IBM VisualAge Generator: transitional technology for rapid e-business application development

The application development challenge

Today more than in the past, enterprises worldwide are calling on Information Technology to support strategic business re-architecture in an effort to sharpen their competitive edge. The expectations on IT have never been higher. Organizations must find ways to quickly provide the right information to the right people at the right time within the redefined business processes.

Such climate continues to create a strong demand for application development tooling that boost the productivity of programmers and allows them to build traditional systems while, at the same time, lets them exploit new computing technologies, such as network computing and web enablement.

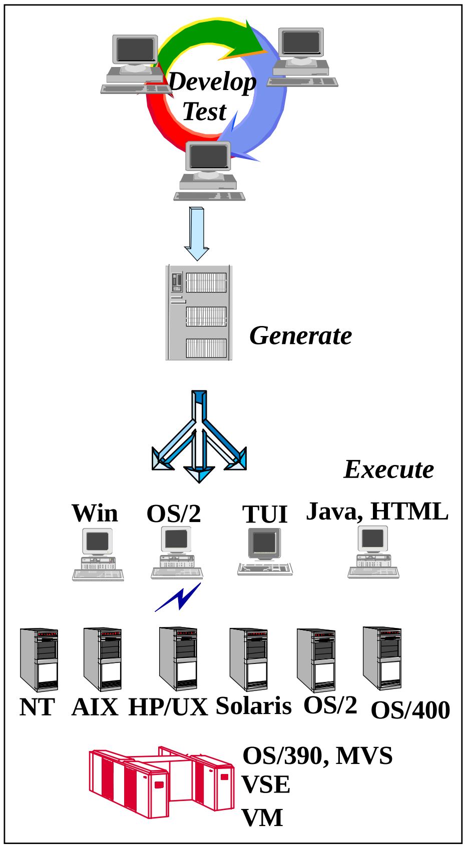
IT managers, in the middle of this firestorm, must deliver new function while finding ways to deal with their existing environments. They must find ways to leverage existing programs and data. They must transition the existing skill of their staff to the objects oriented and component world. And they must deliver scalable solutions across a widely heterogeneous system environment bridging them over to the new web-centric infrastructure. These challenges are defining the key requirements for a new generation of Application Development tools.

IBM’s answer: VisualAge Generator

IBM is today one of the recognized leaders in Application Development technology with its VisualAge family of products, which pioneered the now widely accepted concept of visual construction from components.

VisualAge Generator is IBM's VisualAge offering focused on bringing productivity to organizations facing these challenges. It is a powerful, integrated development workbench used by programmers to fully define, test, build and deploy traditional as well as web-ready enterprise level systems on a variety of platforms and in record time.

With VisualAge Generator, applications are defined in a productive desktop environment using easy to learn, powerful, high level specifications that are completely independent from the target runtime environment and hide the complexity of the system software infrastructure (transactional and DBMS APIs, web server complexity, communications protocols, etc.).



A powerful simulation and test environment enables the programmer to fully test the system without ever compiling or deploying to the final target system. This environment, built and integrated into the development workbench, allows rapid iteration between specification and verification.

Once the application is fully verified, a code generation facility can be invoked to transform the high level environment-neutral specifications into native 3GL source code (Java, C++ or COBOL) optimized for the selected execution systems for compilation and deployment in production.

The solution combines the best of both worlds...all the productivity of an iterative desktop development environment with the scalability and the performance of a compiled and optimized 3GL production application.

To fully understand the power of VisualAge Generator, the rest of this article will review the primary design points of this technology:

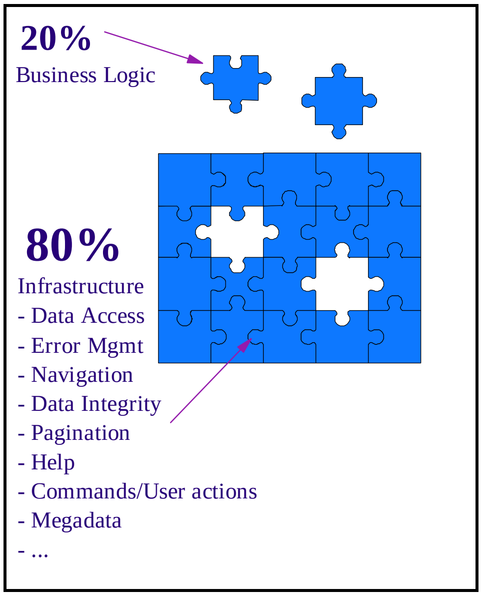
- Provide the highest level of programmer productivity
- Fulfill the demands of an enterprise environment
- Enable rapid deployment of web-enabled e-business solutions

Programmer Productivity.

To address the first primary design point of programmer productivity, VisualAge Generator has been equipped with four core facilities:

1. Data model driven automatic code generation

Industry research has shown that the bulk of the code (up to 80%) written to implement business applications has nothing to do with the business problem, but it is necessary to establish the "mechanical" infrastructure of the system providing functions such as handling error conditions, keeping track of positioning within an array of data, etc.



Using the VisualAge Generator Templates facility, developers can have the application infrastructure (framework) automatically generated by the system, reducing the coding requirements up to 80% and allowing the creation of a fully functional database manipulation application in a matter of minutes!

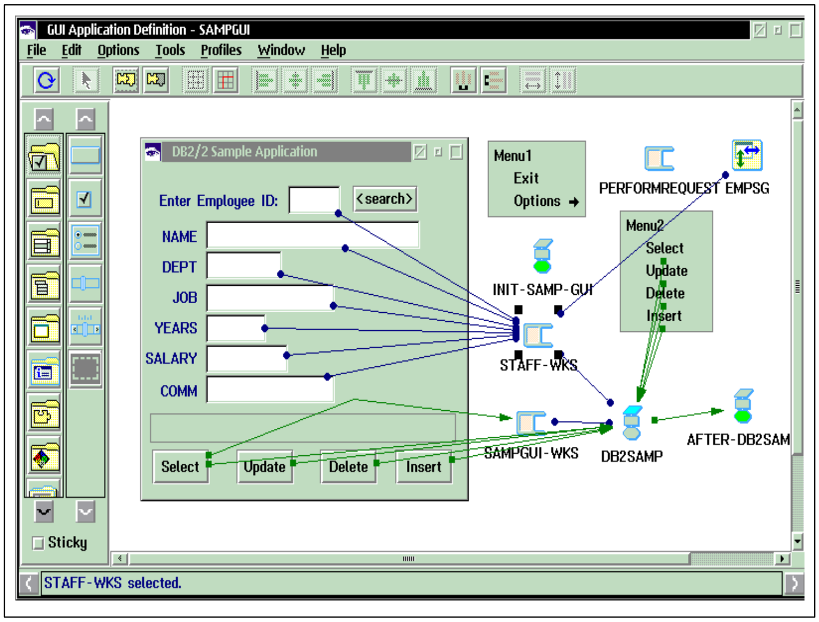
Programmers will build the bulk of their application by simplydefining instances of thetemplate's **Information Model**: this consists of choosing the data to be manipulated (tables, views and columns), and selecting how it will be displayed.This set of intuitive, and seemingly simplespecifications allows VisualAge Generator Templates to automatically generate surprisingly comprehensive application codeto :

- Access and manipulate database (Create, Read, Update, Delete)
- Present the data to the end-user in different ways (Java or traditional GUI, JSP or TUI)
- Manage navigation among multiple windows and session state
- Manage multi-user data access concurrency
- Manage paging when data result set is larger than page size
- Manage error conditions
- Provide window and field level online help

Once this application framework is generated, developers simply add the business logic using the standard VisualAge Generator specification facilities to complete the creation of the system.

2. Visual construction from components

VisualAge Generator includes IBM's award winning visual, component-based construction architecture. This facility is much more than a User Interface(UI) layout builder: programmers can build their own components and make them available for reuse, or they can take advantage of pre-built, pre-tested components sold by IBM or third parties, visually assembling them to meet their system specifications.



Hundreds of reusable components are already available and with the growth in popularity of JavaBean and Enterprise Java Bean component models, thousands more will become available, allowing IT shops to implement a “Buy and Integrate” application development strategy instead of the traditional “develop from scratch” approach.

3. High level 4GL specification facility.

VisualAge Generator provides a simple and yet powerful specification language. This language includes a set of high-level and polymorphic constructs, such as I/O verbs, Unit Of Work (UOW) management verbs, and Remote Procedure Call (RPC), and hides the complexity of underlying target execution environment. The programmer, therefore, can focus on the business problem rather than on writing to complex API’s.
An example of the power of the abstraction level of such specifications is the Web Transaction Rapid Development facility: using a logical data structure definition and simple logical verbs, such as CONVERSE, traditional programmers can build and test, without the need to learn and master complex web technologies such as Javaservlets and JSP, fully functional multi-tiered e-business systems. Another example of the power of the abstraction level is the MQ Series Rapid Development facility: using a simple and terse “sequential file” metaphor and a few high level verbs to read and write Message Queues, saving large amount of low level coding to implement Message based business integration systems.

Beyond the abstraction provided in the language, VisualAge Generator includes many aides to make 4GL programming even more productive. These include powerful utilities such as language sensitive editors, wizards, and graphical assistants. Each is fully integrated and designed to speed the programming effort and help to prevent or eliminate errors.

- Interactive test facility integrated with the development facilities

The most powerful feature of the development workbench is the Interactive Test Facility (ITF). This facility is tightly integrated with the specification environment, allowing programmers to easily specify, animate and verify the application without leaving the development environment. This rapid iteration between specification and verification, frees the developers from costly generation/compilation and deployment steps, and facilitates an evolutionary prototyping development approach.

Fulfilling the enterprise requirements.

An “enterprise class” application development solution must be capable of fulfilling the following key requirements:

- Provide a robust development workbench that can scale up to large development teams.

VisualAge Generator is equipped with integrated repository services that facilitate Software configuration management and version control. In addition, it is possible to transfer VisualAge Generator source specifications between the integrated repository and external SCM tools to be managed along with other application development artifacts participating in the enterprise-wide problem and change tracking and development process management.

- Deliver systems that can scale up to the highest transactional throughputs.

VisualAge Generator generates COBOL and C++ source that can be compiled and run under the control of TX Series, CICS/ESA, and IMS/TM transaction managers. This native support of the most proven TP monitors in the industry assures the transactional scalability required by the most demanding business critical systems.

- Produce systems that can run on a wide variety of platforms

The code generated by VisualAge Generator can run on a variety of client and server platforms, including Windows 95/98, Windows NT, Sun Solaris, OS/2,AIX, HP/UX, SCO Unix, OS/390, OS/400, as well as other 390 systems such as VSE and VM. IBM also intends to continue to expand platforms coverage to other key environments such Linux and others according to market requirements.

- Ease business integration, allow access to legacy data and integration with existing legacy systems,

VisualAge Generator systems can easily take advantage of Message and Queuing technology through the powerful MQSeries RAD facility, allowing rapid construction of business integration solutions. VisualAge Generator programs can reuse existing legacy programs through a simple CALL API, and can also be invoked by hand-crafted existing COBOL or other 3GL programs. In addition, VisualAge Generator programs can access not only the most popular Relational DBMS, DB2/UDB family, Oracle and others, but can easily manipulate legacy file systems, such as VSAM, and non relational IBM databases, such as IMS/DB and DL/I. This allows to easily integrate new applications into existing IT infrastructures maximizing the protection of the previous investments.

- Deliver enterprise IT assets to the web

  As explained in more detail in a separate section of this paper, VisualAge Generator allows to develop end-to-end multi-tier systems with Java servlet/JSP dynamic HTML front-ends, transparently connected to transactional servers running on any of the supported server platforms and transactional systems mentioned earlier in this paper.

- Be useable by traditionally skilled programmers

  Although VisualAge Generator in tightly integrated with a pure Object Oriented programming environment (Smalltalk or Java) the programmer can exploit these technologies without the need to learn OO programming, and simply use the procedural 4GL specifications. The power of the underlying OO technology, however, can be gradually unleashed as the developer becomes more familiar with the component architecture and begins to explore and exploit the additional facilities.

Delivering e-business solutions

Developing e-business solutions is quickly becoming the next strategic direction as companies see the potential for cost saving, better customer service, and streamlined business processes. This will be possible only if the enormous value of the existing IT assets can be easily brought into a web centric world. But the web is an unforgiving new market, where your competitor is just one click away: building self-serve systems that do not provide adequate response and availability can be more damaging than not entering the market at all!

IBM is leveraging it’s experience in delivering highly available, scalable and secure systems to equip this new world with the correct infrastructure and tools. The result is a set of products and architectures that will support the creation and deployment of end-to-end multi-tiered e-business systems. If we think of an e-business system as a logical 3-tier solution, IBM WebSphere Application Server provides the runtime environment for tier-2, WebSphere Studio provides the tools for web site management and page composition, and the VisualAge products allow professional programmers to rapidly create transactional data servers and business logic. In this context, VisualAge Generator plays a central role for the development of new third-tier transactional servers which must guarantee high performance and transaction volumes, and reach a variety of legacy platforms and data, automatically generating the code necessary to use their services in a servlet/JSP dynamic HTML context. This is possible today through three facilities.

- Web Transaction Rapid Development

  The Web Transaction Rapid Development facility is an innovative and simple methodology that allows traditionally skilled programmers to develop, test and deploy multi-tiered web systems without having to deal with the complexities of tier-2, such as designing and developing servlets and JSP that manage and contain data provided by tier-3 servers (mapping data between java objects and flat data structures, understanding API for host connectivity, and so forth), manage sessions and state, etc. Using this approach, developers can specify their system at a logical level, and let VisualAge generate both tier-2 (Java and JSP) and tier-3 (C++ or Cobol) code that implements those specifications.

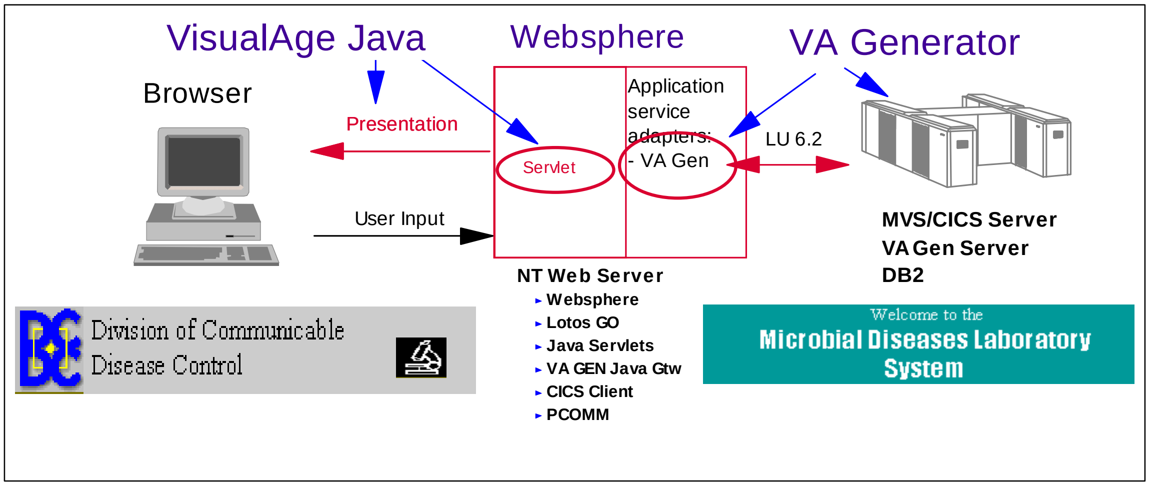
- Integration with VisualAge for Java

  The VisualAge Generator Developer workbench is seamlessly integrated with the award winning VisualAge for Java workbench to provide the most powerful e-business Integrated Development Environment (IDE) in the industry. Programmers can develop and test Java clients (applications, applets or servlets) using either Java or 4GL specifications, or a mix of both, easily exchange data content between Java variables and VisualAge Generator data items, cross invoke business logic implemented in the two languages, and animate both source definitions in a seamless interactive debugging environment. Tier-3 server logic, specified in 4GL, can be defined and tested together and seamlessly with the client prior to the generation of Java, C++, or COBOL for the target runtime. A full end-to-end multi-tiered Java-to-any-server solution can easily be developed, rapidly and without necessarily having to learn deep Java skills, in a iterative and dynamic environment that does not require lengthy runtime software setup.

- Generation of Java Beans and Enterprise Java Beans.

  Programmers can automatically generate, from VisuaAge Generator server programs specifications, Java Beans and Enterprise JavaBeans that can be used within any Java program (applet, application or servlet) to connect to and exchange data with a VisualAge Generator server program. Generation of EJB makes it possible to reach tier-3 VisualAge Generator programs within a EJS (Enterprise Server for Java) runtime environment, such as that provided by IBM WebSphere Application Server Enterprise Edition. The generated bean and the gateway program automatically perform all the data marshaling and conversion necessary to map Java to traditional back-end transactions.

  The State of California Department of Health Services has recently brought to the web their Microbial Diseases Laboratory System using a razor-thin client architecture and Java Servlets running on IBM's WebSphere and connecting to new MVS back-end transactions developed using VisualAge Generator. The system, currently in production throughout the State, was developed in a few months by 6 programmers all new to VisualAge



Product Evolution

As the topology of e-business solutions continues to evolve towards thin clients and component-based servers, the Enterprise JavaBeans (EJB) open industry standard will establish itself as the key technology for deploying scalable and flexible solutions.

VisualAge Generator, is designed to enable enterprise developers to create application systems through very productive high level abstractions that simplify the programming complexity of a wide range of runtime platforms and transactional environments. As the EJB platforms become available, the need for high productivity tooling will extend to such new environments, and VisualAge Generator intends to continue supporting the emerging Enterprise JavaBean Servers (EIS) as new runtime environments within the context of web application development VisualAge Generator will therefore be enhanced with the ability to logically define business objects which will be automatically generated into EJB Entity Beans.

IBM also intends to continue enhance VisualAge Generator code generation capabilities with additional server side Java generation, and expand to additional runtime server platforms (such as Linux, and others) according to market requirements. This will enable Customers to continue to rapidly build and deploy e-business systems that meet the most stringent availability, scalability and performance requirements of today's and tomorrow's networked business world.