 **IT**

Services Dashboard

Version: 0.5

**Department: Healthcare Apps Management**

**Author: Nitin.Jain**

**Last saved on: 14/09/2016**

# Table of Contents

[1 Table of Contents 2](#_Toc461467144)

[2 Summary 3](#_Toc461467145)

[3 To be Design 4](#_Toc461467146)

[3.1 Level 1: Consolidated View 4](#_Toc461467147)

[3.2 Level 2: Application / Process View 6](#_Toc461467148)

[3.3 Level 3: Component View 7](#_Toc461467149)

[3.4 Information View 7](#_Toc461467150)

[4 Dashboard Components 9](#_Toc461467151)

[4.1 Level 1: Consolidated View 9](#_Toc461467152)

[4.2 Level 2: Application / Process View 9](#_Toc461467153)

[4.3 Level 3: Component View 9](#_Toc461467154)

[4.4 Information view 9](#_Toc461467155)

[4.5 News feeds (To be considered in subsequent phases) 9](#_Toc461467156)

[4.6 Update Health status against an Application / component 9](#_Toc461467157)

[4.7 Monitoring Setup request Form 10](#_Toc461467158)

[4.8 Alerts 10](#_Toc461467159)

[4.9 Mobile version 10](#_Toc461467160)

[4.10 Available monitoring components 10](#_Toc461467161)

[5 Assumptions, Constraints 12](#_Toc461467162)

[6 Version Control 13](#_Toc461467163)

# Summary

Purpose of this document to outline the requirements for monitoring dashboard to show health status of critical applications / processes / services, a single view of all services status and approach to achieve it.

There are numerous systems / services exists in IT systems landscape with detailed level of monitoring already exists to generate out alerts / reports in various formats for respective apps team to consume. However current level of monitoring processes are lacking in terms of common stakeholder views of the health status, real time alerts, standard approach to present status.

Not all monitoring procedures are generating real time alerts as well as scattered in form of various reports, scripts, mails etc. along with different formats. This is causing shortcomings like

* Lack of real time alerts causing delays in resolutions as well as in proactive actions.
* No common method to set weightage and aggregation in order to define criticality as well heath status.
* Lack of common single view of all critical service status causing limited visibility into systems statuses, gaps in communications, delays in information going out to stakeholders like Business, Service Desk, IT management.

# To be Design

This document is to propose structure of **Services Dashboard** to provide a platform to integrate various monitoring processes in a single view, define weightages and aggregations to define service criticality, standard approach to consolidate monitoring processes to show consolidated health status from underneath components.

Services Dashboard designed to work on top of all monitoring process whether through monitoring tools like SCOM, Logic Monitor etc. OR via in house developed monitoring processes to consume and show consolidated status using pre-configured services in Services Dashboard.

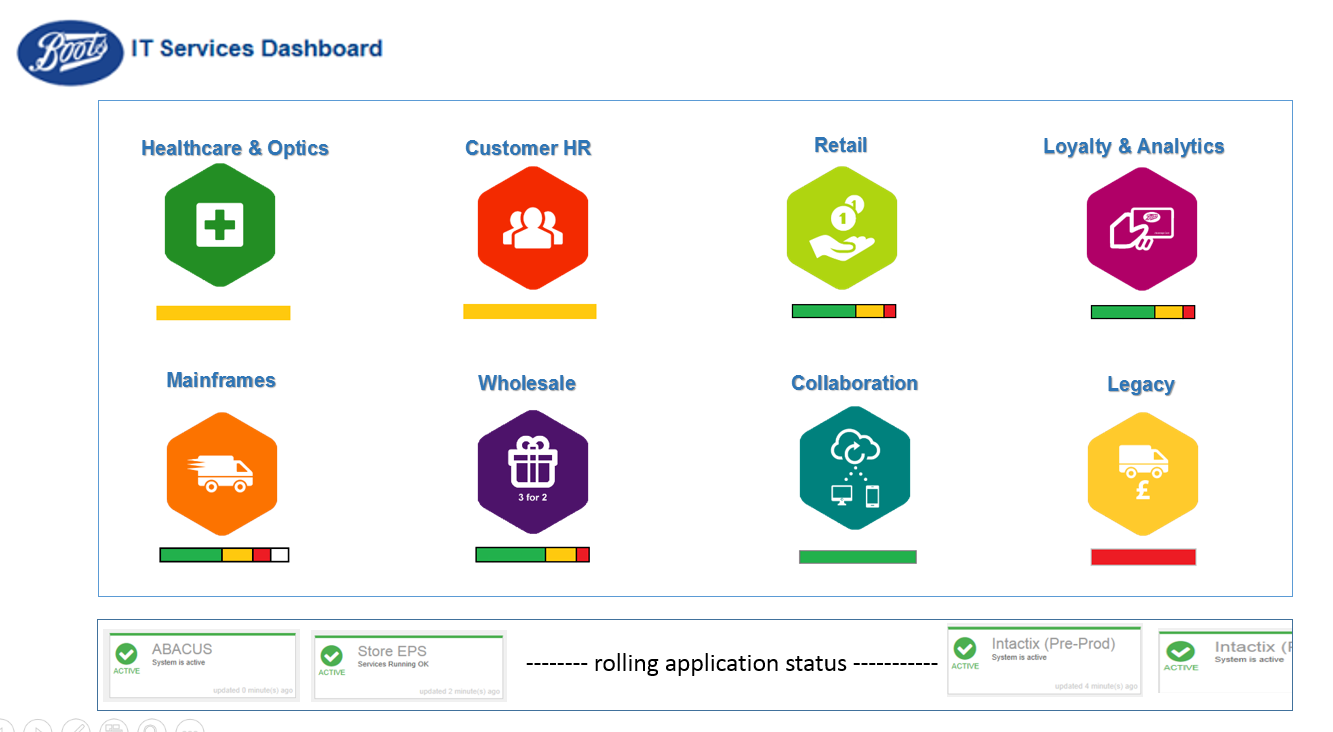
The approach is to use layered approach where highest level will show overall service statuses consolidated from underneath levels using configurable weights and aggregation methods. Integration of Services Dashboard with other monitoring processes can be at application / process or at component level to consolidate in top level single view.

Views in Services Dashboard proposed to segregate out as below:

## Level 1: Consolidated View

This would be top level view where final consolidate monitoring data will used to show service statuses as a single view.

View to be setup based on Application segregation in different portfolios to align with business perspective of IT landscape and show statuses via health bar consolidated from underneath levels. Applications / processes and components can be configured to receive integrated monitoring data or via manual methods.



**Main Area**

Main area of the page will be used to display Application portfolio along with consolidated health status bar for underlying applications / processes.

Health status bar details

|  |  |
| --- | --- |
| Health Status Bar | Detail |
|  | All applications are ok |
|  | All applications are in warning |
|  | All applications in failed status |
|  | Consolidated bar from underlying applications, length of bar would be calculated based on weight of application in CBP. By default equal will be assigned to each underlying applications, however can be configurable |
|  | Applications / components are manually monitored with no health status updated in last 24 hours |
|  | White portions specify few applications / components are manually monitored with no health status updated in last 24 hours |

**Right Pane (to be considered in subsequent phases)**

This area can be used to flash news on dashboard. There would be an application page available to target users where news feeds can be entered to show on dashboard. An automatic rolling would be available in case more news available than page can display. News feed page required to be secured using application level security.

**Bottom Pane**

This area will be used automatically to show application / process status from Level 2 view with an auto rolling to accommodate all underneath applications / processes.

Level 1: Consolidated View can be used to display on larger screens for various stakeholders like ITLT, business, service desk.

## Level 2: Application / Process View

This view will be show portfolio split into specific applications / processes along with respective health status as below –



Where application status can be aggregated using underneath component status OR can be posted directly to dashboard via available interfaces from existing monitoring procedures. Available interfaces can be -

* Web services
  + Method 1 – Post application status
  + Method 2 – Post application status with an inline detail string for information. Acceptable formats – json
  + Method 3 – Post application status with an attached file information. Acceptable formats – json,
  + Method 4 – Post component level monitoring status in an inline detail string. Application health status will be aggregated from component health. Acceptable formats – json
  + Method 5 – Post component level monitoring status in an attached file. Application health status will be aggregated from component health. Acceptable formats – json,
* Monitoring Files in shared area (future version)
  + Method 1 – Create file for application status
  + Method 2 – Create file with application status along with information data to show when an application is clicked. Acceptable formats –,json, pipe delimited
  + Method 3 – Create file for component level monitoring status. Application health status will be aggregated based on component level health. Acceptable formats –json, ,pipe delimited

-Component level monitoring data will be used to show component health status on next level

Define format for inline strings – json?

Define format for attached files – json,?

Define format for shared folder files - json, ,?

-Static information data will be used to show information when there is no component level monitoring configured application / application service -

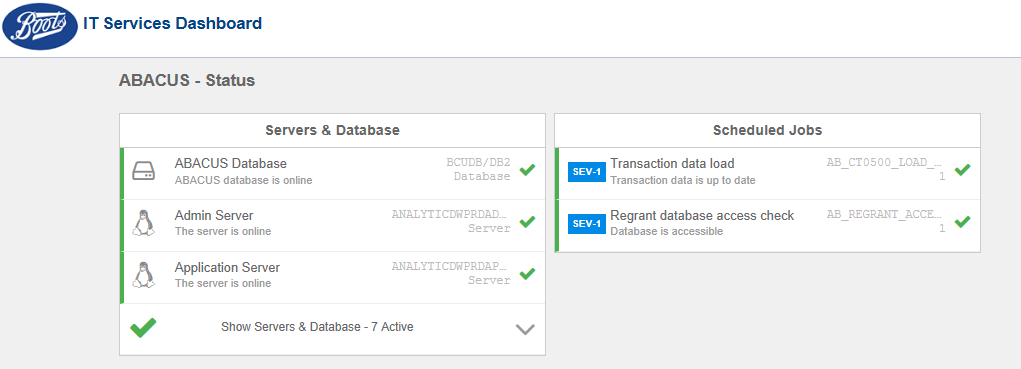
Define format for inline strings – json?

Define format for attached files – json, ,?

Define format for shared folder files for accepting application status and information –, json,?

## Level 3: Component View

This level will be used to component level health status if configured under an application / process.



This level will be used to show health status of components where monitoring data can be received via web services, shared folder files or manual methods.

Monitoring data can be posted on preconfigured components similar to application level integration. Available methods can be

**Web services**

* + Method 1 – Post component health status
  + Method 2 – Post component status with an inline detail string for information. Acceptable formats – json, Method 3 – Post component status with an attached file for information. Acceptable formats – json,

**Monitoring Files in shared area (future version)**

* + Method 1 – Create file for component status
  + Method 2 – Create file with component status along with information data to show when a component is clicked. Acceptable formats –, json,

-Static information data will be used to show information when a component is selected -

Define format for inline strings – json,?

Define format for attached files – json, ,?

Define format for shared folder files for accepting application status and information –, json,?

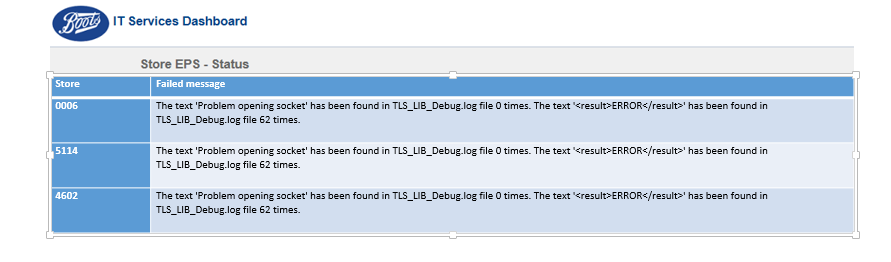
**Component Configuration:**

There are role based methods available to configure components under an application in different ways –

* **Reflex** – This role can be used to assign equal priority to all primary components or can be configured to setup low precedence on secondary components so that secondary components can reflect warnings only to the applications on failures.
* **Dimension** – This role can be used to configure components to define health status only after certain iterations of monitoring data is received.
* **Cluster** – This role can be used to group components together to reflect health status on application only when all components under a group failed or in warnings.

## Information View

A pop up page is required to show detailed information collected for an application / process OR component



# Dashboard Components

## Level 1: Consolidated View

A new page to show consolidated view of all applications, regular news feeds and all application status (auto rolling) is required to incorporate in existing dashboard. This would be first level of view and can be used to display on screens.

## Level 2: Application / Process View

Existing application level page to be modified as Level 2 view page to show application split and health status for the selected CBP from level 1 view.

## Level 3: Component View

Existing component level page can be used.

## Information view

A new page to show information data for the selected application or component.

## News feeds (To be considered in subsequent phases)

A new page / form to add news to dashboard to display in news feed section. Suggested fields would be

* **News Heading**
* **Details**
* **Start Date and Time**
* **End Date and Time**
* **Checkbox to enable / disable**

Function to update existing news via selection from a list.

Page to be secure via application maintained authorization groups along with AD integration for authentications.

## Update Health status against an Application / component

A new page / form to update Health status against an Application / Component. This would be used as an alternative to update dashboard till monitoring of specific applications / services integrated into Dashboard. Suggested fields:

* **Portfolio Select**
* **Application Select**
* **All Components in a table**
* **Update Status (Fail, Warning, Success)**

Page to be secure via application maintained authorization groups along with AD integration for authentications.

## Monitoring Setup request Form

A new page form to request configuration setup for new applications / components under a CBP. Following details would be required

**Portfolio: Name**

**Application / Process:** *Name, (new / existing), (Application / Process), (Integration Method: Webservice, Shared folder, Manual), (Integration Format: jSon, XML, CSV, Pipe Delimited), (Integration Data: Information / Component Level monitoring)*

**Alerts: Mail group, MSP group, Incident CIs.**

**Components:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Display text** | **Actual name** | **Type** | **Grouping?** | **Monitoring Frequency** | **Priority order (display)** | **Primary / Secondary** | **Integration method** | **Integration format** |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

**Available Type are:**

Application Server (Windows / UNIX / Linux / IBM 4690)

Database Server (Oracle, SQL, DB2)

Windows Services

UNIX background process

Windows Scheduled task

UNIX Jobs

Database scheduled jobs

Above could be built as an application page if schedule permits or can be used on a paper form.

## Alerts

It is suggested to enhance dashboard application to generate email alerts as well as my service portal incidents using pre-configured data around target mailboxes and Incident CIs. Expected to generate alert for every failure and warning situation using appropriate severities.

## Mobile version

Suggestion (future version) is to develop a mobile version of the consolidated view so that real time statuses can be seen on mobiles / other portable devices.

## Available monitoring components

Following is a list of existing working components used for monitoring in loyalty area and can be extended further to setup basic level of monitoring in other areas.

* Server / device availability check – using pings.
* Database availability checks – DB2, SQL, Oracle (under development)
* Database count checks to match against thresholds – DB2, SQL, Oracle (Under development)
* UNIX background process check
* UNIX file monitoring – check strings

Future development plans for monitoring components (if required)

* Windows file monitoring – check strings
* Windows services check
* Windows Scheduled task check
* Windows background process check

# Assumptions, Constraints

**Assumptions**

* Application management team required to provide configuration data on applications / processes and components in required format.
* Application level / Component level monitoring setup would be the responsibility of respective application management teams.
* Application management teams would require to consolidate data from existing monitoring procedures and post to dashboard via available methods in acceptable formats.
* Respective application management team required to post status against manually monitored applications / components with a frequency of at least 24 hours.
* Dashboard team (currently Customer Loyalty application management team) would responsible for all configurations, user accounts, BAU support of the dashboard application.
* Existing monitoring and alerting processes will continue till dashboard is developed to generate alerts via mails, my service portal incidents.

**Constraints**

* Health status for manually monitored applications / processes and component may not be available on holidays so may show some blanks in consolidated view.

# Version Control

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version Number** | **Name** | **Date** | **Items Changed** | **Next Review Date** |
| 0.1 | Nitin Jain | 31/08/2016 | First Draft |  |
| 0.2 | Nitin Jain | 02/09/2016 | Added “Available Monitoring Components” |  |
| 0.3 | Nitin Jain | 06/09/2016 | Revised following internal review |  |
| 0.4 | Nitin Jain | 12/09/2016 | Revised following review with apps managers |  |
| 0.5 | Nitin Jain | 14/09/2016 | Marked shared folder interfaces to be considered in later versions |  |
|  |  |  |  |  |