System Performance Dashboard
Use Cases

Business Analyst: Mark Whitney
Oct. 18, 2009

Table of Contents

Introduction

A prototype and use cases have been developed for a dashboard that allows system administrators to view a range of performance data about a system of 15 servers.

The dashboard is intended to supply current and historical data, and warn system administrators when servers are approaching high loads that can lead to unscheduled shutdowns.

Other goals reflected in this prototype:

- Use of asynchronous techniques to update data frequently without page reloads.
- Placement of all critical data in one window, and use of pop-up windows only for non-critical data.
- Use of icons only for critical data, and use of simple icons that can be visually processed instantly.

The prototype presented in this report is intended for a screen size of 1024 pixels x 768 pixels. Beyond modern operating systems and web browsers, the one notable system requirement is having Javascript enabled in a user's browser.

Use Case 1: View Current Status

Description: This functionality shows whether each of the system's 15 servers are up or down.

Actor(s): System administrator(s)

Preconditions:
1. The user is logged in to the system with the appropriate status and permissions.
2. User has launched the dashboard from the "applications" folder.
3. User is viewing the main dashboard window.

Post-condition: The user is viewing data in the first and second columns of the dashboard.

Primary Scenario: The user wants to see which servers currently are up and which are down.

Primary Task Flow:
1. The user scans the first and second columns of the dashboard, associating each server number with the icon next to it.

Alternate Scenario: The user is unsure of an icon's meaning.

Alternate Task Flow:
1. The user scans the first and second columns of the dashboard.
2. The user mouses over the icon in the status column, and information in the title attribute clarifies whether the icon means server is up or down.

Technical Specs: **AJAX**
- A script checks whether a server is replying to queries, and triggers display of the appropriate "up" or "down" icon in the second column.
- The script updates the information every five seconds, and stores the results in a database.
- A script detects the user mousing over the icon in the status column, and triggers display of information in the title attribute.
- A script detects if the user clicks the Close button and triggers unloading of the main window.

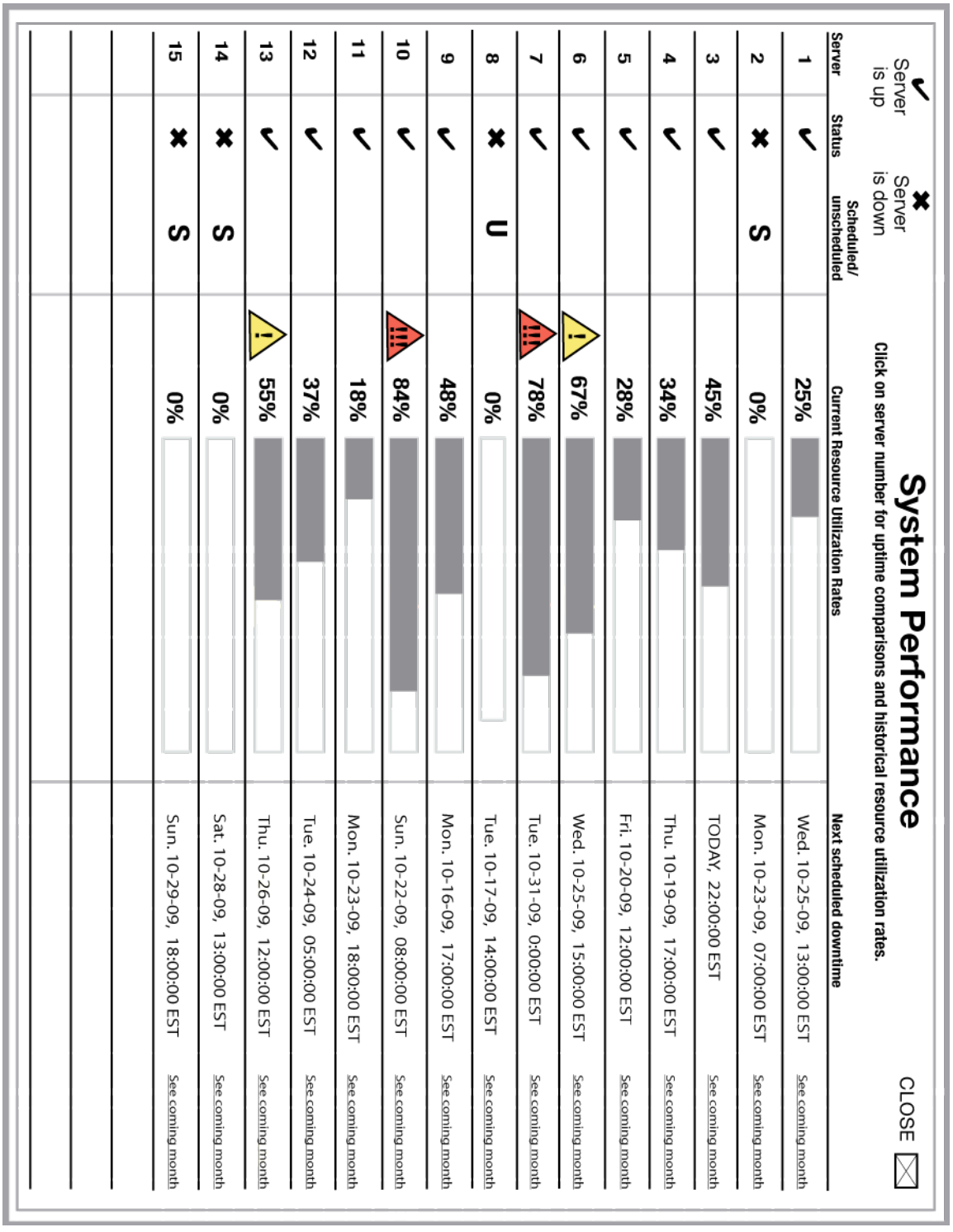
Display

- A graphic checkmark icon will appear if a server is up.
- A graphic "x" icon will appear if a server is down.
- If a server changes from up to down or from down to up, the status icon that has been showing disappears quickly, and the new icon showing the updated status fades in from low opacity to full opacity.
- A Close button is located in the upper right-hand corner of the pop-up window.

Title attribute values

- Each checkmark icon in the status column corresponds to only one server, and has a title attribute value of "Server 1 is up," "Server 2 is up," etc.
- Each "x" icon in the status column corresponds to only one server, and has a title attribute value of "Server 1 is down," "Server 2 is down," etc.

Figure 1 – Main Dashboard window



Use Case 2: View Whether Server is Scheduled to be Down

Description: This functionality shows whether a server that currently is down is scheduled to be down.

Actor(s): System administrator(s)

Preconditions:
1. The user is logged in to the system with the appropriate status and permissions.
2. User has launched the dashboard from the "applications" folder.
3. User is viewing the main dashboard screen.

Post-condition: The user is viewing data in the third column.

Primary Scenario: The user wants to see whether a server that's down is scheduled to be down.

Primary Task Flow:
1. The user scans the first, second and third columns.
2. User associates a checkmark icon or an "x" icon in the second column with the text readout "U" (for unscheduled) or "S" (for scheduled) in the third column.

Technical Specs: **AJAX**

• When the script that checks whether a server is replying to queries finds a server that is not doing so and reports the server as down, it triggers a script that searches a database of scheduled downtimes.

If the current time does not fall within the time range listed as the next scheduled downtime, the script triggers display of the "U" value.

If the current time falls within the time range listed as the next scheduled downtime, the script triggers display of the "S" value.

Use Case 3: View Current Resource Utilization Rate for Each Server

Description: This functionality displays the current Resource Utilization Rate for each server.

Actor(s): System administrator(s)

Preconditions:
1. The user is logged in to the system with the appropriate status and permissions.
2. User has launched the dashboard from the "applications" folder.
3. User is viewing the main dashboard screen.

Post-condition: The user is viewing data in the fourth column.

Primary Scenario: The user wants to see the current Resource Utilization Rate of one or more servers.

Primary Task Flow:
1. The user scans the third column of the dashboard.
2. The user sees a numerical percentage, followed by a bullet graph that corresponds to the percentage rate, possibly preceded by a warning or critical warning icon if the current Resource Utilization Rate surpasses certain levels.

Alternate Scenario: The user is unsure of the meaning of the warning and critical warning icons.

Alternate Task Flow:
1. The user scans the third column of the dashboard.
2. User sees a numerical percentage, a warning or critical warning icon and a bullet graph.
3. User mouses over the icon.
4. Information in the icon's title attribute explains its meaning.

Technical Specs: **AJAX**
- A script checks current server load every five seconds, and triggers updates of the percentage values and bullet graphs.
- If the script detects a server load of 50%-74%, it triggers display of the warning icon.

- If the script detects a server load of 75%-100%, it triggers display of the critical warning icon.
- A script detects the user mousing over the icon, and triggers display of the title attribute value.

Display

- Text field for the numerical percentage can accommodate three digits and a percent symbol.
- Bullet graphs are rendered as gray bars against white backgrounds, with gray borders outlining the graphs
- The warning icon consists of the triangular international warning sign, filled with yellow and containing one exclamation point.
- The critical warning icon consists of the triangular international warning sign, filled with red and containing three exclamation points.
- As a percentage value is updated, the corresponding number and bullet graph disappear instantly, and the new number and bullet graph fade in from low opacity to full opacity.
- As a server enters one of the warning ranges and its updated percentage value and bullet graph fade in, the appropriate warning icon appears instantly at full opacity.

Title attribute values

- For the warning icon: "Warning: Server 1 current Resource Utilization Rate is 50%-74%."
- For the critical warning icon: "Critical: Server 1 current Resource Utilization Rate is 75%-100%."

Use Case 4: View Next Scheduled Downtime

Description: This functionality shows each server's next scheduled downtime.

Actor(s): System administrator(s)

Preconditions:
1. The user is logged in to the system with the appropriate status and permissions.
2. User has launched the dashboard from the "applications" folder.
3. User is viewing the main dashboard screen.

Post-condition: The user is viewing data in the fourth column.

Primary Scenario: User wants to know the next scheduled downtime for a particular server.

Primary Task Flow:
1. The user scans the fourth column of the dashboard.
2. The user sees a text display of the date and time of the next scheduled downtime.

Technical Specs:

AJAX
- A script checks a database of scheduled downtimes and triggers display of a text readout of the date and time of the next scheduled shutdown.
- Data is updated every six hours.

Display
- Date values are expressed using the abbreviation for the day, followed by the date in the style mm-dd-yy, followed by a comma, a word space and the time expressed in the style hours:minutes:seconds, followed by the letters "EST" for Eastern Standard Time, followed by the hypertext link "See coming month."
- If the date of the next scheduled downtime is the same as the current date, the word "Today" is substituted for the date value.
- When the values are updated, the text that has been showing disappears instantly, and the updated text fades in from low opacity to full opacity.

Use Case 5: Obtain Schedule of Downtimes

Description: This functionality loads a calendar-style schedule for each server's scheduled downtimes over the coming month.

Actor(s): System administrator(s)

Preconditions:
1. The user is logged in to the system with the appropriate status and permissions.
2. User has launched the dashboard from the "applications" folder.
3. User is viewing the main dashboard window.

Post-condition: The user is viewing the main dashboard after closing a pop-up window.

Primary Scenario: The user wants to know the schedule of upcoming planned downtimes for one or more servers.

Primary Task Flow:
1. The user scans the fourth column of the dashboard.
2. User clicks on the text "See coming month."
3. A pop-up window loads containing a schedule.
4. User views the schedule.
5. User closes the pop-up window.

Technical Specs:

AJAX
- A script detects the user's click on the text "See coming month" and loads a pop-up window containing the corresponding server's downtime schedule.
- A script checks the database for downtimes that are scheduled to occur over the next 28 days, and triggers display of the time and planned duration of each shutdown.
- A script will detect the user clicking in the Close box and unload the pop-up window.

Display
- A 28-day calendar grid with the current date as the first date.
- Time and duration values appear within the box(es) containing the correct date(s).

- Time values are expressed in the style hours:minutes:seconds, followed by the letters "EST" for Eastern Standard Time.
- Duration values are expressed in the style hours:minutes:seconds.
- The pop-up window does not block the user's view of the first three columns of the main window.
- The pop-window has a clickable "Close" box in the upper right-hand corner.

Figure 2 – Dashboard and Downtime Schedule Pop-up Window

Use Case 6: Obtain Uptime Comparisons and Historical Resource Utilization Rates

Description: This functionality loads a pop-up window containing year-to-date values for uptime percentages, unscheduled and scheduled downtimes, and historical Resource Utilization Rates.

Actor(s): System administrator(s)

Preconditions:
1. The user is logged in to the system with the appropriate status and permissions.
2. User has launched the dashboard from the "applications" folder.
3. User is viewing the main dashboard window.

Post-condition: The user is viewing the main dashboard after closing a pop-up window.

Primary Scenario: The user wants to compare performance benchmarks for the entire system and individual servers, and/or the historical Resource Utilization Rate for individual servers.

Primary Task Flow:
1. The user scans the first column.
2. User clicks on a server number in the first column.
3. A pop-up window loads.
4. User views data displayed in the pop-up window.
5. User closes the pop-up window.

Technical Specs: **AJAX**
- A script detects the click on the numeral identifying each server, and loads a pop-up window.
- A script checks a database of system performance, and triggers display of the overall system's year-to-date uptime average percentage.
- A script checks a database of system performance, and triggers display of the individual server's year-to-date uptime average percentage.
- A script checks a database of system performance, and triggers display of the year-to-date unscheduled downtime total for all servers in the system.

- A script checks a database of system performance, and triggers display of the year-to-date unscheduled downtime total for the individual server.
- A script checks a database of system performance, and triggers display of the year-to-date scheduled downtime total for all servers in the system.
- A script checks a database of system performance, and triggers display of the year-to-date scheduled downtime total for the individual server.
- A script records each server's Resource Utilization Rate at two-minute intervals, and stores the results for all servers in a database.
- A script compiles the data from the two-minute intervals and computes/renders a line graph showing individual server's historical Resource Utilization Rates.
- A script detects the user clicking in the Close box and unloads the pop-up window.

Display

- The pop-up window does not block the user's view of the first three columns of the main window.
- Year-to-date comparisons will appear in three boxes, with each box containing two textual data displays.
- The first box shows average uptime percentage for all servers and average uptime percentage for the individual server.
- The second box shows total unscheduled downtime for all servers and total unscheduled downtime for that server.
- The third box shows total scheduled downtime for all servers and total scheduled downtime for that server.
- Text readouts for average uptime percentages contain one to three digits and a percent sign.
- Text readouts for downtime totals are in the style hours:minutes:seconds.
- The Historical Resource Utilization Rate line graph will consist of: a vertical axis ranging from 0% to 100%, labeled in 10% increments; a horizontal axis of 60 minutes, labeled in increments of four minutes; a graphic curve representing the data; and a gray shaded area between the curve and the horizontal axis.
- A clickable "Close" box is located in the upper right hand corner of the pop-up window.

Figure 3 – Dashboard and Uptime Comparisons Pop-up Window