



Bluebird Project

<http://www.opennms.org>

Build the next generation network and service management system using state-of-the-art web technologies to deliver:

- **Speed:** High speed adaptability to changes in network technology
- **Customizability:** Open architecture in a modular system
- **Price:** No cost implementation
- **Technology:** Portable, modular, web-based architecture for scalability and openness

Jan 1999 - began design work on Bluebird architecture

Mar 1999 - built prototype application using IDE

Apr 1999 - finished prototype, began final design work

July 1999 - quit jobs and moved to small office in Cary, NC

Nov 1999 - took EUI alpha code on a road show to Wash DC, Denver, Florida, Virginia to several very large network providers. Received feedback and integrated changes.

Jan 2000 - tested distributed poller discovery on very large outsourcer, found problems, make corrections

Mar 2000 - announced new project, turned on web site, released source code

Apr 2000 - hired new manager to coordinate 500 team members

April 27, 2000 - 975 members

We currently have 10 full-time team members dedicated to the task of making this successful. We have over 975 signed contributors who have agreed to help us. Management team:

Shane O'Donnell: Open Source Project Manager, 2 years consulting and training, 10 years, primary network management architect for largest private IP network in the world

Brian Weaver: Development Manager, 7 years network management development experience

Steve Giles: Architect, 11 years network management development experience

Luke Rindfuss: Coordinator/Marketing, 2 years sales and marketing experience in network management software

**Not only will we
give you the
shirts off our
backs ...**



Some late breaking news:

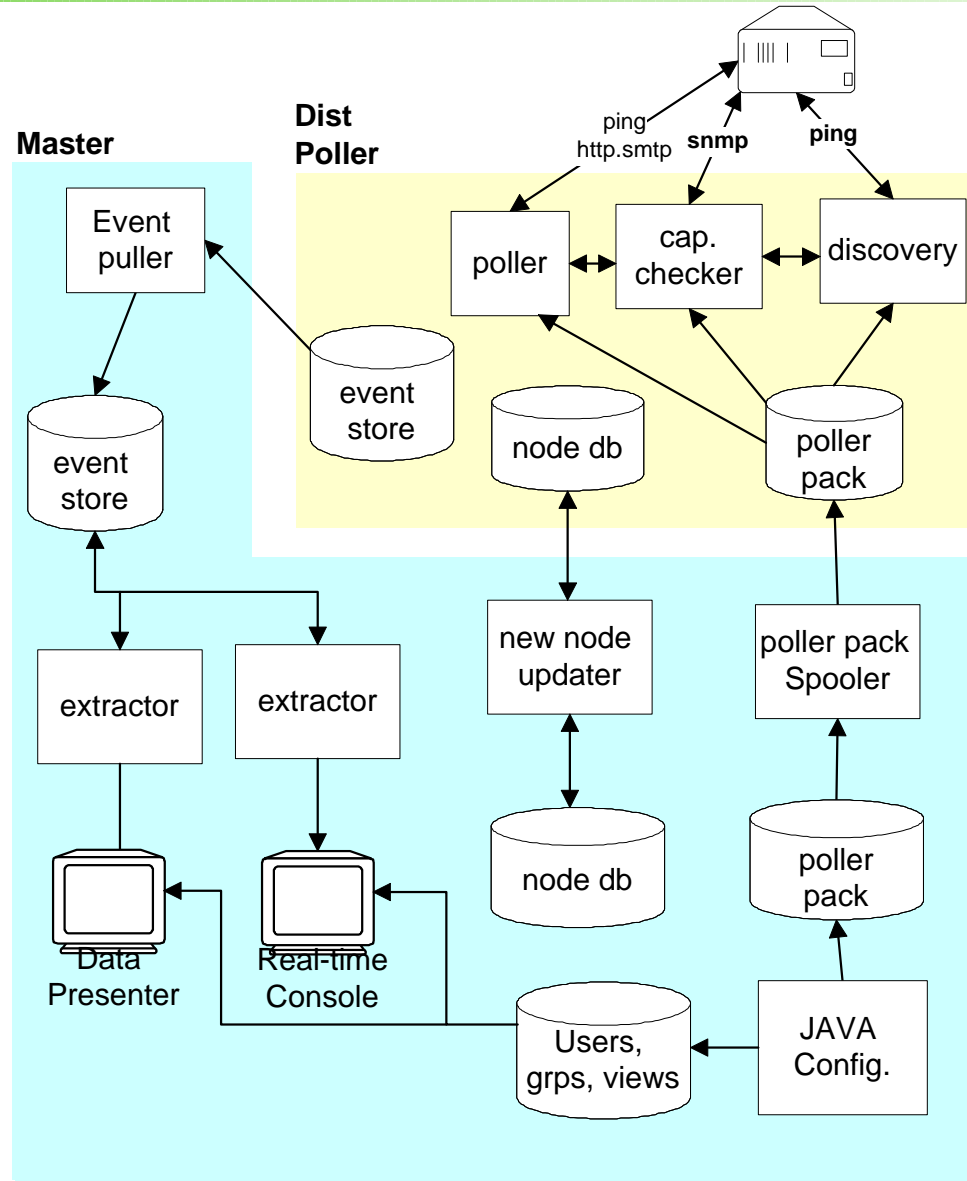
- Over 975 contributors have signed up in first month to help us build Bluebird.
- We have release source code for testing (JAR files) and development (CVS).
- North Carolina's state network system is collaborating with OpenNMS.org to standardize on Bluebird for network management
- Several network management open source projects are closing down and folding their efforts into Bluebird.
- Several very large network element providers have approached us about partnering.

Bluebird Project Overview

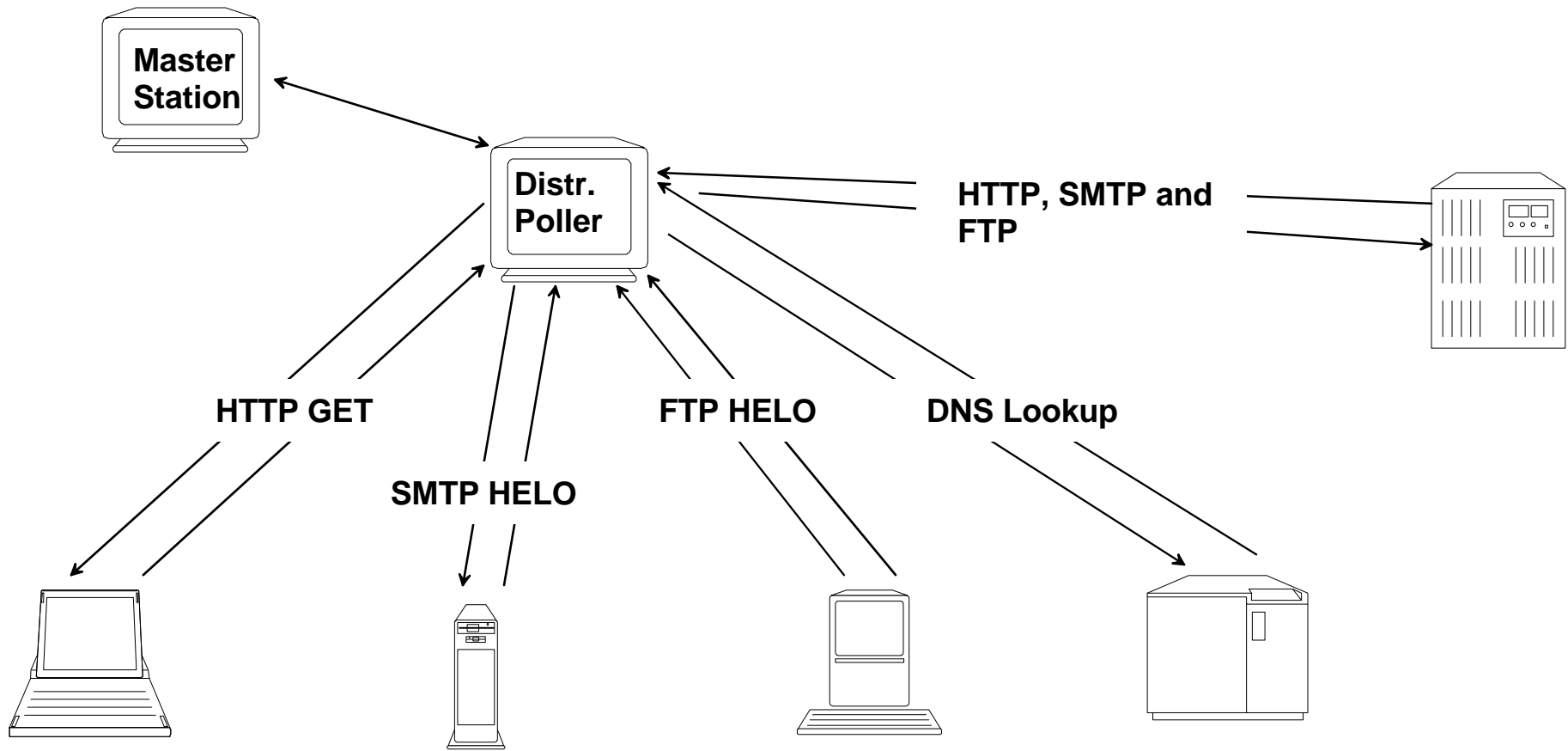
Incorporate the following functional areas into the first release:

- **Discovery:** automatically discover IP addressable devices
- **Filtering:** exclude devices based on rules
- **Service Polling:** HTTP, ICMP, FTP, DNS, SMTP
- **Calendaring:** exclude certain times of day/month from polling
- **Distributed Architecture:** use resilient and redundant methods to communicate with remote pollers (distributed pollers)
- **Events:** consolidate internal events from various pollers into a single store
- **Availability Reporting:** style based reporting and web viewing using XML and XSL
- **Graphical configuration:** JAVA based tools for drag/drop configuration including wizards and help tools
- **Business Views** - different devices for different operators

Functional Diagram



Synthetic Transactions



Synthetic transactions test a service on a network device for service availability. Service tests are configured using rules.

JAVA Configuration:

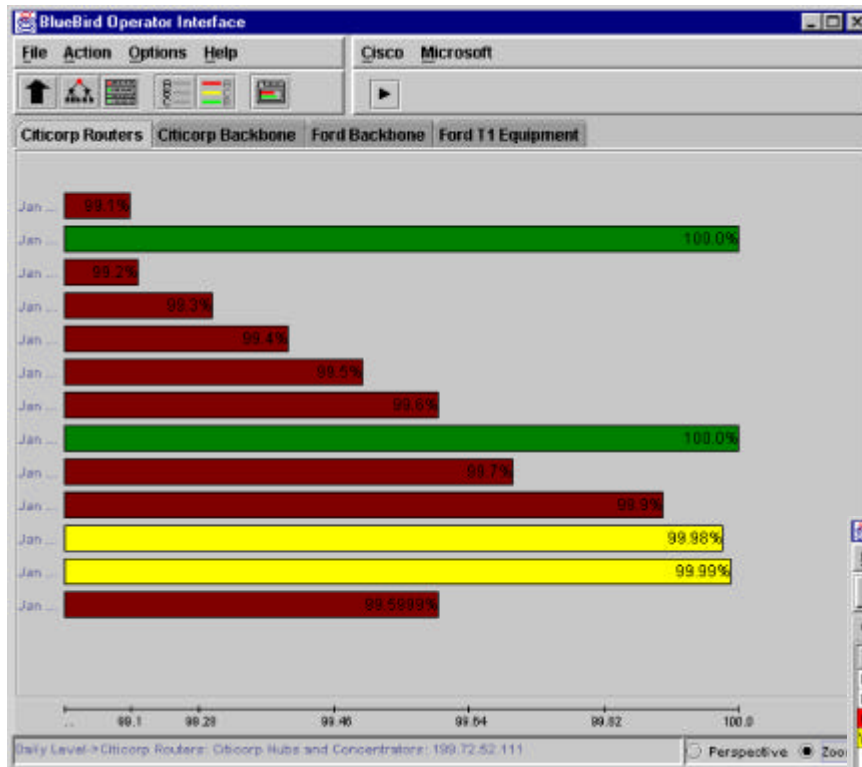
The image displays two overlapping windows from the BlueBird Administrator software. The background window is titled "BlueBird Administrator Administrator Main" and features a sidebar with icons for "Configure SNMP Information", "Configure Distributed Pollers", "Configure Users, Groups and Views", and "Reports and Auditing". The foreground window is titled "Users, User Groups and Views Configuration" and is divided into three panes: "Users", "User Groups", and a hierarchical tree view. The "Users" pane lists names like SallyJ, BobH, SteveH, ScottT, ChittaB, JayK, RandyF, JanetP, BrianW, BrettD, and LisaH. The "User Groups" pane shows a hierarchy of shifts and roles, including "First Shift", "second shift", "Weekend Shift", "Administrators", and "Managers". The tree view on the right lists geographical locations: "New York", "Chicago", "East Coast", "Atlanta", and "Boston". A status bar at the bottom of this window reads "New user LisaH added. Double click to modify."

The second window, titled "Filter Rule Builder", is positioned over the top right of the first. It contains two panes: "Custom Rules" and "Template Rules". The "Custom Rules" pane lists criteria such as "IPAddr ~ 199.72.52.*", "IPAddr ~ 199.72.52.128-256", and "IPHostName ~ ws*52". The "Template Rules" pane lists criteria like "IFType = s", "SNMP Information", "ENMP sysDest ~ xxxx", "ENMP sysCon ~ xxxxx", "ENMP sysOID ~ 1.3.6.1.4.1.x", "ENMP sysLoc ~ xxxxx", and "ENMP sysName ~ xxxxx". Below these panes is a diagram showing a network topology with nodes labeled "source", "IP", "IPAddr", and "sink" connected by arrows. At the bottom of this window is a "Test Rule:" field and a "Redo Layout" button.

Master station configuration tools are built in pure JAVA 1.2.2 using drag/drop metaphor.

User Presentation:

Bluebird eschews the topological view of the network for a service level view.



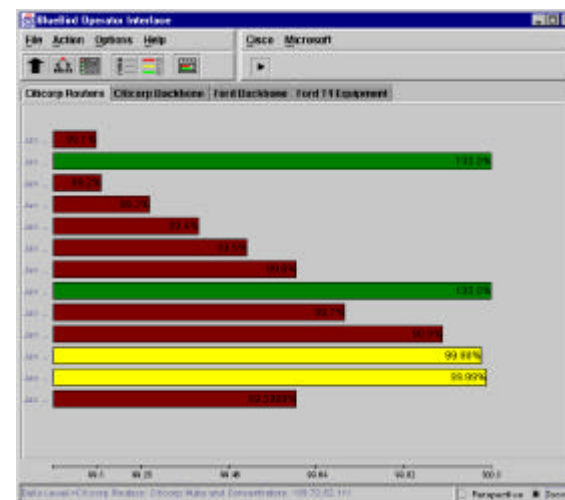
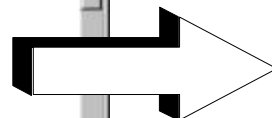
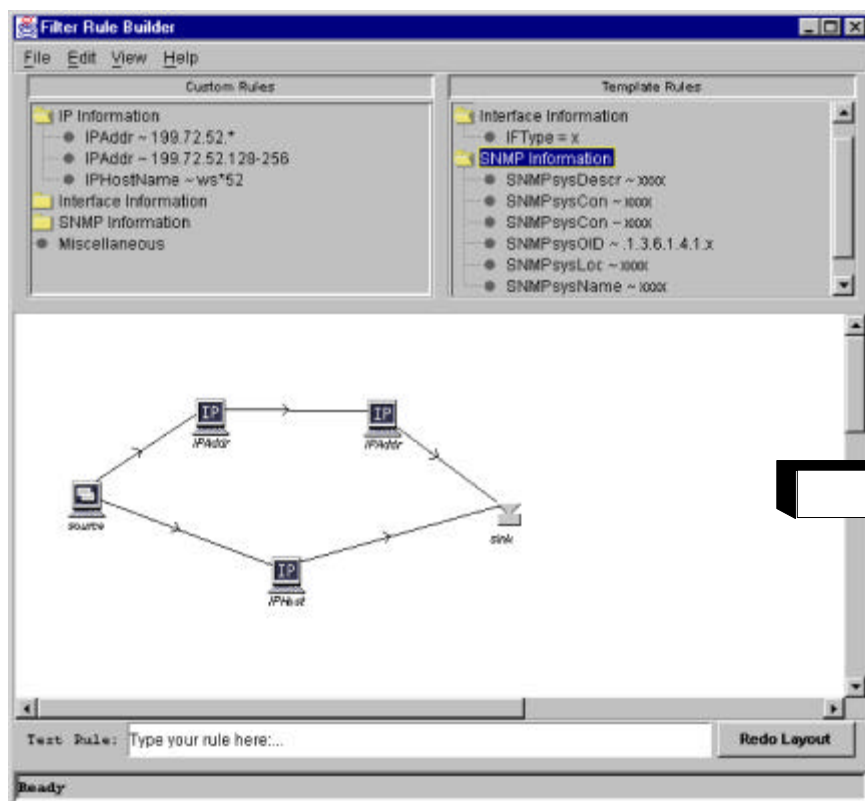
Devices are grouped into business views and categories. Drilling down takes the user to the problem.

The screenshot shows a window titled "BlueBad Operator Interface" with a menu bar (File, Action, Options, Help) and a toolbar. Below the toolbar are tabs for "Citicorp Routers", "Citicorp Backbone", "Ford Backbone", "Ford T1 Equipment", and "Lucent". The main area displays a table with the following data:

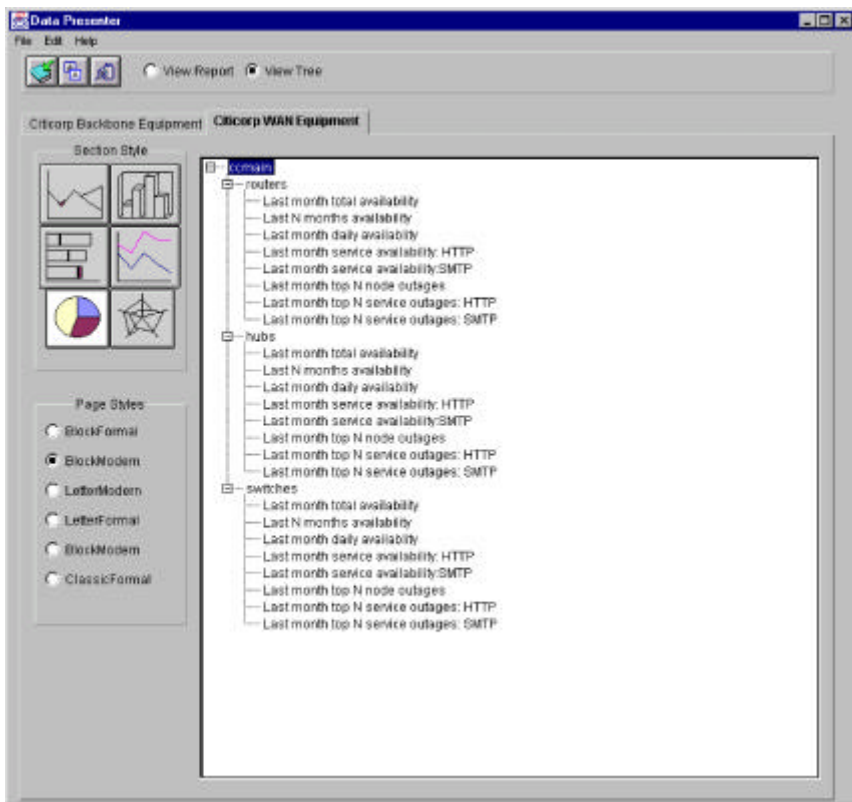
Severity	Time	Host	Event Overview
Informational	Wed, May 12, 1999 01:03:21 AM	12345678 (int)	Roundtrip delay event
Informational	Wed, May 12, 1999 01:01:02 AM	128366019221 (hw)	New node discovered.
Critical	Wed, May 12, 1999 01:03:37 AM	199.72.51.1	Critical node delay exceeds threshold.
Warning	Wed, May 12, 1999 01:03:37 AM	199.72.51.33	Cisco router fan overtemp situation.

A user's view of the network is determined by rules established by the administrator. If a device matches the rule, it is included in the business view.

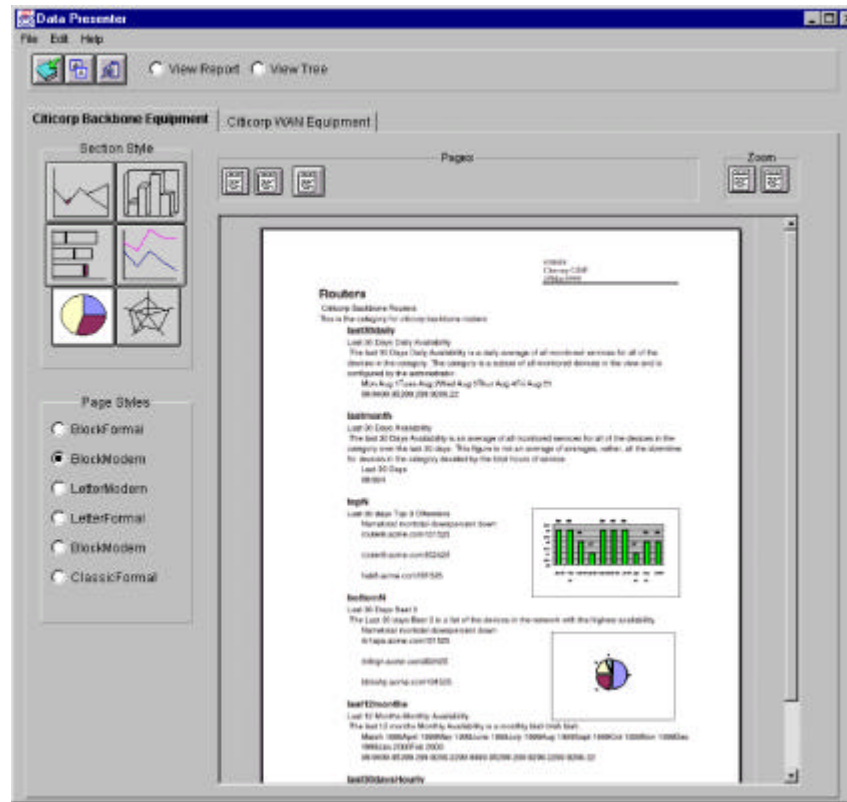
Rules are built using the graphical rule builder.



Tree View



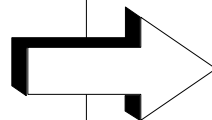
Preview



Reporting is via the JAVA Data Presenter. There is no configuration to the Data Presenter, it automatically builds reports based on users, groups and views.

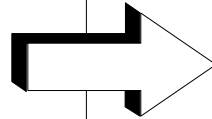
```
<?xml version="1.0"?>
<?XML-stYLESHEET type="text/xsl" href="formdef.txt"?>
<!DOCTYPE report [
<!ELEMENT report (created, viewInfo, categories) >
<!ATTLIST value type (data|title) "data" >
]>
<report>
  <created year="1999" month="May" day="15" hour="19" min="47" sec="03">
    37872728723
  </created>
  <author>Joe Bob Briggs</author>
  <viewInfo>
    <viewName>ccmain</viewName>
    <viewTitle>Citicorp GISP</viewTitle>
    <viewComments>Citicorp Backbone Equipment</viewComments>
```

XML



```
<xsl:stylesheet
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:fo="http://www.w3.org/1999/XSL/Format">
<xsl:output indent="yes"></xsl:output>
<xsl:template match="report">
  <fo:root xmlns:fo="http://www.w3.org/1999/XSL/Format">
    <fo:layout-master-set>
      <!-- COVER PAGE MASTER -->
      <fo:simple-page-master page-master-name="cover" page-
width="8.5in" page-height="11in"
margin-top="4in" margin-bottom="1in" margin-left="1in" margin-
right="1in">
        <fo:region-body border-width="1pt"
vertical-align="middle" margin-top="0.5in"/>
      </fo:simple-page-master>
      <!-- REST PAGE MASTER -->
      <fo:simple-page-master page-master-name="rest" page-
width="8.5in" page-height="11in"
margin-top="0.5in" margin-bottom="0.25in" margin-left="1in"
margin-right="1in">
        <fo:region-body column-count="1" margin-top="0.5in"
```

XSL



Report

AWT Application

Wrapping Label Example (48,24)

The wrappingLabel appears to be a standard part of the swing classes, however, there appear to be a large number of controls for the javax.swing.text component.

In order to position a block of text in a panel like this, it seems to me you need 6 things, the x and y coordinates of the start of the text, the width and height of the text area and the font size and style. BTW, the font and style are only needed if the component cannot tell you how many lines of text it takes to handle the text.

Overlapping text blocks are to be avoided since the text underneath becomes invisible. This is not desirable. Therefore the height and width must be computed exactly or blocks may overlap.

X position (60,216)

Positioning the left margin (x axis) requires knowledge of the region before, the number of margin-befores etc. Margin-top simply moves the text box down the number of pixels.

Y position (60,288)

Positioning the y axis requires knowledge of the relative position of the end of the previous block, the amount of margin-top, the region-start size etc.

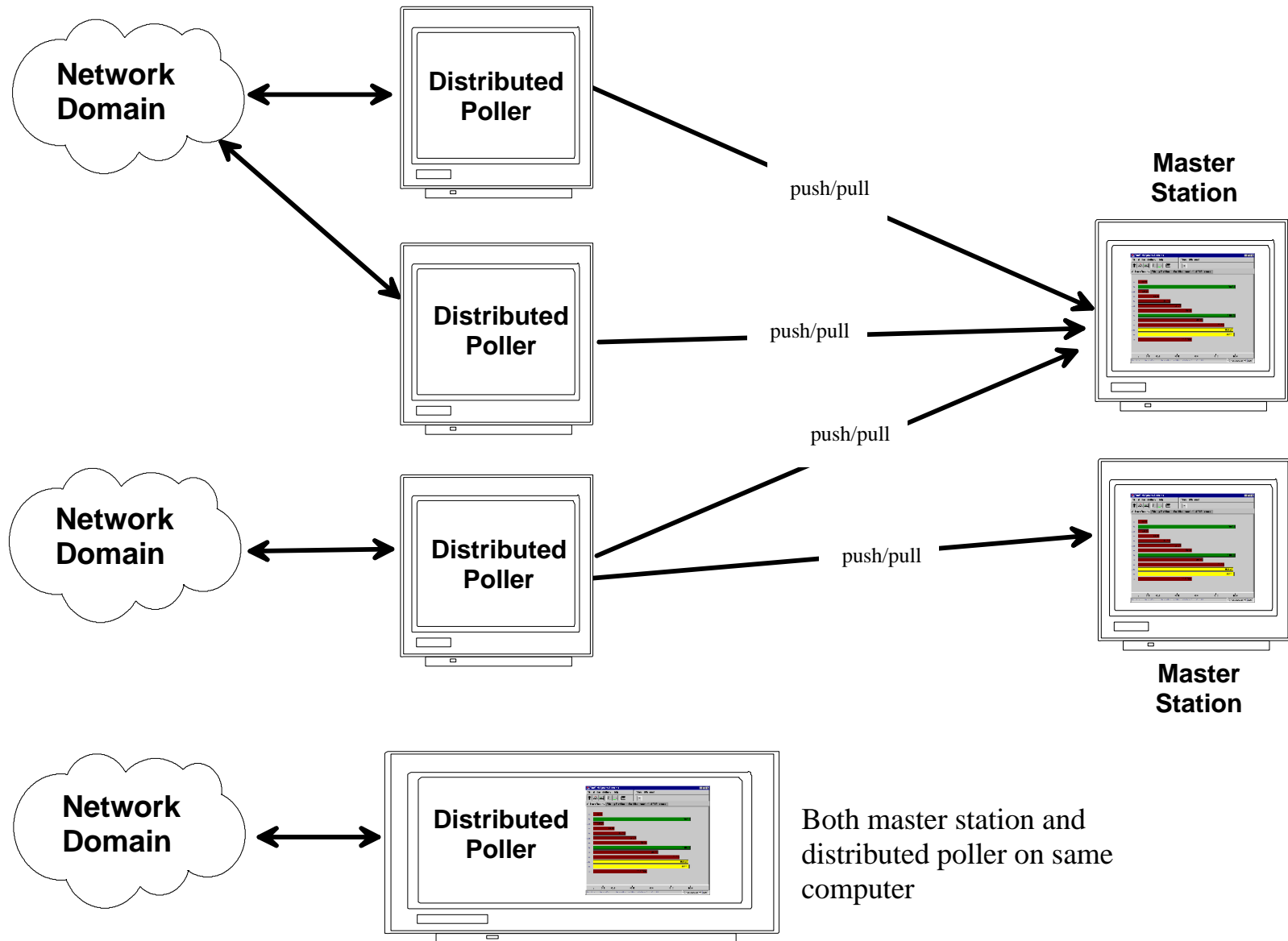
Height (60,360)

Calculating the height of a text box requires knowledge of the spacing between lines (is this controllable in any text components?), the font size, style and text itself. The procedure could calculate the breaks and therefore, the number of lines of text.

Width (60,444)

Calculating the width of the text box requires knowledge of the right margin and the current left margin including indents. This should be a simple computation.

Distributed Architecture



...

```
<panel bgColor="gray" layout="vertical" rowCol="3"/>  
<iconLayout labelPos="top" fontType="Helvetica" fontStyle="italic" fontSize="10"/>
```

```
<tools>
```

```
<tool>
```

```
<icon>com/nm/bb/common/images/bwuser.gif</icon>
```

```
<label>
```

```
<text>Configure SNMP Information</text>
```

```
<hotkey>S</hotkey>
```

```
</label>
```

```
<hint>Set up SNMP timeouts, retries and community strings</hint>
```

```
<classname>com.nm.bb.ms.admin.snmp.panels.SnmpConfig</classname>
```

```
</tool>
```

```
<tool>
```

```
<icon>com/nm/bb/common/images/bwuser.gif</icon>
```

```
<label>
```

```
<text>Configure Distributed Pollers</text>
```

```
<hotkey>D</hotkey>
```

```
</label>
```



Unmodified Regular GPL

Agents - portable, intelligent agents for systems using standard lightweight protocols and localized processing and filtering, SNMP V3 focus for configuration, resource and partition management

Topology - level 2 and level 3 connectivity based on RFC 1493 and proprietary vendor extensions.

Event correlation - critical path analysis to eliminate downstream events

Critical path polling - eliminate downstream polling for devices on the back side of down devices

Development:

- JAVA, C, C++, Servlets, SNMP, ODBC

Documentation:

- XML/DocBook
- User Manuals,
- Convert MS Word Design Documents to DocBook

Speaking/Marketing:

- Local user group presentations
- Web site links (use "Link-to-us" graphic at www.opennms.org)
- Tell your friends
- Company sponsorships

Testing:

- QA, Usability, Scalability

Design/General Ideas:

- Event correlation/root cause