Advanced Data Analysis Environment with PI System on MHPS-TOMONI

Hiroyasu Ishigaki
Leandro Hideo Iha
Agenda

1. MHPS-TOMONI Digital Solutions
2. Interactive Advanced Data Analysis
3. Demo
4. Conclusion
1. MHPS-TOMONI Digital Solutions
Our Background

MHPS At a glance

“Our Mitsubishi Heavy Industries x Hitachi”
Power plant equipment OEM / EPC contractor
The Prove of Engineering Excellence

2,000+ patents

Headoffices in 17 countries with 48 group companies

1,000 Gas turbines,
2,600 Steam turbines
5,600 Boilers made

20,000+ staffs

Over a century of experience in turbines & Boilers.

- Service branches
- Remote Monitoring Centers
- Manufacturing sites

120+ years of solid background
MHPS-TOMONITM

A suite of digital solutions driven by user needs fueled by data analytics to make power plants digital.
MHPS-TOMONI™

Bring in software experts to provide the best solution based on customer needs and priorities.
MHPS-TOMONI™ covers the entire thermal power plant, regardless of fuel and original equipment manufacturer, and is being steadily deployed around the world.
MHPS-TOMONI: Roadmap

Autonomous Operation

Advanced O&M

O&M Support

Monitoring

Optimize the overall fleet portfolio

Remote Operation

Optimized Performance

Extended Outage Intervals

Shorter Outages

Higher Reliability

Global Service Center (Philippines)
Remote monitoring and O&M support

Predictive Analytics by diagnostics of motor current

EMS (Energy Management System)

Big Data

monitoring

Energy supply

Demand forecast

optimization

Advanced control

Automated Boiler Combustion Tuning

Remote Operation

Optimized Performance

Extended Outage Intervals

Shorter Outages

Higher Reliability

Global Service Center (Philippines)
Remote monitoring and O&M support

Predictive Analytics by diagnostics of motor current

EMS (Energy Management System)

Big Data

monitoring

Energy supply

Demand forecast

optimization

Advanced control

Automated Boiler Combustion Tuning

Remote Operation

Optimized Performance

Extended Outage Intervals

Shorter Outages

Higher Reliability

Global Service Center (Philippines)
Remote monitoring and O&M support

Predictive Analytics by diagnostics of motor current

EMS (Energy Management System)

Big Data

monitoring

Energy supply

Demand forecast

optimization

Advanced control

Automated Boiler Combustion Tuning

Remote Operation

Optimized Performance

Extended Outage Intervals

Shorter Outages

Higher Reliability

Global Service Center (Philippines)
Remote monitoring and O&M support

Predictive Analytics by diagnostics of motor current

EMS (Energy Management System)

Big Data

monitoring

Energy supply

Demand forecast

optimization

Advanced control

Automated Boiler Combustion Tuning

Remote Operation

Optimized Performance

Extended Outage Intervals

Shorter Outages

Higher Reliability

Global Service Center (Philippines)
Remote monitoring and O&M support

Predictive Analytics by diagnostics of motor current

EMS (Energy Management System)

Big Data

monitoring

Energy supply

Demand forecast

optimization

Advanced control

Automated Boiler Combustion Tuning

Remote Operation

Optimized Performance

Extended Outage Intervals

Shorter Outages

Higher Reliability

Global Service Center (Philippines)
Remote monitoring and O&M support

Predictive Analytics by diagnostics of motor current
MHPS-TOMONI architecture

- AI CPFM
- Auto Vlv. tuning
- Boiler AI tuning
- GT AI flow modulation
- SecureGW
- Edge application
- Power plant

- KPI Analyst
- Pre Act
- Plant performance
- GT parts lifetime
- ST/GEN Predict.
- STGEN prediction
- Boiler diagnosis

- Customer application
- Internal application

- Cloud
- PI Syster
- Microsoft Azure

- RMC
- MHPS users

- Customer
- Users
MHPS-TOMONI™ applications

- **24/7 Remote monitoring center**
- **Plant performance analysis**
- **Predictive analysis**
- **KPI Analyst**
- **AI combustion tuning**
- **Boiler lifetime assessment**
- ... and more
Important PI components

Calculation / Visualization

- **PI Asset Framework**
  Handling multiple power plants with structure/templates

- **PI ProcessBook / PI DataLink**
  Data visualization, easy, rich functionality

- **PI Vision**
  Easy, Ad-hoc, mobility visualization

- **PI AF Analytics**
  Various data analysis including Heat rate, equipment efficiency
We need more powerful tools!!

**Calculation**
- R, Python
- AI/Machine learning

**Visualization**
- Web based interactive framework
- Easy to develop
2. Interactive Advanced Data Analysis
Change of Paradigm for Data Analysis Report

MHPS have shifted from Classical Data Analysis Report to Interactive Data Analysis. Communication between issue owner and data scientists is improved.

Classic Data Analysis

Problem Detection → Problem Definition → Data Acquisition → Data Analysis → Results Reporting (PDF)

Interactive Data Analysis

Problem Detection → Problem Definition + Interactive Data Analysis (HTML)
Our Tools

Consistent PI historical data is leveraged by data analyst’s tools like R and Python. Mechanism is simple. Fast GUI implementation by short code.
Connections between Data Analysis Server and outside servers are secured connections. A two-factor authentication is used to control access to TOMONI & Data Analysis Server.
Integration between Shiny Server and PI System

Shiny applications access PI data using PI Web API. Security is hardened by using Kerberos instead of plain user / password.
Shiny applications access PI data using PI Web API. Security is hardened by using Kerberos instead of plain user / password.

```r
library(shiny)
library(dygraphs)
library(piwebapi)
library(xts)

server <- shinyServer(
  function(input, output) {
    output$dygraphsPlot <- renderDygraph({
      ### Secured access to PI System ###
      piWebApiService <- piwebapi$new("https://piserver.diasysnetmation.com/piwebapi/",
        TRUE, NULL, NULL, FALSE, FALSE)
      res <- piWebApiService$data$getInterpolatedValues(paste0("pi:\\SERVERDA\", input$tagId),
        startTime = format(input$dateBegin),
        endTime = format(input$dateEnd), interval = "60s")

      ### Type formatting ###
      res$timestamp <- as.POSIXct(res$timestamp, format="%Y-%m-%dT%H:%M:%S", tz='GMT')
      res <- xts(res, order.by = res$timestamp)

      ### Plot drawing ###
      dygraph(res[,2:2]) %>% dyOptions(includeZero = TRUE) %>% dyRangeSelector()
    })
  }
)

ui <- fluidPage(
  titlePanel("Hello PI World!")
  sidebarLayout(
    sidebarPanel(
      textInput("tagId", label = "PI Tag", value = "sinusoid"),
      dateInput("dateBegin", label = "Begin", value = (Sys.Date() - 7)),
      dateInput("dateEnd", label = "End", value = Sys.Date())
    ),
    mainPanel(
      dygraphOutput("dygraphsPlot", width = "100\%", height = "640px")
    )
  )
)

### Shiny Application Ready ###
shinyApp(ui = ui, server = server, options = list(host = "0.0.0.0"))
```
Fast GUI implementation with Customized Logic - Shiny

Shiny applications access PI data using PI Web API.

Security is hardened by using Kerberos instead of plain user / password.

```r
library(shiny)
library(dygraphs)
library(piwebapi)
library(xts)

server <- shinyServer(
  function(input, output) {
    output$dygraphsPlot <- renderDygraph(
      ### Secured access to PI System ###
      piWebApiService <- piwebapi$new("https://piserver.diasysnetmation.com/piwebapi/", TRUE, NULL, NULL, FALSE, FALSE)
      res <- piWebApiService$data$getInterpolatedValues(paste0("pi:\\SERVERDA\", input$tagId),
        startTime = format(input$dateBegin),
        endTime = format(input$dateEnd), interval = "60s")

      ### Type formatting ###
      res$timestamp <- as.POSIXct(res$timestamp, format="%Y-%m-%d %H:%M:%S")
      res <- xts(res, order.by=res$timestamp, tz='GMT')

      ### Plot drawing ###
      dygraph(res[,2:2]) %>%
        dyOptions(includeZero=TRUE) %>%
        dyRangeSelector()
    })
  })

ui <- fluidPage(
  titlePanel("Hello PI World!"),
  sidebarLayout(
    sidebarPanel(
      textInput("tagId", label = "PI Tag", value = "sinusoid"),
      dateInput("dateBegin", label = "Begin", value = (Sys.Date() - 7)),
      dateInput("dateEnd", label = "End", value = Sys.Date())
    ),
    mainPanel(
      dygraphOutput("dygraphsPlot", width = "100\%", height = "640px")
    )
  )

### Shiny Application Ready ###
shinyApp(ui = ui, server = server, options = list(host = "0.0.0.0"))

piWebApiService <- piwebapi$new("https://piserver.diasysnetmation.com/piwebapi/", TRUE, NULL, NULL, FALSE, FALSE)

res <- piWebApiService$data$getInterpolatedValues(paste0("pi:\\SERVERDA\", input$tagId),
  startTime = format(input$dateBegin),
  endTime = format(input$dateEnd), interval = "60s")
```
Fast GUI implementation with Customized Logic - Shiny
Example: Comparison of Plant Performance

The user can choose and shift the sampled data to compare plant performance between before and after upgrade event occur. Easy curve fitting and external effect removal is easy using statistic model.
Video Demo

Analysis Week Range

Model 2 formula

Correction Difference

Range 1 Weeks to Compare (see ‘Analysis Week Range’ Tab)

Red and Blue zones are weeks compared in the XY graph

Range 1 Weeks to Compare (see ‘Analysis Week Range’ Tab)
In the future…

• Data analysis for all! Let’s empower our engineers with data literacy!
• Continuous training to get more out of our data.

PI Web API could…

• Implement claim based authentication for PI Web API for R/Python. (Eg.: Oauth)
• Return data frames from PI Web API with adequate data types.
• Improve caching for repeated queries.

Transformation with data and cloud
Mitsubishi Hitachi Power Systems

ADVANCED DATA ANALYSIS ENVIRONMENT WITH PI SYSTEM ON MHPS-TOMONI

CHALLENGE

Build interactive applications with advanced data analysis within a short time

- Difficult to obtain data from field
- Multiple programming languages / Multiple programmers needed

SOLUTION

Access PI System using PI Web API in R and developing web application using Shiny package

- PI System in the cloud provides real time data
- PI Web API interfaces with the programming language of your choice
- One programming language from data extraction to web application

RESULTS

Time to deploy interactive data analysis application reduced from 2 months to 2 days

- Application development needs one person instead of three (Field Engineer, Data Scientist, Web Developer)
- Improved communication between issue owner and data scientist
Hiroyasu Ishigaki
General Manager, ICT department
Mitsubishi Hitachi Power Systems, LTD.
hiroyasu_ishigaki@mhps.com

Leandro Hideo Iha
Software and Control Engineer
Mitsubishi Hitachi Power Systems, LTD.
leandrohideo_iha@mhps.com
Questions?

Please wait for the microphone

State your name & company

Please rate this session in the mobile app!

DOWNLOAD THE MOBILE APP

- Rate sessions and provide feedback
- Meet and connect with other attendees

Join the conversation and SHARE what you saw #osisoft #piworld

Search “OSIsoft” in your app store