktuell stellt Mike Ferrante in einem Whitepaper von November 2018 die neuen Forms-Entwicklungsmöglichkeiten (DevOps) in der Cloud vor. Zur Oracle Cloud wird es auch einen Vortrag des Autors auf dem nächsten Forms Day geben. Bereiten wir uns schon langsam auf das fünfzigjährige Forms-Jubiläum im Jahr 2029 vor.

Weitere Informationen

- Interessante Versions-Tabelle mit Erscheinungsdaten der Oracle-Datenbanken: https://en.wikipedia.org/wiki/Oracle_Database
- Whitepaper DevOps für Oracle Forms in der Oracle Cloud (11.2018): <https://www.oracle.com/technetwork/developer-tools/forms/documentation/oracleforms-in-dcs-5216022.pdf>



Frank Christian Hoffmann
fch@cologne-data.de