



Objects in a typical nTegrator application are made up of a mix of standard components supplied with nTegrator and custom components developed by the customer.

At its current state of development, nTegrator generally requires some programmer intervention for development of some of the more complex custom components. Using nTegrator **Explorer**, end users can easily create application systems by organizing those components into collections.

As more standard components are developed, both by nBoundary Software Inc. and by customers, there will be a diminishing need for IT professionals to be involved in the development of business information solutions built using nTegrator.

- ☑ Ultra-rapid development of ad-hoc and production applications
- ☑ Access to any data in any form anywhere
- ☑ Simple reusable components
- ☑ Distributed, secure systems, permission protected and public key encrypted
- ☑ Drop-in functionality without recompilation
- ☑ Data internally always in standard XML format
- ☑ Top-down specification, design, implementation using drag and drop construction
- ☑ Small footprint, runs unobtrusively in any Windows device, limits paging, no server needed
- ☑ Infinite reusability of components
- ☑ 'Geographically irrelevant' architecture: applications can be arbitrarily split among computers
- ☑ Minimal involvement of IT professionals required



nBoundary Software Inc.

Alastair Ross Technology Centre
130, 3553—31 Street N.W.
Calgary, AB T2L 2K7

Phone: 403-270-2443
Fax: 403-270-2384
cbuchanan@nboundary.com
www.nboundary.com



Business without boundaries



*Say goodbye to expensive EAI products,
dedicated server software, and complex,
high-maintenance systems.
Say hello to nTegrator.*



Business without boundaries



nTEGRATOR

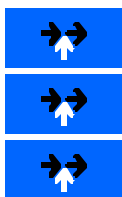


nTegrator is a toolkit for rapidly and simply developing distributed, secure business information systems that provide users with exactly and only the information they need, in exactly the form in which it is needed.

The required data may be drawn from any database, in any format, anywhere.

nTegrator is predicated on the belief that all of an enterprise's data should be available to all authorized users with minimal system development effort. The data's owner should be able to provide access to the data and manage its integrity with ease. There should be no noticeable impact on the performance of the computer that hosts the data.

Application systems may be distributed among as many Windows devices as wished provided they are connected through TCP/IP networks. Compute-intensive tasks can easily be delegated to high-powered servers with results displayed on a PDA, for example.



Every 'point of intelligence' in an application system is an object made up of two types of simple scripted program. All objects are permission protected against unauthorized use or modification, and all data communications are public key encrypted.



Any component and any object may be reused ad infinitum. Additional functionality may be added to any reuse without affecting its original use. Functionality may be added at any time without the need to recompile objects. [Patent pending]

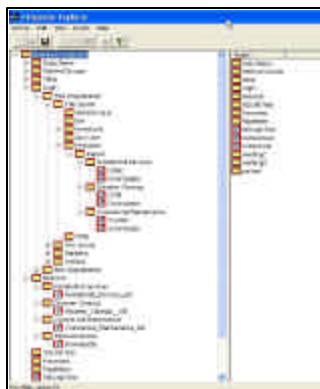
No networking knowledge is required on the part of a system's developer.

```
<?xml version="1.0" ?>
<xmltest>
  <Source1>
    <record loc = 14>
      <Author>Adams, Pat</Author>
      <Phone>(403)555-1234</Phone>
    </record>
  </Source1>
  <Source2>
    <record key = 77>
      <Name>John Doe</Name>
      <Phone>(403)555-5678</Phone>
    </record>
  </Source2>
</xmltest>
```

Objects communicate with each other through messages that contain data and commands in XML format.

nTegrator applications are defined and designed top-down. They are then built in two steps:

Object components are either developed or selected from stock.

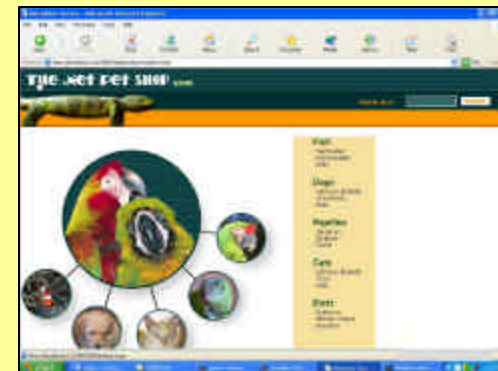


Components are then dragged and dropped into objects using nTegrator **Explorer**, a Windows Explorer-like tool. Objects are organized into collections that define the business logic of the system, also using **Explorer**.

The same tool is used for testing and other system development functions.

Benchmarks for the Pet Shop demo developed by Sun, reproduced by Microsoft, and then implemented using nTegrator, indicate that nTegrator offers significant savings in development effort:

- for user interface, 11% fewer lines of code than .NET and 72% fewer than J2EE;
- for business logic, 53% fewer lines of code than .NET and 93% less than J2EE;



- for database processing, 89% fewer lines than .NET and 81% less than J2EE.

Fewer lines of code means lower cost to implement, easier testing, simpler maintenance. And that doesn't even take into account all of nTegrator's other benefits.

Needing as little as 200K, the nTegrator engine runs unobtrusively in any Windows device. The small footprint limits being paged out of memory by the operating system, thus enhancing application performance.

