Reality
Web Services & XML

Introduction – What are Web Services?

Web Services provide a connection between processes, which are independent of programming languages, platforms and operating systems and are exposed as services. They are built on industry standards and backed by all major software vendors.

Web Services provide a structured way to format data, a standard transportation mechanism and a standard means to describe what the service does. The effect of this is to make the service available to others.

Data is described and transferred using the open industry-standard extensible mark-up language (XML).

Simple Object Access Protocol (SOAP) enables systems to communicate with each other and make requests. SOAP describes what is in the message, who should deal with it and whether a response is optional or mandatory.

Web Services enables partners, customers and internal users to become consumers of services. However, this requires a way to describe which functions are available from a specific Web Service and what information must be passed.

This is achieved using Web Services Description Language (WSDL) to expose sets of services with business partners. Consumers can use applications which access the WSDL of the Web Service to enable fast utilization of the service.

Once a Web Service has been created it can be advertised to consumers by Universal Discovery and Integration (UDDI). This provides a directory, enabling businesses to list services that they provide.

Benefits of Web Services

Greater efficiency:. one standard, one set of tools.

Greater interoperability:. can interact with any consumer - worldwide or locally.

Reduced integration costs:. one service can service all consumers on any platform.

Reduced training requirements:. only need one technology.

Simplified business communications:. one service can be offered to everyone.

**Responsiveness:**. organizations can respond quicker to market demands.

Reduced maintenance:. Web Services promote the use of clean interfaces, enabling components to be modified or switched without effecting business or applications.

Component re-use:. developers now have access to a wide range of services offered by third parties to deliver flexible and integrated solutions.

Externalize business assets:. publish services in a secure manner to provide business integration to customers, partners and suppliers.

Reduced costs:. does not require expensive third party middleware. Uses standard web communication technologies.

Service consumers:. can use any environment which supports Web Services.
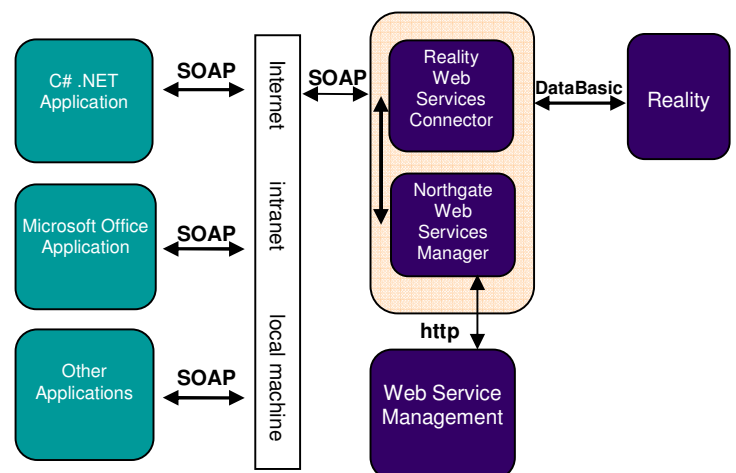
northgate
INFORMATION SOLUTIONS

How Reality Web Services Work

Reality deploys Northgate's Web Services Framework to expose and publish existing DataBasic subroutines as standard Web Services. This is achieved using a connector mechanism designed to simplify integration with proprietary architectures. The Web Services Framework provides a simple web-based management tool allowing Application developers to define connectors to Reality databases and create Web Services using those connectors.

To make use of this essential new technology there is no need to re-engineer existing business logic – it simply needs to be enabled. This allows existing logic and services to be extended to any consumers worldwide, or within the organization via the intranet. The Northgate Web Service Framework transparently constructs WSDL, based on the DataBasic subroutine name and its defined parameters. This removes the need for developers to have any knowledge of XML, WSDL or SOAP technologies.

XML Parsing

Numerous applications, including Web Services, exchange information in XML format. This has generated requirements to construct XML documents and extract data from them.

Reality supports an XML parser, to examine and create XML documents. This provides simple data extraction from an XML document using a query template.

XML Document Generation

This is a simple mechanism for generating an XML document using a template. It combines a template document, with a dynamic array to generate the document easily, defining how attributes are mapped into the generated output. In addition, it supports optional fields in the template in order to minimise the number of templates required.

```
Input = "N Kelly^24 Some road^^^HP21 6NW"
```

Input

```
<customer>
        <name>\1\</name>
        <address>\2\</address>
]       <address>\3\</address>]
]       <address>\4\</address>]
        <postcode>\5\</postcode>
</customer>
```

Template

```
<customer>
        <name>N Kelly</name>
        <address>24 Some road</address>
        <postcode> HP21 6NW</postcode>
</customer>
```

Resulting XML document

XML Document Extraction

This functionality processes an XML document and returns a multi-valued string containing the result set.

A query template is an XML document that defines which elements of the source document are searched, which data elements are to be extracted and any conditions the data must meet; with the result returned to specified attributes.

The template below extracts port names for port 5001 from the <Services> element of <Server> in the XML source document shown and places the output in attribute 1.

```
<server>
    <services>
        <service port="5001" name="%1%" />
    </services>
</server>
```

Template

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
 <server>
    <miscellaneous>
        <logfilespec>
            <entry alue="c:\tcpBridge.log" />
        </logfilespec>
        <maxconnections>
            <entry value="20" />
        </maxconnections>
    </miscellaneous>
    <control-service>
        <service name="Control"port="5400"/>
    </control-service>
    <services>
        <service name="Time" port="5001" />
        <service name="TimeL" port="5001"/>
        <service name="Time" port="5002" />
        <service name="MGate" port="5100" />
        <service name="MGate" port="6100" />
    </services>
    <serial-ports>
        <entry device="COM1" baud="38400"/>
    </serial-ports>
 </server>
```

Source document

```
Result in attribute 1 is  'Time]TimeL'
```

Result in Attribute 1

↑northgate
INFORMATION SOLUTIONS