

# MySQL Architecture



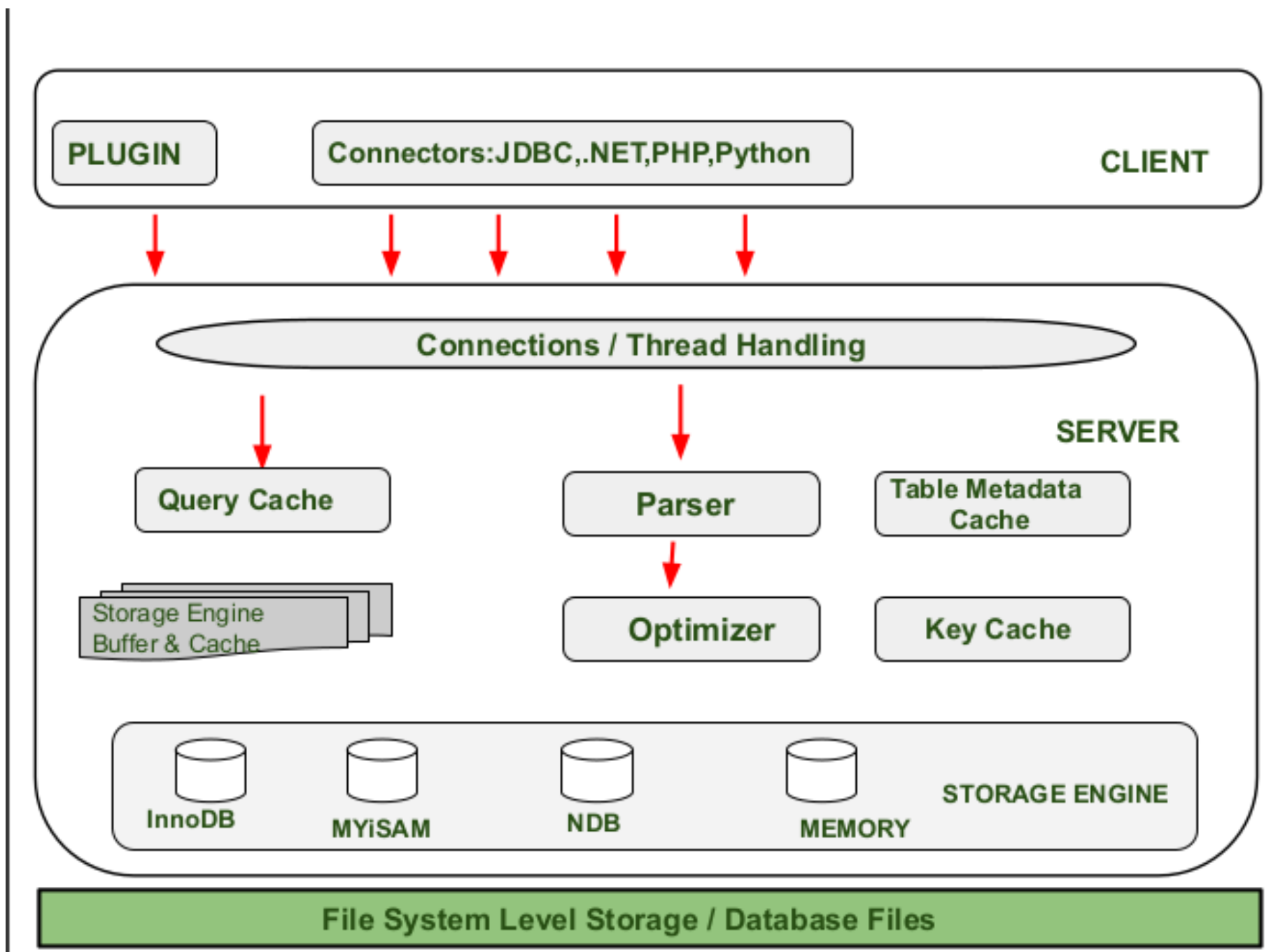
# สถาปัตยกรรมของ MySQL

เป็นฐานข้อมูลเชิงสัมพันธ์ที่มีสถาปัตยกรรมแบบเลเยอร์จะประกอบด้วยส่วนทรัพยากร server ที่อยู่ตรงกลาง เครื่องมือจัดเก็บข้อมูลที่อยู่ด้านล่างและปลาย client หรือการดำเนินการ query จะสิ้นสุดที่ด้านบนเป็นระบบฐานข้อมูลสถาปัตยกรรม 3 ชั้น

สถาปัตยกรรมของฐานข้อมูลจะอธิบายความสัมพันธ์และการโต้ตอบระหว่าง client & server และ storage

**ส่วนของผู้ให้บริการ (server)** ทำหน้าที่ บริหารจัดการระบบฐานข้อมูลคือตัว MySQL Server และเป็นที่จัดเก็บข้อมูลทั้งหมดข้อมูลที่เก็บไว้จะมีทั้งข้อมูลที่จำเป็นสำหรับการทำงานกับระบบฐานข้อมูลและข้อมูลที่เกิดจากผู้ใช้

**ส่วนของผู้ใช้บริการ (server)** คือ ผู้ใช้โดยโปรแกรมสำหรับใช้งานในส่วนนี้ได้แก่ MySQL Client, Access, Web Development Platform ต่างๆ เช่น Java, Perl, PHP, ASP



# Client Layer

เลเยอร์บนสุดใน diagram ด้านบน client จะส่งคำขอหรือคำสั่งไปยัง server ด้วยความช่วยเหลือของ client layer โดย client ส่งคำขอผ่านพร้อมรับคำสั่งหรือผ่านหน้าจอ GUI โดยใช้คำสั่งและนิพจน์ MySQL ที่ถูกต้อง หากนิพจน์และคำสั่งถูกต้องจะส่งไปยังปลายทาง server ผลลัพธ์ที่ได้จะแสดงบนหน้าจอสำหรับทุกการส่งคำสั่งที่ถูกต้องบนหน้าจอทุกครั้งที่ส่งคำสั่งผิดจะมีข้อความแสดงข้อผิดพลาดส่งกลับมาที่หน้าจอ เมื่อผู้ใช้ส่งคำขอไปยัง server ปลายทางและ server ยอมรับคำขอการเชื่อมต่อจะถูกสร้างขึ้นทันทีเพื่อให้ผู้ใช้ดำเนินการตามคำขอเพิ่มเติมได้ซึ่งอาจเรียกว่าการจัดการเชื่อมต่อและเป็นฟังก์ชันที่แสดงผลโดยฝั่ง client ของสถาปัตยกรรม

**บริการที่สำคัญของชั้น client** ได้แก่

**1. Connection Handling** การจัดการการเชื่อมต่อ : เมื่อ client ส่งคำขอไปยัง server และ server จะยอมรับคำขอและเชื่อมต่อ client เมื่อ client เชื่อมต่อกับ server ในเวลานั้น client จะได้รับเธรดของตัวเองสำหรับการเชื่อมต่อด้วยความช่วยเหลือของเธรดนี้แบบสอบถามทั้งหมดจากฝั่ง client จะถูกดำเนินการ

**2. Authentication** การรับรองความถูกต้อง : การรับรองความถูกต้องจะดำเนินการในฝั่ง server เมื่อ client เชื่อมต่อกับ server MySQL การรับรองความถูกต้องทำได้โดยใช้ชื่อผู้ใช้และรหัสผ่าน

**3. Security** ความปลอดภัย : หลังจากตรวจสอบความถูกต้องเมื่อ client เชื่อมต่อ server MySQL สำเร็จ server จะตรวจสอบว่า client ใดมีสิทธิ์ออกคำสั่งกับ server MySQL



# Server Layer

ขั้นที่ 2 ของสถาปัตยกรรม MySQL ที่รับผิดชอบการทำงานด้วยตรรกะของระบบฐานข้อมูลเป็นสมองของสถาปัตยกรรม MySQL ได้รับทุกคำขอที่ส่งโดยฝั่ง client และยังส่งคืนข้อเสนอแนะ เมื่อประมวลผลทุกคำสำหรับทุกคำขอที่ส่งโดยฝั่ง client มีการสร้างการเชื่อมต่อ การเชื่อมต่อนี้เรียกว่า เธรด

โดยฝั่ง server ส่วนท้ายของสถาปัตยกรรม server ช่วยในการจัดการทุกเธรดที่สร้างขึ้นและสิ่งนี้เรียกว่าการจัดการเธรด การจัดการเธรดเป็นฟังก์ชันที่ดำเนินการโดยที่ฝั่ง server ของสถาปัตยกรรม

## ส่วนประกอบย่อยต่างๆ ของ server MySQL ดังนี้

**1. Thread Handling :** เมื่อ client ส่งคำขอไปยัง server และ server ยอมรับคำขอและเชื่อมต่อ client แล้วหาก client เชื่อมต่อกับ server ในเวลานั้น client จะได้รับเธรดของตัวเองสำหรับการเชื่อมต่อ เธรดนี้มีให้โดยการจัดการเธรดของ server layer นอกจากนี้ query ของฝั่ง client ซึ่งดำเนินการโดยเธรดยังได้รับการจัดการโดยโมดูลการจัดการเธรด

**2. Parser** เป็นส่วนประกอบ software ประเภทหนึ่งสร้างโครงสร้างโมดูล (parse tree) ของ input ที่กำหนดก่อนแยกวิเคราะห์ คำศัพท์จะเสร็จสิ้น กล่าวคือ input ถูกแบ่งออกเป็นจำนวน tokens หลังจากข้อมูลพร้อมใช้งานในตัวแยกวิเคราะห์องค์ประกอบที่เล็กกว่าแล้วดำเนินวิเคราะห์ไวยากรณ์ การวิเคราะห์ความหมายหลังจากแยกวิเคราะห์ parse tree จะถูกสร้างขึ้นเป็น output

# Server Layer

3. **Optimizer** ทันทีกี่การแบกวิเคราะห์เสร็จสิ้นจะมีการใช้เทคนิคการเพิ่มประสิทธิภาพประเภทต่างๆ ที่ Optimizer block เทคนิคเหล่านี้อาจรวมถึงการเขียน query ใหม่ลำดับการสแกนตารางและการเลือกดัชนีที่เหมาะสมเพื่อใช้เป็นต้น
4. **Query Cache** เก็บชุดผลลัพธ์ทั้งหมดสำหรับคำสั่ง query ที่ป้อนก่อนแบกวิเคราะห์ server MySQL จะรักษาแคชแบบสอบถาม เมื่อ client เขียนแบบสอบถาม ถ้าแบบสอบถามที่เขียนโดย client เหมือนกันในแคชนั้น server จะข้ามการแบกวิเคราะห์การปรับให้เหมาะสมและแม้กระทั่งการดำเนินการ server จะแสดงผลลัพธ์จากแคช
5. **Buffer and Cache** จะเก็บข้อความค้นหาหรือปัญหาที่ผู้ใช้ถามก่อนหน้านี้ เมื่อผู้ใช้เขียนแบบสอบถามคนอื่นจะไปที่ query cache จากนั้น query cache จะตรวจสอบว่ามีแบบสอบถามหรือปัญหาเดียวกันอยู่ในแคช หากมีแบบสอบถามเดียวกันก็จะให้ผลลัพธ์โดยไม่รบกวน parser, optimizer
6. **Table Metadata Cache** แคชข้อมูล Meta เป็นพื้นที่สำรองของหน่วยความจำที่ใช้สำหรับติดตามข้อมูลบนฐานข้อมูล ดัชนีหรือวัตถุ ยิ่งฐานข้อมูลดัชนีหรือ object ที่เปิดอยู่มีจำนวนมากขึ้นขนาดแคชข้อมูล Meta ก็ยิ่งมากขึ้นเท่านั้น
7. **Key Cache** รายการดัชนีที่ระบุวัตถุในแคช โดยเฉพาะตามค่าเริ่มต้น server ชอบจะแคชเนื้อหาตามเส้นทางทรัพยากรทั้งหมดและสตริงข้อความค้นหา

# Storage Layer

ฐานข้อมูล MySQL มีกลไกการจัดเก็บข้อมูลประเภทต่างๆ ที่มีอยู่ซึ่งเป็นผลมาจากความต้องการที่แตกต่างกันของฐานข้อมูล เครื่องมือจัดเก็บข้อมูลใช้เพื่อเก็บทุกตารางที่ผู้ใช้สร้างขึ้นในระบบฐานข้อมูล ได้แก่ InnoDB, MYISAM, NDB, Memory เป็นต้น จุดสิ้นสุดการจัดเก็บอำนวยความสะดวกในการจัดเก็บและดึงข้อมูล MySQL ทั้งนี้ engine การจัดเก็บข้อมูลมี API ที่ช่วยในการดำเนินการค้นหาจากจุดสิ้นสุดของ client ของสถาปัตยกรรมเพียงแค่ส่งแถวไปมา

คอมพิวเตอร์ ทำหน้าที่เป็น client, server อาจอยู่บนเครื่องเดียวกันหรือแยกเครื่องก็ได้ทั้งนี้ขึ้นอยู่กับลักษณะการทำงานหรือการกำหนดของผู้บริหารระบบตามปกติถ้าเป็นการทำงานในลักษณะ web-based มีการใช้ฐานข้อมูลขนาดใหญ่ไว้กับ MySQL server, client มักจะอยู่บนเครื่องเดียวกัน โดยเครื่องคอมพิวเตอร์ดังกล่าวจะต้องมีทรัพยากรเพื่อการทำงาน (เช่น เนื้อที่ฮาร์ดดิสก์, RAM เป็นต้น) มากพอสมควรแต่สำหรับการทำงานจริง (real-world application) ก็มักจะแยก client, server ออกเป็นคนละเครื่องกัน เพราะสามารถรองรับงานได้ดีกว่าและมากกว่าดังนั้นผู้บริหารระบบหรือผู้กำหนดนโยบายสำหรับการทำงานเครือข่ายจะต้องคำนึงถึงเรื่องที่เกี่ยวข้องเหล่านี้ให้ดีเพื่อที่จะทำให้ระบบมีการทำงานรองรับการให้บริการแก่ผู้ใช้ได้อย่างมีประสิทธิภาพและข้อมูลมีความปลอดภัยมากที่สุด



# คุณสมบัติของ MySQL

1. ประกอบด้วย data security layer ซึ่งจะปกป้องข้อมูลจากผู้ละเมิดนอกจากนี้รหัสผ่านจะถูกเข้ารหัสใน MySQL
2. ทำตามสถาปัตยกรรม client server ที่ client ร้องขอคำสั่งน่านะ server จะสร้างผลลัพธ์ทันทีที่คำสั่งตรงกัน
3. ใช้มัลติเธรดซึ่งทำให้ปรับขนาดได้สามารถจัดการข้อมูลจำนวนเท่าใดก็ได้ขีดจำกัดขนาดไฟล์เริ่มต้นที่ 4GB แต่เราสามารถเพิ่มได้ตามความต้องการของเรา
4. มีความยืดหยุ่นสูงเนื่องจากรองรับระบบฝังตัวจำนวนมาก
5. รองรับการทำงานบนระบบปฏิบัติการต่าง
6. อนุญาตให้ทำธุรกรรมย้อนกลับ คอมมิตและกู้คืนแคช
7. ถ้ามีปัญหาเร็วไหลของหน่วยความจำต่ำซึ่งจะช่วยเพิ่มประสิทธิภาพหน่วยความจำ
8. Ver. 8.0 ให้การสนับสนุนรหัสผ่านคู่หนึ่งคือรหัสผ่านปัจจุบันและรหัสผ่านรองด้วยความช่วยเหลือของสิ่งนี้เราสามารถสร้างรหัสผ่านใหม่ได้
9. มีคุณสมบัติแบ่งพาร์ติชันซึ่งปรับปรุงประสิทธิภาพของฐานข้อมูลขนาดใหญ่