Bluebird Project

http://www.opennms.org

Mission Statement

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

OpenNMS.org
Open Network Management Software

Project History

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

Team 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

OpenNMS.org
Open Network Management Software

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

News Flash!

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.

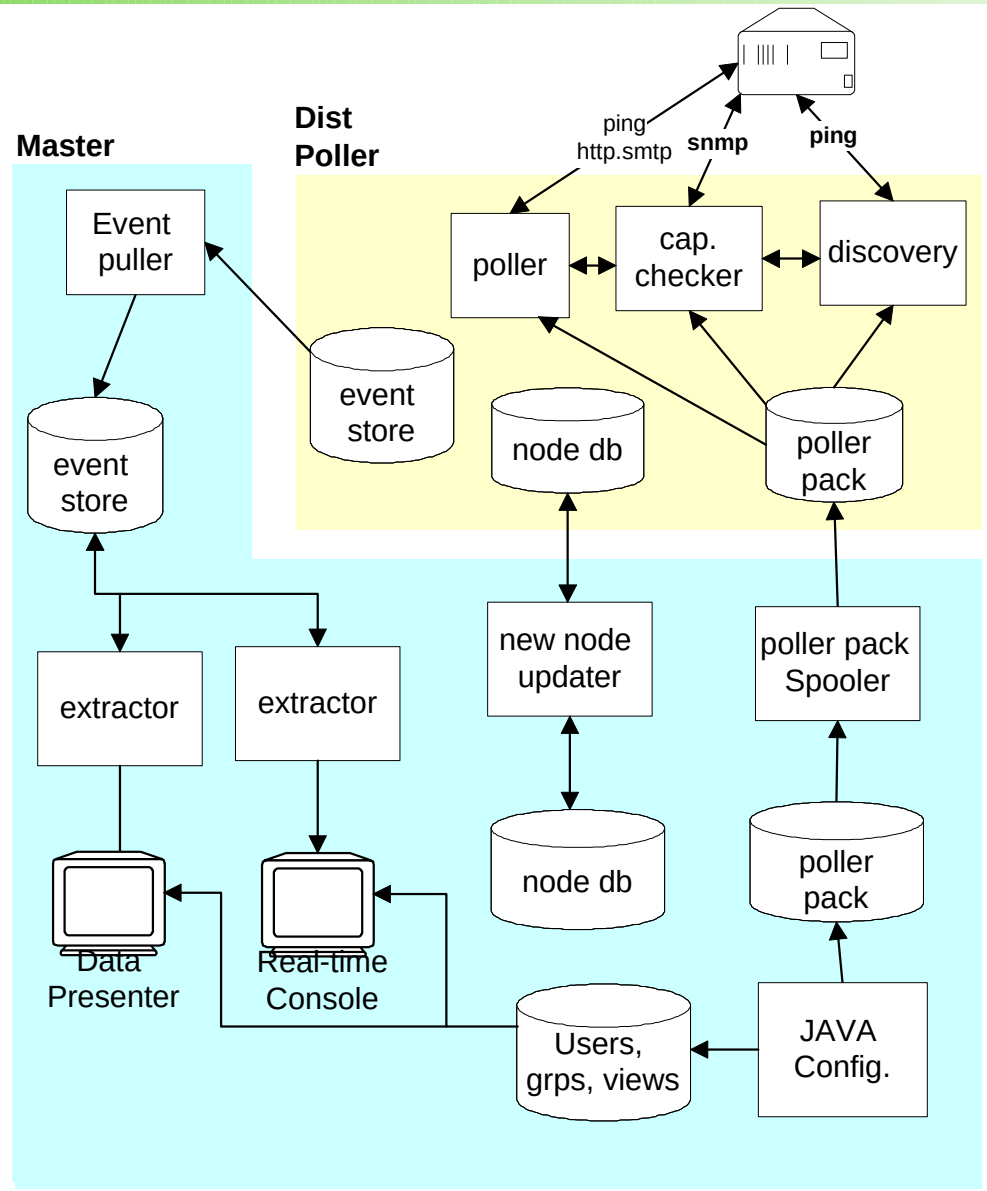
- Several very large network element providers have approached us about partnering.

Bluebird Project Overview

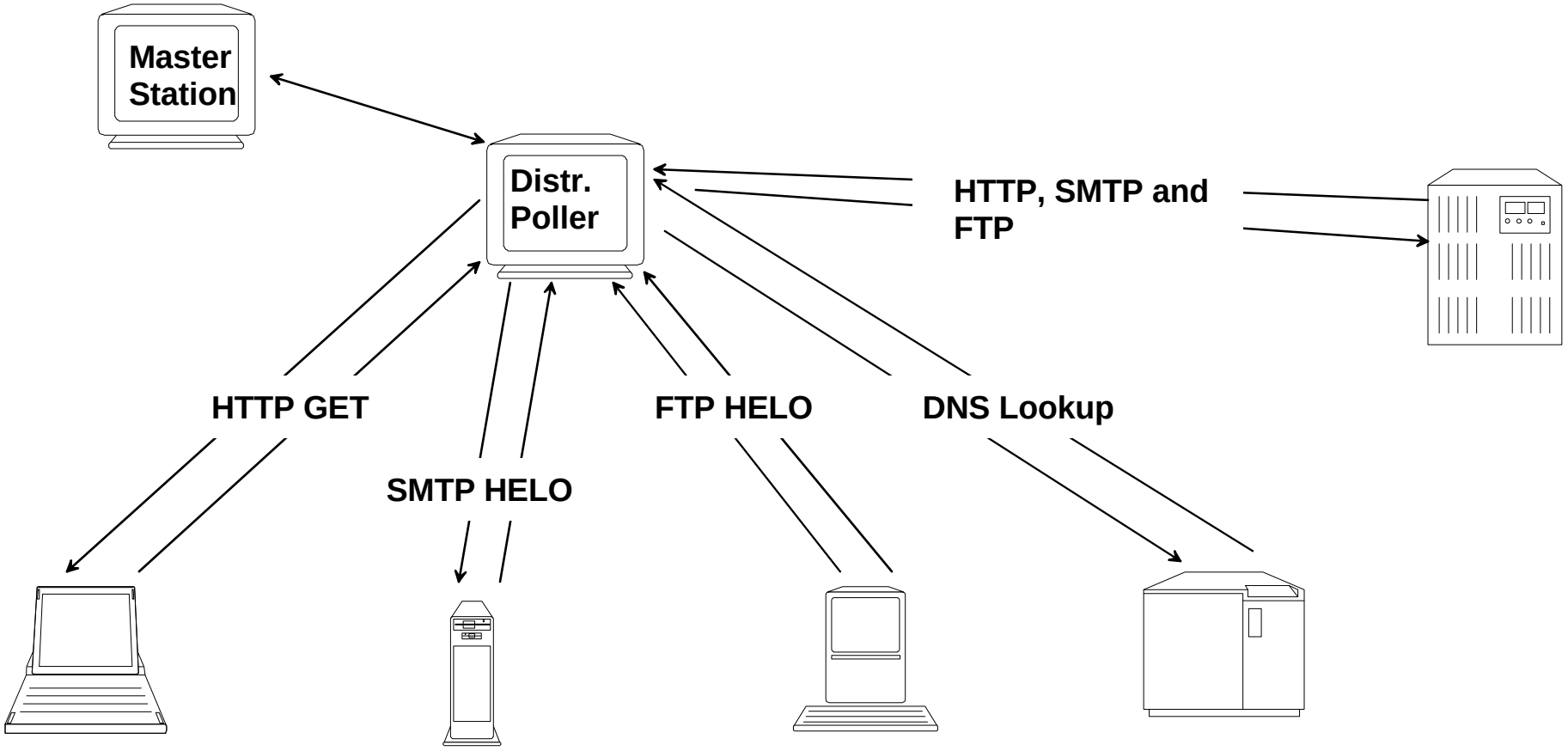
Incorporate the following functional areas into the first release:

- **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

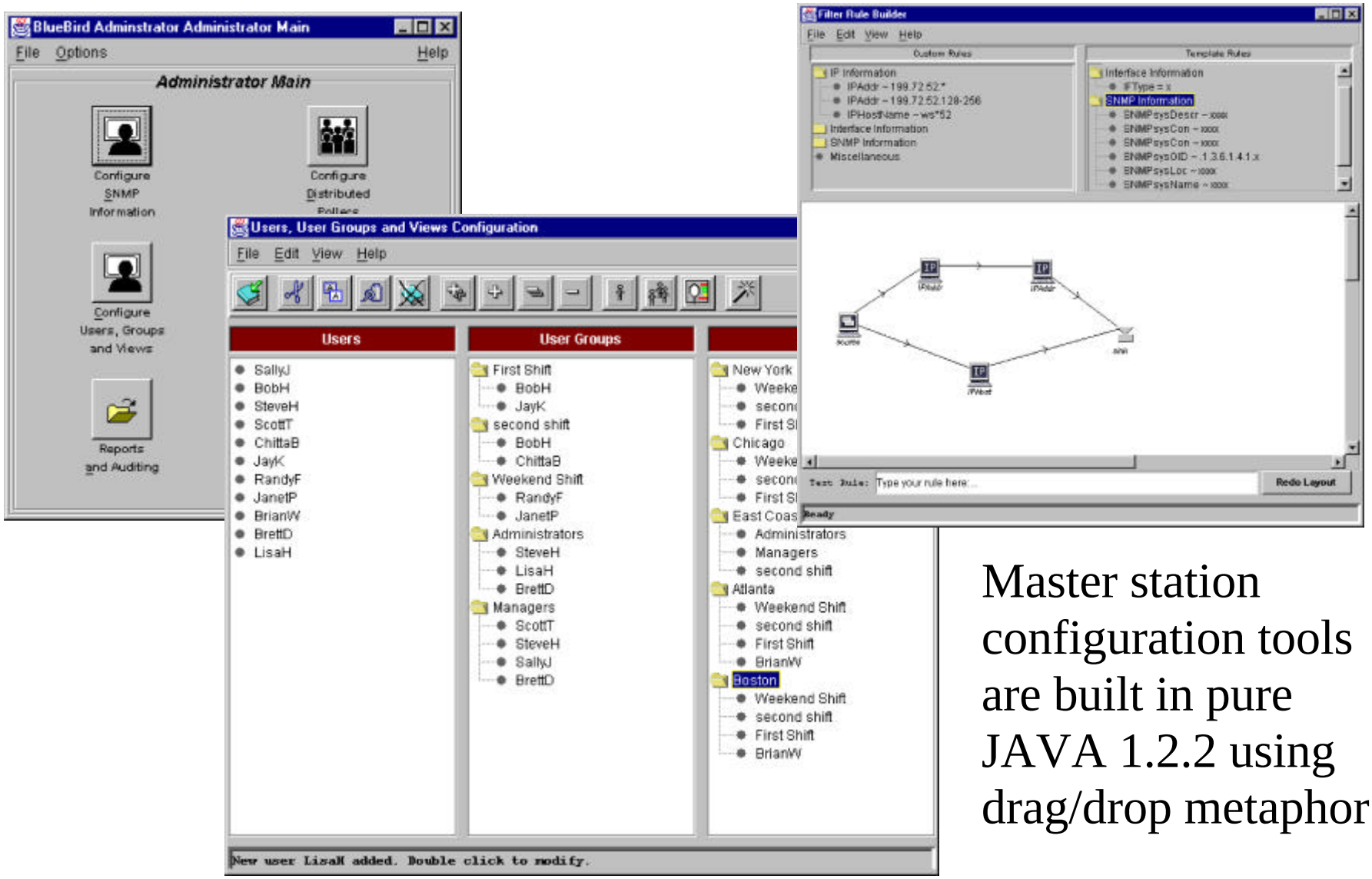


Synthetic Transactions



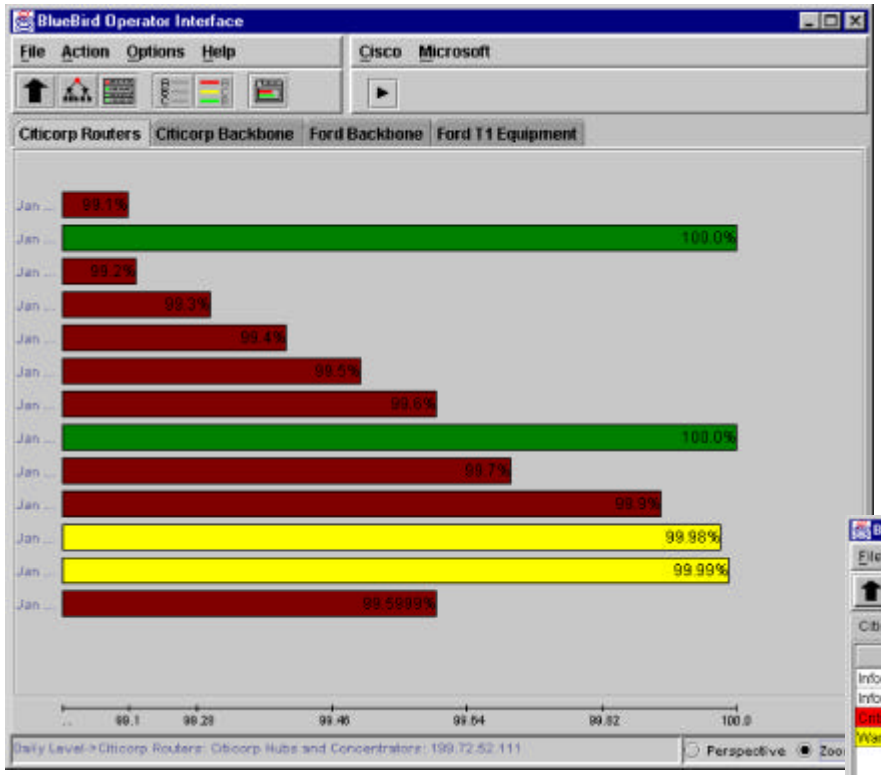
Synthetic transactions test a service on a network device for service availability. Service tests are configured using rules.

JAVA Configuration:



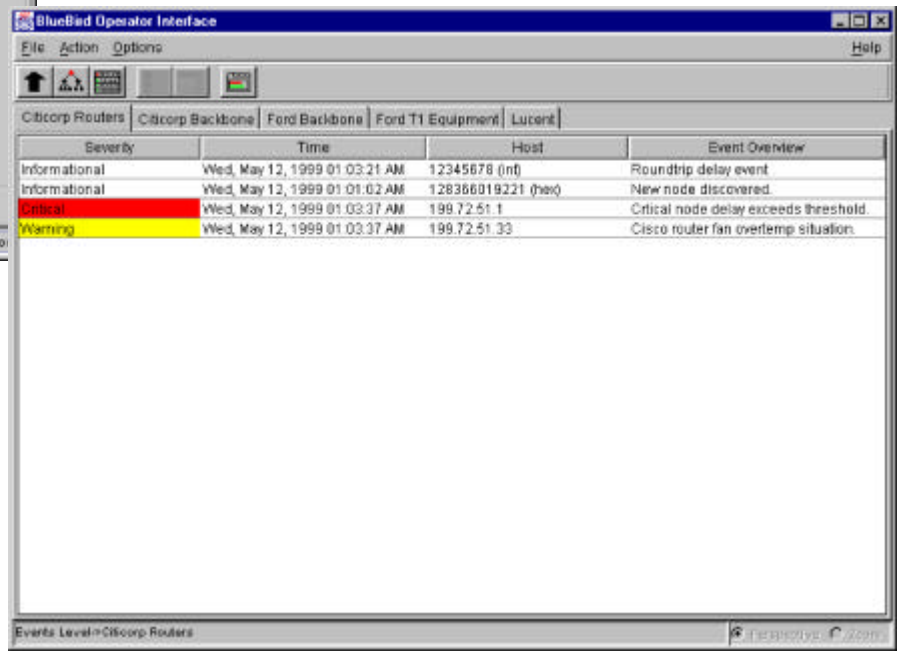
Master station configuration tools are built in pure JAVA 1.2.2 using drag/drop metaphor.

User Presentation:



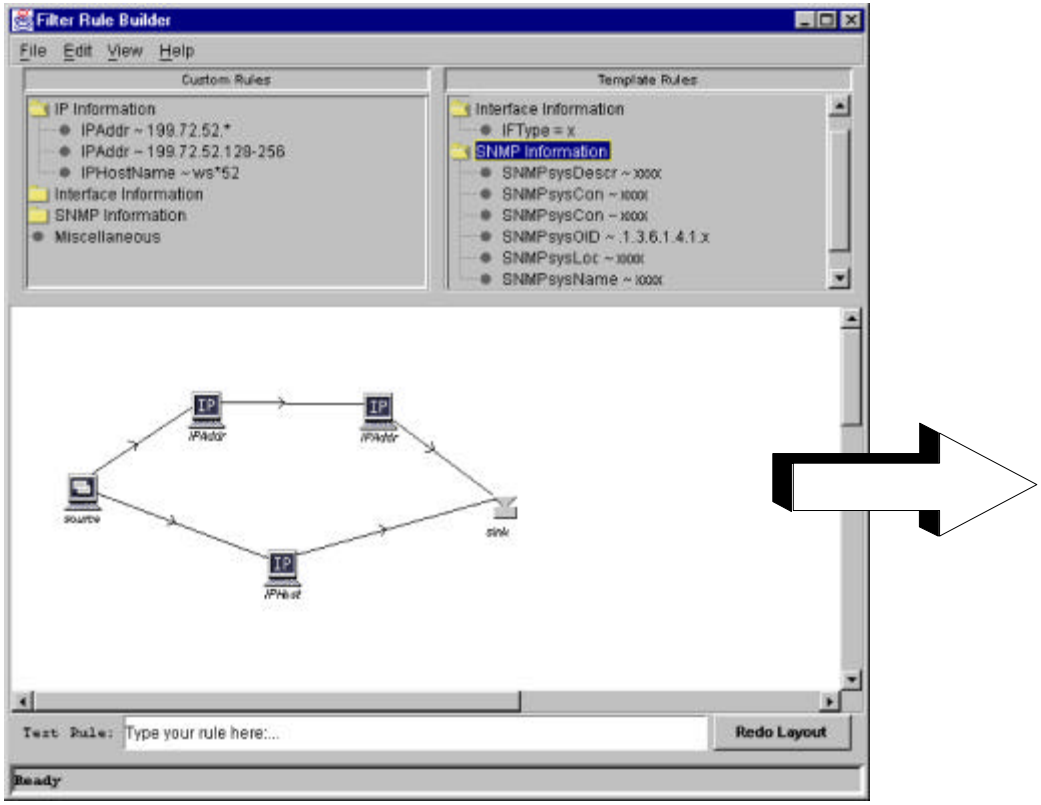
Bluebird eschews the topological view of the network for a service level view.

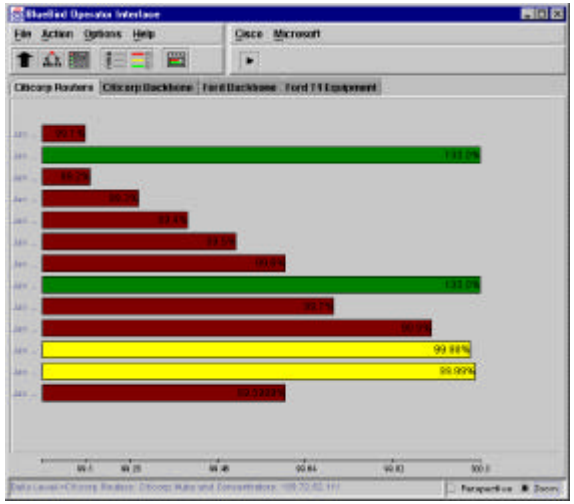
Devices are grouped into business views and categories. Drilling down takes the user to the problem.



Business Views

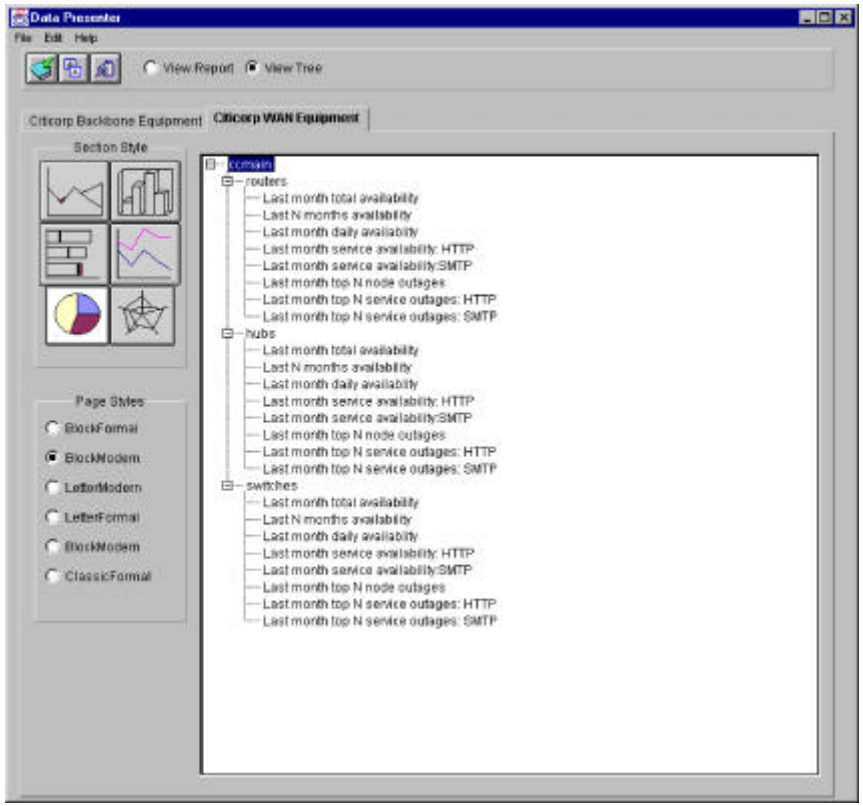
A user's view of the network is determined by rules established by the administrator. If a device matches the rule, it is included in the business view.
Rules are built using the graphical rule builder.



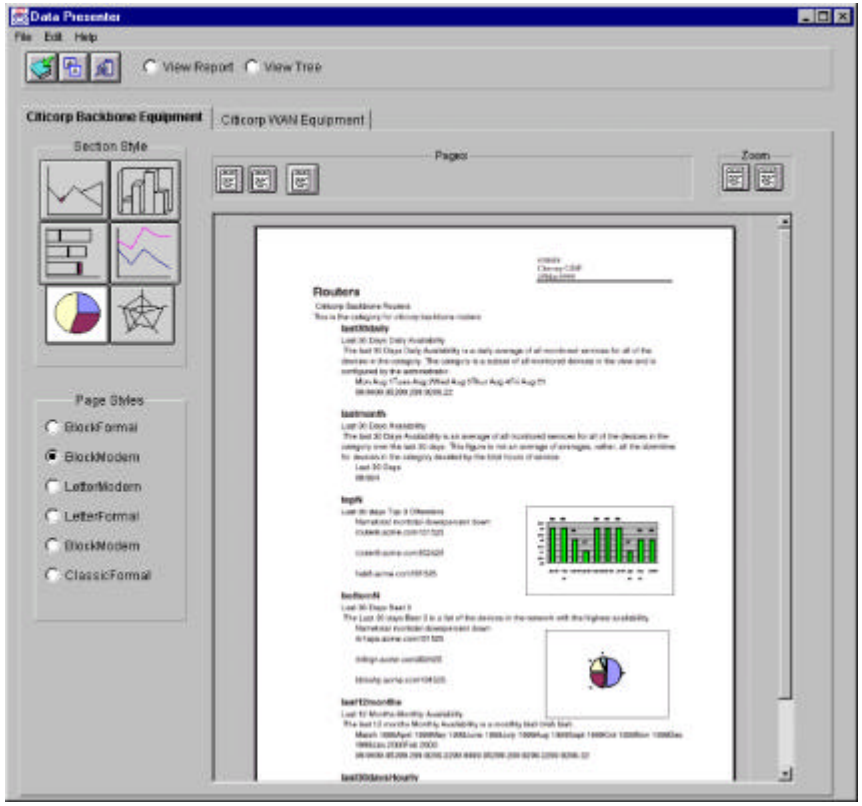


Reporting and Web Presentation

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.

Data Presenter Styles

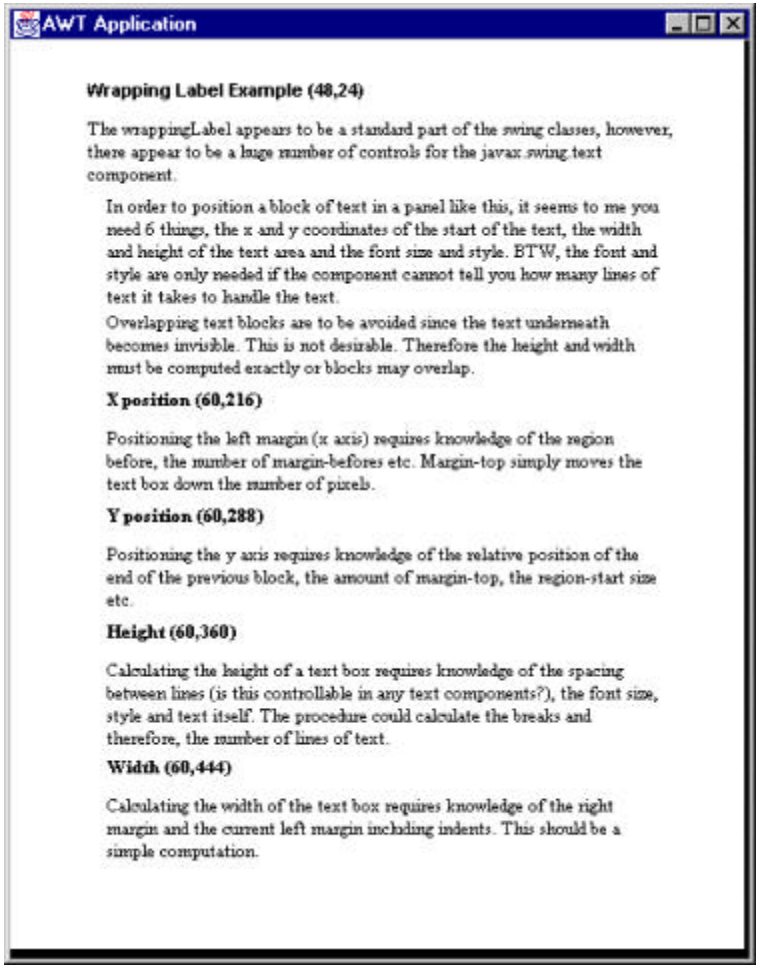
XML

```
<?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>
```

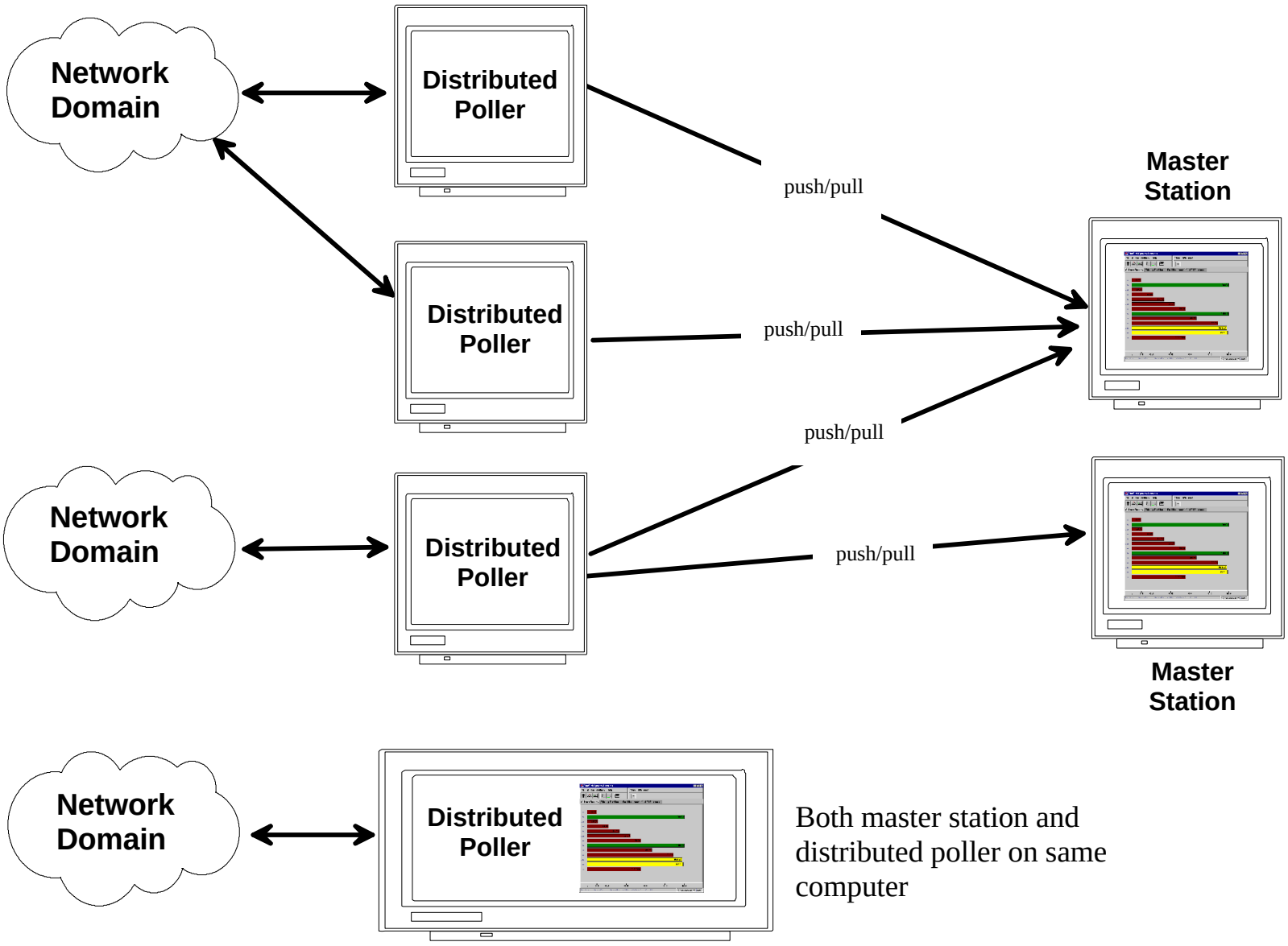
XSL

```
 <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>
                  <!-- C O V E R   P A G E   M A S T E R -->
                          <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>
                  <!--  R E S T   P A G E   M A S T E R -->
                          <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"
```

Report



Distributed Architecture



Both master station and distributed poller on same computer

Integration Points

```
...
<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>
```

Licensing

Unmodified Regular GPL

The Future

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

You Can Help!

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