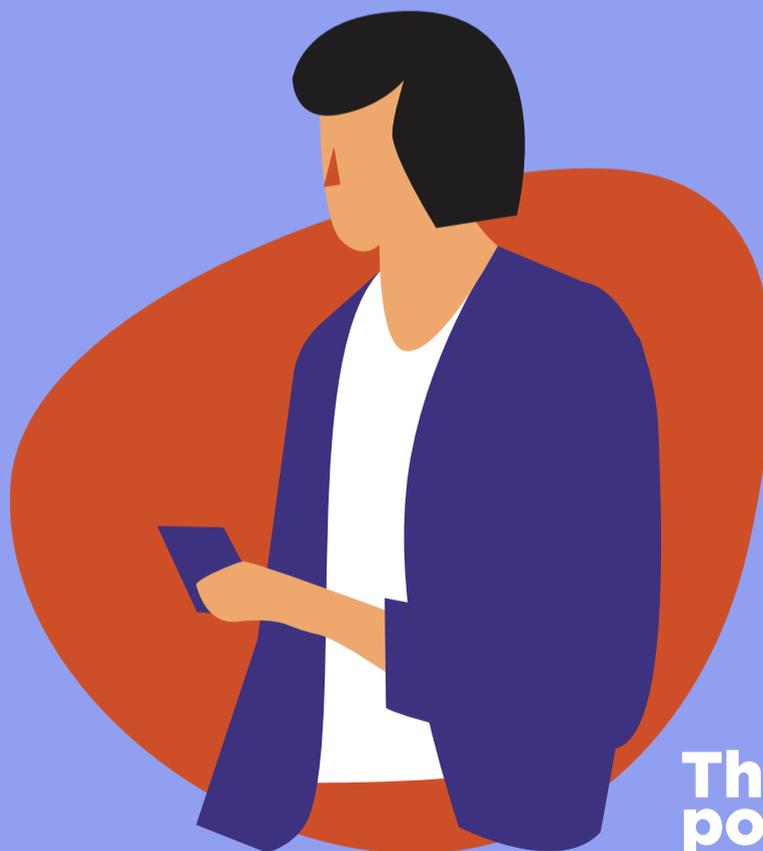


La génesis de GeneXus: algunas reflexiones

By Breogán Gonda *
Copyright © GeneXus, February 2019

Whitepaper



GeneXus™

**The
power
of
doing.**

La génesis de GeneXus: algunas reflexiones

Por Breogán Gonda *

Copyright © GeneXus, febrero de 2019

En 1984, una gran empresa de Brasil, nos encargó la reingeniería total de su informática. El cliente quería desarrollar todos sus sistemas interactuando con una única base de datos corporativa.

El desafío era grande: en aquel momento, todo el mundo hablaba de sistemas y de bases de datos corporativos, pero la realidad era muy diferente y todas las empresas seguían utilizando múltiples "subject databases".

Cada subject database se utilizaba para apoyar una pequeña familia de aplicaciones. Cada subject database era actualizada con total independencia de las otras. Como consecuencia, la consistencia general era imposible. Las empresas lo sabían y, por ello, no combinaban datos de diferentes subject databases.

Resumiendo: se tenía una informática operativa que funcionaba dentro de ciertas condiciones bastante restrictivas, pero no existía una informática corporativa.

El cliente nos planteaba un desafío grande: una única base de datos centralizada para servir a todas sus necesidades.

El cliente pensaba, con buen criterio, que podría obtenerse cualquier información que fuera necesaria, en cualquier momento, a partir de esa base de datos corporativa.

No parecía haber mayores restricciones en lo relativo a los recursos necesarios, pero había un plazo máximo de desarrollo de un año.

La tarea puso a prueba nuestras convicciones, nuestra experiencia, nuestras metodologías y nos enfrentó a un conjunto de problemas adicionales. Todos los obstáculos fueron superados y el proyecto terminó con éxito.

El haber enfrentado tantas dificultades nos hizo aprender muchas cosas y nos alentó a seguir. La duda era, ¿haciendo más proyectos similares o construyendo una metodología propia y una herramienta para soportarla? Éste último fue el camino elegido.

Participamos en varios proyectos para diferentes empresas, mientras seguíamos investigando hasta que finalmente lanzamos nuestro primer GeneXus en 1989.

Hoy, 34 años después de aquel proyecto, nos surge un conjunto de reflexiones sobre un evento que fue trascendente para nuestro desarrollo tecnológico y empresarial.

Teníamos un importante contrato para dar un gran paso adelante, constituíamos un muy buen equipo y, a pesar de que mucha gente en todo el mundo, hablaba o escribía mucho sobre informática corporativa, no había, o por lo menos eran muy escasas, realizaciones exitosas en el área.

Encaramos el tema con enorme seriedad y responsabilidad, lo que nos llevó a enfrentar un conjunto de problemas:

PROBLEMAS

Base de datos. En el supuesto que fuéramos exitosos: nuestra primera estimación nos indicó que la base de datos necesaria iba

a tener más de 500 tablas (en el momento de la finalización de nuestra consultoría tenía 750), ¿funcionarían nuestras metodologías para un problema tan grande? Había que dar respuesta a cuestiones importantes:

Explotación de la solución: El cliente pretendía que su alta gerencia y/o sus asistentes fueran capaces, en casi todos los casos, de obtener sin ayuda de expertos, las consultas que necesitaran en cada momento.

Se dispondría del lenguaje SQL que, desde el punto de vista del usuario no informático, es de bajo nivel, porque para formular la mayoría de las consultas, se le debe indicar de qué tabla deberá obtener cada dato y cómo combinar esas tablas.

Construcción de la solución: ¿Cómo diseñar una base de datos tan grande? Era nuestra especialidad, éramos docentes universitarios, pensábamos que conocíamos bien el tema, asesorábamos en la materia a varias grandes corporaciones brasileñas (petroleras, financieras, industriales, gubernamentales, comerciales...), pero ninguna de ellas pretendía una base de datos corporativa muy grande, muy compleja y que requiriera un alto régimen transaccional, sino que todas tenían varias "subject databases" cada una de las cuales les ayudaba a resolver una pequeña familia de problemas.

Tamaño del modelo de datos. ¿Cómo razonar, en la fase de diseño, con un modelo de datos de más de 500 tablas?, ¿cómo visualizarlo?, ¿cómo normalizarlo, o deberíamos renunciar a la normalización?

Seguramente se cometería un conjunto de errores humanos. Sería bueno tener herramientas para ayudarnos: ¡pero no existían!

Las fuentes del conocimiento. Suponiendo que obtuviéramos o construyéramos las herramientas necesarias, precisaríamos materia prima para hacerlas funcionar. Esa materia prima era el conocimiento, a todo nivel y

de una manera rigurosa, de la realidad del usuario (visones de datos, reglas del negocio, reglas generales de consistencia, etc.).

Para construir el modelo ER – Entity Relation-

Tener herramientas para ayudarnos con esos problemas hubiera sido muy beneficioso, ¡pero no existían!

ship, que pretendíamos, ¿quién en la empresa cliente tenía ese conocimiento de los diferentes objetos de la realidad y sus relaciones, con la objetividad y el detalle necesario?: NADIE.

El mantenimiento del sistema. Dado el tamaño de la solución y suponiendo que la hubiéramos construido siguiendo estrictamente una buena metodología de desarrollo, ¿qué ocurriría al retirarse los consultores?, ¿se seguiría escrupulosamente con la metodología?, ¿cuándo se necesitara recurrir a la documentación, se obtendría información fiel, precisa y actualizada?

Es casi imposible que una metodología utilizada para construir una solución se siga aplicando cabalmente en el tiempo, a menos que sea soportada por una **herramienta de desarrollo** que guíe y obligue a cumplir estrictamente con sus reglas.

SOLUCIONES

Describir precisa y rigurosamente. Sólo podría resolverse el problema y mantenerse válida con el tiempo dicha solución, si lograra-

GeneXus infiere el modelo de datos, el esquema de la base de datos y los programas de aplicación.

mos describir la realidad del cliente de una manera precisa y rigurosa.

Fuentes del conocimiento. Concluimos que la fuente básica del conocimiento son las visiones de datos de usuarios, expresadas dentro de un marco de referencia sólido y simple.

Inteligencia Artificial. Las soluciones manuales no resolvían un problema tan grande, por ello recurrimos a la Inteligencia Artificial. Podría pensarse que esto significa, simplemente, el uso de ciertas herramientas. Esa es una manera simplista e incorrecta de ver las cosas.

Desde luego que analizamos los lenguajes más utilizados en la época que eran LISP y PROLOG y aparecieron dos posiciones: utilizar LISP porque, para él, existía una pluralidad de procesadores de lenguaje y, además, era el lenguaje más antiguo y más utilizado, o escoger PROLOG un lenguaje nuevo, con mayores pretensiones, pero con poco uso y para el que existían muy pocos procesadores

de lenguaje. Optamos por PROLOG y fue una buena decisión.

La Inteligencia Artificial parecía muy promisoriosa y muchas empresas, sobre todo en EE UU, estaban tratando de desarrollar sistemas expertos, especialmente para diagnósticos.

En un diagnóstico, una confiabilidad de un 95 o 97%, por ejemplo, es excelente (la de un muy buen especialista humano es generalmente menor). Pero en la construcción de grandes modelos de datos y sistemas es pésima: ¡se necesita un 100%!

Pero había otro detalle: nosotros éramos ingenieros en computación, con una buena formación matemática, pero no sabíamos nada de Inteligencia Artificial.

UN MUNDO NUEVO

El problema nos llevó a un mundo nuevo. ¡El mundo del conocimiento puro!

Learning by doing. Cuando no sabemos resolver un problema pero, lo estudiamos y finalmente acabamos resolviéndolo, logramos mucho más que eso: **learning by doing** es hoy la manera de mantenerse actualizado y poder utilizar de inmediato y con gran provecho las nuevas tecnologías.

Rigor. Trabajar con conocimiento puro tiene un conjunto de requerimientos que podemos sintetizar en una palabra: **rigor**.

Debíamos y debemos (los investigadores, los que participamos de alguna manera en la concepción y construcción de GeneXus) trabajar con un nivel de abstracción muy alto, de modo de expresar cada problema con toda precisión, como un problema de matemática y lógica puras.

Todas nuestras especificaciones deben ser totalmente rigurosas y basarse en un claro y sólido marco de referencia:

- Representar el **significado** de los diferentes elementos, de manera de poder operar automáticamente con él.
- Adoptar una **regla básica**; la realidad es consistente, luego, cualquier representación que hagamos de ella debe ser consistente.
- Implementar **operadores poderosos** a ser utilizados automáticamente.
- ¿Y nuestros usuarios? (los desarrolladores): No sería realista exigirles trabajar permanentemente con un muy alto nivel de abstracción. Debemos **esconder la complejidad** de manera que los desarrolladores trabajen en alto nivel, de una manera directa y lo más simple posible, con elementos concretos y requiriéndoles una abstracción normal.

Conocimiento puro. Basarnos en conocimiento puro significa no apoyarnos en elementos físicos ni tecnológicos de ningún tipo (que pueden cambiar y, ciertamente cambiarán con el tiempo).

Sólo en el momento de la generación de código, incluir automáticamente los elementos físicos y/o tecnológicos necesarios.

Lo anterior permite crear un alto nivel de abstracción, totalmente independiente de elementos variables casuísticos. Todas las especificaciones se almacenan a éste alto nivel

Futureproofing. El conocimiento se almacena y se opera con él en una Base de Conocimiento que tiene una enorme capacidad de inferencia.

GeneXus infiere el modelo de datos, el esquema de la base de datos y los programas de aplicación.

Una de las consecuencias más importantes es que, como este conocimiento es independiente de la tecnología utilizada, podemos siempre generar automáticamente un sistema utilizando otra configuración tecnológica.

Simplemente debemos utilizar, en cada caso, la versión de GeneXus que soporte la configuración tecnológica escogida..

O sea, se opera el Futureproofing: con GeneXus los sistemas quedan protegidos ante los cambios determinados por el continuo desarrollo de la tecnología. **¡No se necesita reprogramar nada manualmente!**

Mantenimiento automático y evolución.

Una consecuencia menos espectacular que la anterior, pero no menos importante, es el mantenimiento automático de sistemas. La capacidad de inferencia de GeneXus permite originalmente la generación automática del sistema (base de datos y programas) y, cuando hay cambios, los propaga automáticamente. El desarrollador sólo necesita actualizar, a nivel conceptual, los elementos descriptivos que cambiaron. GeneXus dará un informe del impacto de dichos cambios y, si son aceptados, los propagará automáticamente.

INTEGRACIÓN

Nadie, ninguna empresa por grande que sea hoy, y en cualquier área, es autosuficiente. Cada vez más es necesario complementarse con los descubrimientos, la experiencia y los productos de otros.

Es bueno:

- Facilitar al máximo la integración con otros productos para resolver los problemas reales de los clientes de la manera más completa, rápida y económica posible.
- No desarrollar nada que se pueda comprar. Definiendo "se pueda comprar" considerando la calidad de la solución a incorporar, su tiempo de disponibilidad y su costo/oportunidad.

GeneXus ha incrementado mucho, en las últimas versiones, sus capacidades de integración.

RESUMEN Y FUTURO

Hemos separado la administración del conocimiento de la administración de la tecnología. Todo lo dicho de nuestra tecnología y, en particular, del uso automático del conocimiento puro podríamos resumirlo en una frase:

“Hemos logrado una muy buena administración automática del conocimiento de los Sistemas de Negocios”

Pero el mundo evoluciona, ¿dónde está el límite de dichos Sistemas de Negocios? Hoy, empresas de los ramos más diversos, se están posicionando como empresas de software con especialización en tales áreas.

GeneXus representa el conocimiento de una manera rigurosa y permanente y, como resultado, opera automáticamente con él. ¿Qué tipo de conocimiento?, ¡cualquiera!, ¡debemos siem-

pre trabajar mucho, pero no existen límites! Por eso será bueno evolucionar permanentemente: nuestros clientes lo necesitarán. Se volverá necesaria la continua adición de nuevos tipos de aplicaciones –impensables hasta poco tiempo atrás– que harán los sistemas cada vez más complejos.

La sólida base científica y tecnológica de GeneXus nos permitirá, incrementalmente, continuar construyendo las soluciones que se tornen necesarias.

***Breogán Gonda.** *Ingeniero, desarrollador de software, consultor, investigador, profesor. Actualmente es presidente del directorio de GeneXus.*

La sólida base científica y tecnológica de GeneXus nos permitirá, incrementalmente, continuar construyendo las soluciones que se tornen necesarias.

Descubra cómo GeneXus
puede ayudarlo con su negocio.

www.genexus.com

