

## GPSD

### Purpose of the project

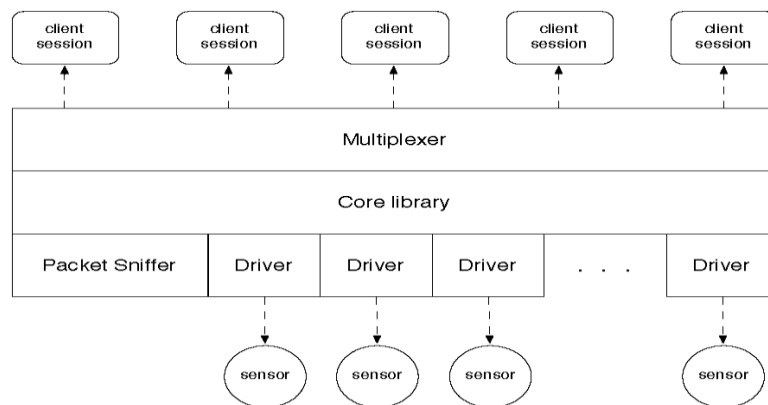
GPSD คือชุดเครื่องมือสำหรับการจัดการชุดของอุปกรณ์จีพีเอส (GPS) และเซ็นเซอร์อื่น ๆ ที่เกี่ยวข้องกับระบบการนำทาง และระบบการจับเวลา รวมถึงระบบวิทยุ AIS และ เข็มทิศดิจิทัล

GPSD ถูกพัฒนาขึ้นมา เนื่องจากแอปพลิเคชันที่มาพร้อมกับจีพีเอส และเซ็นเซอร์ที่เกี่ยวข้องกับการนำทางอื่น ๆ นั้นมีการออกแบบโดยไม่มีประสิทธิภาพ เอกสารคู่มือในการใช้งานไม่สอดคล้องกัน ประเภทของอุปกรณ์เซ็นเซอร์มีหลายรุ่นให้เลือกใช้งาน และโปรโตคอลต่าง ๆ ที่แต่ละผู้จัดทำนายสร้างขึ้นมามากมาย

ถ้าแอปพลิเคชันต้องรับมือความซับซ้อนเหล่านี้ ผลลัพธ์ที่เกิดขึ้นอาจจะทำให้มีโศกในการทำงานที่ซ้ำซ้อนมากขึ้น ซึ่งจะทำให้เพิ่มโอกาสที่ผู้ใช้จะพบเจอกับข้อบกพร่องต่าง ๆ ได้ง่าย และต้องคอยรับมือกับการเปลี่ยนแปลงของอุปกรณ์ฮาร์ดแวร์ในอนาคตอยู่เสมอ

### Architectural patterns/styles

ใช้วิธีแบ่งตาม Layers โดยแบ่งออกเป็น 4 ส่วน คือ The drivers, The packet sniffer, The core library และ The multiplexer



### The drivers

เป็นไดรเวอร์อุปกรณ์สำหรับชิปเซ็ต และเซ็นเซอร์แต่ละประเภทที่ได้รับการสนับสนุน มีหน้าที่แยกแยะเก็บข้อมูล ออกเป็นข้อมูลตำแหน่งเวลา-ความเร็วหรือสถานะ เปลี่ยนโหมดหรือ Baud rate โพรบ สำหรับประเภทย่อยของอุปกรณ์ ฯลฯ และมีปรับการควบคุมไดรเวอร์ เช่น การเปลี่ยนความเร็วอนุกรมของอุปกรณ์. อินเทอร์เฟซทั้งหมดไปยังไดรเวอร์เป็นโครงสร้างภาษา C ที่เต็มไปด้วยข้อมูลและพอยน์เตอร์เมธอด ซึ่งจำลองขึ้นมาเป็นโครงสร้างไดรเวอร์อุปกรณ์ของ Unix

The packet sniffer มีหน้าที่ในการขุดแพ็กเก็ตข้อมูลออกจากสตรีมอินพุตแบบอนุกรม โดยพื้นฐานแล้วมันเป็นเครื่องของรัฐที่คอยเฝ้าดูทุกสิ่งทุกอย่างที่ดูเหมือนหนึ่งใน 20 ประเภทแพ็กเก็ตที่เรารู้จักกันเนื่องจากอุปกรณ์สามารถสลับปลั๊กหรือเปลี่ยนโหมดได้ ประเภทของแพ็กเก็ตที่จะดึงมาจากพอร์ตอนุกรมหรือพอร์ต USB จึงไม่จำเป็นต้องได้รับการแก้ไขอย่างถาวรโดยแพ็กเก็ตแรกที่เราจะรู้จัก

The core library จัดการเซสชัน ด้วยอุปกรณ์เซ็นเซอร์หลัก ๆ คือ

- เริ่มต้นเซสชัน โดยเปิดอุปกรณ์และอ่านข้อมูลจากอุปกรณ์ ได้ตาม Baud rate และ Parity/Stopbit ผสมกัน จนกว่า Packet sniffer จะถือการซิงโครไนซ์กับประเภทแพ็กเก็ตที่เราจะรู้จัก
- การสำรวจอุปกรณ์สำหรับแพ็กเก็ต
- ปิดอุปกรณ์และปิดเซสชัน

คุณลักษณะหลักของ The core library มีหน้าที่ในการเปลี่ยนการเชื่อมต่อจีพีเอส แต่ครั้งเพื่อใช้ไครเวอร์อุปกรณ์ที่ต้องการ โดยขึ้นอยู่กับประเภทแพ็กเก็ตที่ The sniffer จะส่งกลับ ค่าที่ไม่ได้กำหนดค่าไว้ล่วงหน้าและอาจจะเปลี่ยนแปลงไปตามกาลเวลา โดยเฉพาะหากอุปกรณ์สลับไปมาระหว่างการรายงานของโปรโตคอลที่แตกต่างกัน

The multiplexer เป็นส่วนหนึ่งของ Daemon ที่จัดการเซสชันของไคลเอ็นต์และการกำหนดอุปกรณ์ มีหน้าที่ส่งรายงานไปยังไคลเอ็นต์ ยอมรับคำสั่งไคลเอ็นต์ และตอบสนองต่อการแจ้งเตือน Hotplug โดยพื้นฐานแล้วจะมีอยู่ในไฟล์ต้นฉบับหนึ่งไฟล์ gpsd.c และจะไม่เชื่อมต่อกับไครเวอร์อุปกรณ์โดยตรง

### Testability scenario

Source of Stimulus	ผู้ใช้ และผู้พัฒนา
Stimulus	คำสั่ง gpsfake
Artifacts	GPSD
Environment	การใช้งานได้อย่างสมบูรณ์ทั้งระบบ
Response	บันทึกเซ็นเซอร์ (Sensor log) พร้อมกับรายงานข้อผิดพลาด ผลการทดสอบการถดถอย (Regression-test result)
Response measure	สามารถตรวจหาข้อบกพร่องต่าง ๆ ได้ และระบุตำแหน่งที่เกิดความเสียหายได้ สามารถตรวจหาการเปลี่ยนแปลง แล้วตั้งค่าให้กับส่วนอื่น ๆ ได้

**Modifiability scenario**

Source of Stimulus	ผู้พัฒนา
Stimulus	เพิ่ม Class attribute
Artifacts	โค้ด
Environment	เวลาในการพัฒนา
Response	เพิ่ม report type ใหม่
Response measure	เพิ่ม report type ใหม่โดยไม่มีผลกระทบกับอันเดิม

**Usability scenario**

Source of Stimulus	ผู้ใช้ปลายทาง
Stimulus	ผู้ใช้ปลายทางเสียบ USB GPS เข้ากับแล็ปท็อป
Artifacts	GPSD
Environment	การทำงานปกติ
Response	รายงานสถานที่
Response measure	ส่งรายงานสถานที่ทันทีที่เสียบ USB และไม่มี muss จึงไม่มีความยุ่งยาก

Ref: <https://aosabook.org/en/gpsd.html>

## Matplotlib

### Purpose of the project

Matplotlib คือแพ็คเกจในการเขียนกราฟข้อมูล 2 มิติ และแสดงภาพที่เน้นในการทำ Visualization ข้อมูลทางวิทยาศาสตร์ วิศวกรรม และการเงิน สามารถใช้งานผ่าน Python shell หรือผ่านการเรียกใช้งานด้วยการเขียน Python Script หรือใช้งานผ่าน Application GUI ก็ได้ (GTK, Wk, Tk, Windows) สามารถ Export กราฟได้หลายรูปแบบ เช่น JPEG PNG PostScript และ SVG

Matplotlib ถูกออกแบบขึ้นมา โดยให้ผู้ใช้งานจะต้องสามารถสร้างกราฟง่ายๆ และใช้คำสั่งเพียงไม่กี่คำสั่ง หรือเพียงแค่ 1 คำสั่งเท่านั้น

### Architectural patterns/styles

Matplotlib ใช้สถาปัตยกรรมแบบ Layers ซึ่งแบ่งออกเป็น 3 Layers หลัก ๆ คือ

#### Backend Layer

จัดการงานหนักทั้งหมดผ่านการสื่อสารไปยังชุดเครื่องมือ เช่น wxPython หรือ drawing language เช่น PostScript ในเครื่อง ซึ่งเป็นเลเยอร์ที่ซับซ้อนที่สุดของไลบรารี Matplotlib

มีคลาสอินเตอร์เฟสนามธรรม 3 คลาสหลักดังนี้

- FigureCanvas: matplotlib.backend\_bases.FigureCanvasBase Canvasg เป็นรูป ที่ถูกสร้างขึ้น
- Renderer: matplotlib.backend\_bases.RendererBase เป็นคลาสนามธรรมสำหรับการจัดการ การวาด/การเรนเดอร์ โดยจะรับผิดชอบการวาดภาพใน FigureCanvas
- Event: matplotlib.backend\_bases.Event จัดการอินพุตของผู้ใช้ เช่น การกดแป้นพิมพ์และเมาส์

## Artist Layer

ช่วยให้สามารถควบคุมและปรับแต่งองค์ประกอบต่างๆ ของภาพ เช่นเดียวกับที่ศิลปินวาดภาพบนผ้าใบ เลเยอร์นี้จะประกอบด้วยออบเจกต์หลักหนึ่งรายการ โดยใช้ `Renderer` เพื่อวาดภาพบน `Canvas` จะช่วยให้คุณปรับแต่งได้มากกว่าเมื่อเปรียบเทียบกับเลเยอร์ `Scripting` และสะดวกกว่าสำหรับการพล็อตขั้นสูง โดยเฉพาะเมื่อต้องการจัดการกับตัวเลขเยอะๆ หรือ แกนหลาย ๆ แกน จะไม่เกิดการสับสนว่าตัวใดกำลังทำงานอยู่ เนื่องจากทุก ๆ `Subplot` จะถูกมอบหมายให้กับ `Artist Object`

จึงเป็นสาเหตุที่เลเยอร์นี้บางครั้งก็ถูกเรียกว่า `object-based plotting` ซึ่งมีการใช้งานบ่อยเมื่อเขียนเว็บแอปพลิเคชัน หรือแอปพลิเคชัน UI หรือเขียน `Script` เพื่อแชร์ให้กับผู้พัฒนาคนอื่น ๆ โดยทุกอย่างที่มองเห็นได้บนรูป `Matplotlib` เช่น `title` เส้น เครื่องหมายขีด รูปภาพ และอื่น ๆ เป็น `Artist instance` ซึ่งสอดคล้องกับ `Artist object` แต่ละอัน

## Scripting Layer

เป็นเลเยอร์บนสุดที่ออกแบบมาเพื่อให้ `Matplotlib` ทำงานเหมือน `MATLAB script` เป็นชุดของฟังก์ชันรูปแบบคำสั่ง และถือเป็นเลเยอร์ที่ง่ายที่สุดในการใช้งาน `Artist layer` ซึ่งเขียนโค้ดเยอะ เนื่องจากมีไว้สำหรับให้ผู้พัฒนา และเหมาะกับคนที่ต้องการที่จะทำการวิเคราะห์ข้อมูลบางส่วนแบบรวดเร็ว

นี่คือเหตุผลที่ `Matplotlib tutorial` จำนวนมาเริ่มสอนจากเลเยอร์นี้ เพราะเป็นส่วนที่ง่ายที่สุดในการเริ่มต้นและใช้งาน ซึ่งในบางครั้ง `Scripting layer plotting` ก็ถูกเรียกว่า `Procedural plotting`

### Scripting Layer

*matplotlib.pyplot*

### Artist Layer

*matplotlib.artist*

### Backend Layer

*matplotlib.backend\_bases*

Performance Scenario

Source of Stimulus	ผู้ใช้ และผู้พัฒนา
Stimulus	คำสั่ง plot กราฟ
Artifacts	Matplotlib script
Environment	การทำงานปกติ
Response	กราฟ
Response measure	ระยะเวลาตั้งแต่สั่งรันการทำงาน จน ได้ผลลัพธ์ที่เป็นรูปกราฟ

Useability Scenario

Source of Stimulus	ผู้ใช้ปลายทาง
Stimulus	ปรับย่อ/ขยายขนาดการแสดงผลของกราฟ
Artifacts	Matplotlib GUI
Environment	เวลาในการรัน
Response	กราฟที่ย่อ/ขยายขนาด
Response measure	เวลาที่ใช้ในการปรับย่อ/ขยายขนาด

**Testability Scenario**

Source of Stimulus	ผู้ใช้ปลายทาง
Stimulus	ทำการทดสอบการถดถอย (Regression Testing)
Artifacts	Matplotlib GUI
Environment	Testing framework
Response	สร้างรายงาน
Response measure	สามารถชี้แจงว่าทดสอบชุดไหนที่เกิดการข้อผิดพลาดบ้าง

Ref: <https://aosabook.org/en/matplotlib.html>

<https://medium.datadriveninvestor.com/data-visualization-with-python-matplotlib-architecture-6b05af533569>

## Joomla

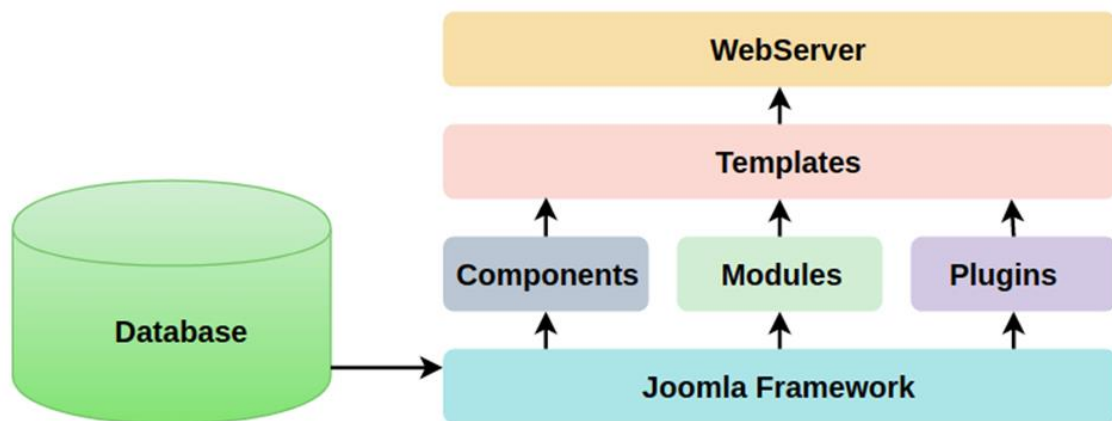
### Purpose of the project

Joomla เป็นหนึ่งในระบบจัดการเนื้อหาโอเพนซอร์ซที่ดีที่สุด (CMS) ที่สามารถใช้สร้างเว็บไซต์และแอปพลิเคชันออนไลน์ได้ เป็นเทมเพลตฟรี ขยายได้ และแยกออกเป็นเทมเพลตส่วนหน้าและส่วนหลัง (เรียกใช้จากฝั่งผู้ดูแลระบบ) มันถูกสร้างขึ้นบนเฟรมเวิร์ก model-view-controller ซึ่งสามารถใช้งานได้โดยไม่ขึ้นกับ Content Management System (CMS)

Joomla เป็นแพลตฟอร์มที่เกิดจากการโต้เถียงกันระหว่างสมาชิกของคณะกรรมการประสานงานของมูลนิธิแมมโบ้ ด้วยเหตุนี้ การเกิดของ Joomla ที่ได้สร้างเอนทิตีใหม่ที่เรียกว่า Open-Source Matter

### Architectural patterns/styles

Joomla เขียนด้วย PHP และขึ้นอยู่กับรูปแบบการออกแบบ MVC (Model-View-Controller) โดยใช้ MySQL เพื่อจัดเก็บข้อมูล มีคุณสมบัติต่าง ๆ ซึ่งทำให้ Joomla เป็นที่นิยมสำหรับเป็นระบบการจัดการเนื้อหา หรือ CMS (Content Management System)



**Joomla Architecture**



สถาปัตยกรรมของ Joomla ประกอบด้วยเลเยอร์ต่อไปนี้

### ฐานข้อมูล (Database)

ฐานข้อมูลประกอบด้วยข้อมูลโดยเฉพาะไฟล์ภาพและเอกสารที่สามารถจัดเก็บจัดการและจัดระเบียบในลักษณะเฉพาะ มันรวมถึงข้อมูลผู้ใช้เนื้อหาและข้อมูลที่จำเป็นอื่น ๆ ของไซต์ นอกจากนี้ยังมีข้อมูลการดูแลระบบเพื่อให้ผู้ดูแลระบบสามารถเข้าถึงไซต์ และจัดการได้อย่างปลอดภัย เลเยอร์ฐานข้อมูลของ Joomla เป็นหนึ่งในปัจจัยที่สำคัญที่สุดที่รับประกันความยืดหยุ่นและความเข้ากันได้สูงสุดสำหรับในสวนขยาย

### Joomla Framework

Joomla Framework ประกอบด้วยคอลเลกชันของไลบรารีซอฟต์แวร์ โอเพนซอร์สหรือแพ็คเกจ ซึ่งระบบการจัดการเนื้อหา Joomla สร้างขึ้น ไม่จำเป็นต้องติดตั้ง Joomla Framework เพื่อใช้ CMS หรือในทางกลับกัน Joomla Framework มีกลุ่มไฟล์ที่มีประโยชน์ในการสร้างทั้งเว็บและแอปพลิเคชัน Command-line มันแบ่งเฟรมเวิร์กออกเป็นแพ็คเกจโมดูลาร์เดี่ยวและยังไปกว่านั้นยังช่วยให้แต่ละแพ็คเกจพัฒนาได้ง่ายขึ้น

### Components

ส่วนประกอบต่าง ๆ เรียกว่ามินิแอปพลิเคชันซึ่งประกอบด้วยสองส่วน คือ

- o Administrator
- o Site

เมื่อใดก็ตามที่โหลดหน้าเว็บส่วนประกอบจะถูกเรียกเพื่อแสดงเนื้อหาของหน้าหลัก ส่วนผู้ดูแลระบบจะจัดการแง่มุมต่างๆ ของส่วนประกอบ และส่วนไซต์ ช่วยในการแสดงเพจเมื่อผู้เยี่ยมชมไซต์ทำการร้องขอ ส่วนประกอบเป็นที่รู้จักกันว่าเป็นหน่วยการทำงานที่สำคัญของ Joomla

## Modules

โมดูลสามารถกำหนดเป็นส่วนขยายที่ใช้ในการแสดงหน้าใน Joomla ใช้เพื่อแสดงข้อมูลใหม่จากส่วนประกอบ ซึ่งสามารถทำงานได้ด้วยตัวเองและได้รับการจัดการโดยตัวจัดการโมดูล ซึ่งเป็นส่วนประกอบที่เป็นส่วน ๆ เช่น โมดูลเข้าสู่ระบบ นอกจากนี้ยังช่วยในการแสดงเนื้อหาและรูปภาพใหม่เมื่อโมดูลเชื่อมโยงกับส่วนประกอบ Joomla

## Plugin

ปลั๊กอินสามารถอธิบายได้ว่าเป็นส่วนขยาย Joomla ที่ยืดหยุ่นและทรงพลังมาก ซึ่งใช้เพื่อขยายเฟรมเวิร์ค ปลั๊กอินเป็นรหัสไม่กึ่งรหัสที่ดำเนินการในบางโอกาสของทริกเกอร์เหตุการณ์ที่เฉพาะเจาะจง โดยทั่วไปจะใช้เพื่อจัดรูปแบบผลลัพธ์ของส่วนประกอบหรือโมดูลเมื่อมีการพัฒนาเพจ ฟังก์ชันปลั๊กอินที่เกี่ยวข้องกับเหตุการณ์มักจะดำเนินการตามลำดับเมื่อใดก็ตามที่มีเหตุการณ์ใดเหตุการณ์หนึ่งเกิดขึ้น

## Templates

เทมเพลตใช้เพื่อจัดการรูปแบบของเว็บไซต์ Joomla โดยทั่วไปมีเทมเพลตสองประเภทให้เลือก Front-end และ Back-end เทมเพลต Front-end เป็นวิธีจัดการรูปแบบของเว็บไซต์ซึ่งผู้ใช้เห็นได้ เทมเพลต Back-end ใช้เพื่อจัดการหรือควบคุมฟังก์ชันโดยผู้ดูแลระบบ เทมเพลตเป็นวิธีที่ง่ายที่สุดในการสร้างหรือกำหนดไซต์เอง ใช้เพื่อเพิ่มความยืดหยุ่นสูงสุดเพื่อให้ไซต์ของคุณดูน่าสนใจ

## Web Server

เป็นเซิร์ฟเวอร์ที่ใช้เชื่อมต่อผู้ใช้กับไซต์ มีหน้าเว็บให้กับลูกค้า HTTP (HyperText Transfer Protocol) ถูกใช้เพื่อสื่อสารระหว่างไคลเอนต์และเซิร์ฟเวอร์

**Useability Scenario**

Source of Stimulus	ผู้ดูแลเว็บไซต์ และผู้พัฒนา
Stimulus	ต้องการที่จะเรียนรู้วิธีการใช้ Joomla
Artifacts	Joomla
Environment	การทำงานปกติ
Response	ทำงานได้ตามที่ผู้ใช้ต้องการ
Response measure	เวลาที่ใช้ในการเรียนรู้

**Security Scenario**

Source of Stimulus	ผู้ใช้ และแฮ็กเกอร์
Stimulus	คำขอเข้าถึงโดยไม่ได้รับอนุญาต
Artifacts	ฐานข้อมูล Joomla
Environment	การทำงานปกติ
Response	ปฏิเสธการเข้าถึง
Response measure	จำนวนครั้งที่ป้องกันการเข้าถึงโดยไม่ได้รับอนุญาต

**Availability scenario**

Source of Stimulus	ผู้ใช้
Stimulus	มีการ Request จำนวนมาก ๆ
Artifacts	Joomla WebServer
Environment	Production
Response	ตอบสนองการ Request ได้ทัน (Inform operator)
Response measure	เปอร์เซ็นต์ความพร้อมใช้งาน

REF: <https://www.javatpoint.com/architecture-of-joomla>