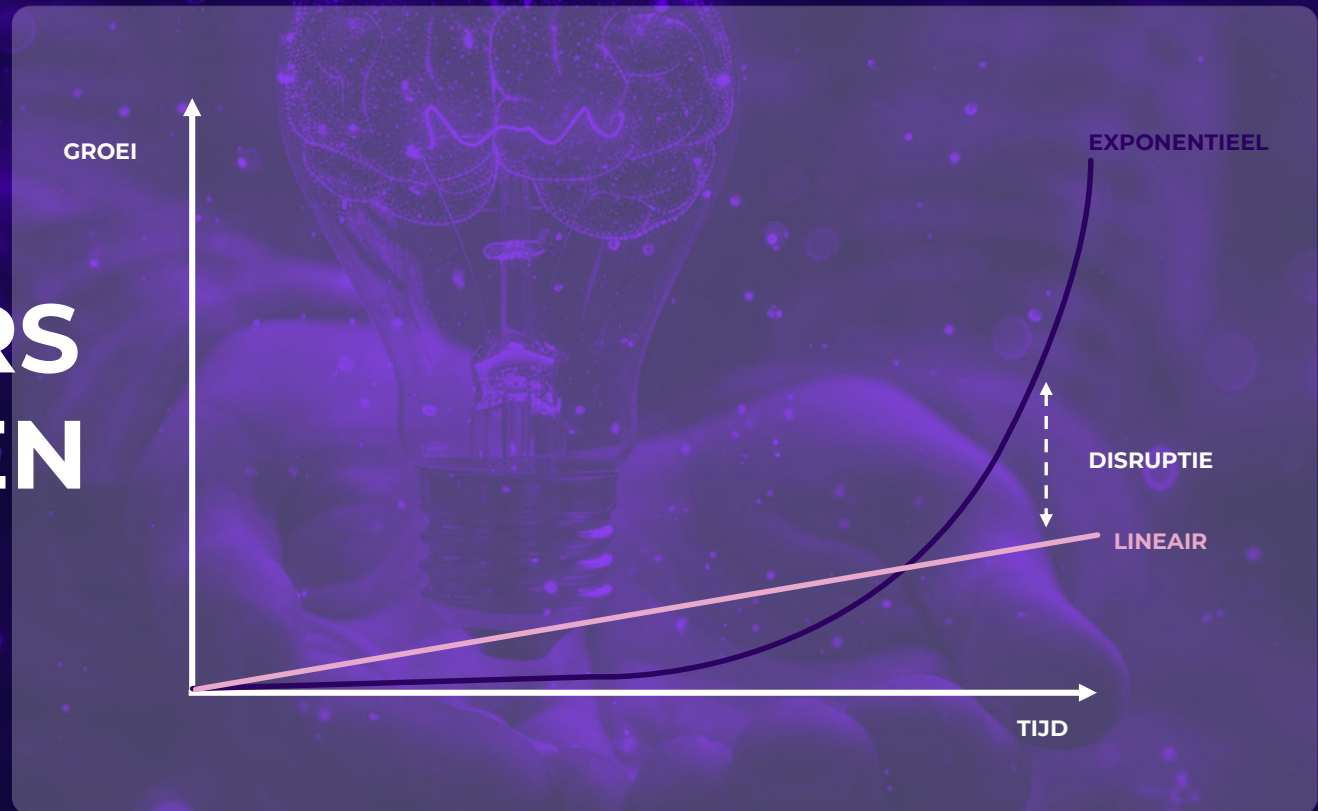


De fusie van Ai en hardware



**RANDALL
VAN
POELVOORDE**

ANDERS DENKEN



- Revision of some individual country maps, first introduced in the 2001 edition, is continued in this edition. Several regional maps have also been updated to reflect boundary changes and place name spelling changes.

The World Factbook (2008) - Country Listing
public static void main(String[] args)
{
int[] list = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
int sum = sumListEnhanced(list);
System.out.println("Sum of elements in list: " + sum);

System.out.println("Original List");
printList(list);
System.out.println("Calling addOne");
addOne(list);

change.");

DE DATA-EXPLOSIJE

System.out.println(" Mondays are bad.");
break;
case FRIDAY:
System.out.println("Fridays are better.");
break;
case SATURDAY:
case SUNDAY:
System.out.println("Weekends are best.");
break;
default:
System.out.println("Midweek days are so-so.");
break;
}
// Driver method
public static void main(String[] args) {
String str = "MONDAY";
Test t1 = new Test(Day.valueOf(str));
t1.dayIsLike();

```
public static void alterParametersExample ()  
{  
    System.out.println("In method alterParametersExample");  
    int x = new Parameters().x; // 20, 100  
    System.out.println("In method alterParametersExample");  
}  
  
// public static void get() {  
//     int x = 2;  
//     System.out.println("In method get, x = " + x + " y = " + y);  
//     falseSwap(y);  
//     System.out.println("In method get, x = " + x + " y = " + y);  
//     moveParameters(x,y);  
//     System.out.println("In method get, x = " + x + " y = " + y);  
// }  
  
public static void falseSwap(int x, int y)  
{  
    System.out.println("In method falseSwap, x = " + x + " y = " + y);  
    int temp = x;  
    x = y;  
    y = temp;  
    System.out.println("In method falseSwap, x = " + x + " y = " + y);  
}  
  
public static void moveParameters(int a, int b)  
{  
    System.out.println("In method moveParameters, a = " + a + " b = " + b);  
    a = a * b;  
    b = 12;  
    System.out.println("In method moveParameters, a = " + a + " b = " + b);  
    falseSwap(a,b);  
    System.out.println("In method moveParameters, a = " + a + " b = " + b);  
}  
  
// db credentials  
String jdbcEndpoint  
    = "jdbc:mysql://localhost:3300/GECKSFORGEKES";  
String username = "GFG";  
String password = "GEEKSFORGEKES";  
// create a connection to db  
Connection connection = DriverManager.getConnection(  
    jdbcEndpoint, username, password);  
// construct a query  
Statement statement = connection.createStatement();  
String deleteQuery  
    = "DELETE FROM USER WHERE AGE > 15";  
// Disable auto commit to connection  
connection.setAutoCommit(false);  
// Table User Connection can = null  
ResultSet resultSet = null;  
try {  
    Class.forName("com.mysql.cj.jdbc.Driver");  
} catch (ClassNotFoundException e) {  
    // TODO Auto-generated catch block  
    e.printStackTrace();  
}  
con = DriverManager.getConnection(jdbcEndpoint, username, password);  
if (con != null) {  
    // Retrieve all data  
    String selectQuery = "SELECT * FROM book";  
    PreparedStatement selectStatement = con.prepareStatement(selectQuery);  
    resultSet = selectStatement.executeQuery();  
    System.out.println("The Available Data is");  
    while(resultSet.next())  
    {  
        int id = resultSet.getInt("id");  
        // Check for NULL values  
        String author_name = resultSet.getString("author");  
        if(resultSet.wasNull()) {  
            author_name = "N/A"; // Or any default value you prefer  
        }  
        String book_name = resultSet.getString("name");  
        if(resultSet.wasNull()) {  
            book_name = "N/A"; // Or any default value you prefer  
        }  
        String book_price = resultSet.getString("price");  
        if(resultSet.wasNull()) {  
            book_price = "N/A"; // Or any default value you prefer  
        }  
        System.out.println(id + " * Author Name: " + author_name + " Book Name: " + book_name + " Book Price: " + book_price);  
    }  
    else {  
        System.out.println("Not Connected.");  
    }  
}  
  
} catch (SQLException e) {  
    System.out.println("Exception is " + e.getMessage());  
    connection = DriverManager.getConnection(jdbcEndpoint, username, password);  
    // Your SQL query with an intentional error (table name is incorrect)  
    String sql = "SELECT * FROM non_existent_table";  
    // create a PreparedStatement  
    PreparedStatement = connection.prepareStatement(sql);  
    // execute the query (this will throw an SQLException as the table name is incorrect)  
    PreparedStatement.executeQuery();  
    // catch the SQLException (Handled user-defined exception e)  
    // handle the SQL exception  
    handleSQLExceptions();  
    // Finally  
    // close the resources in the finally block  
    try {
```



NEURAL NETS

ONS BREIN IN SOFTWARE

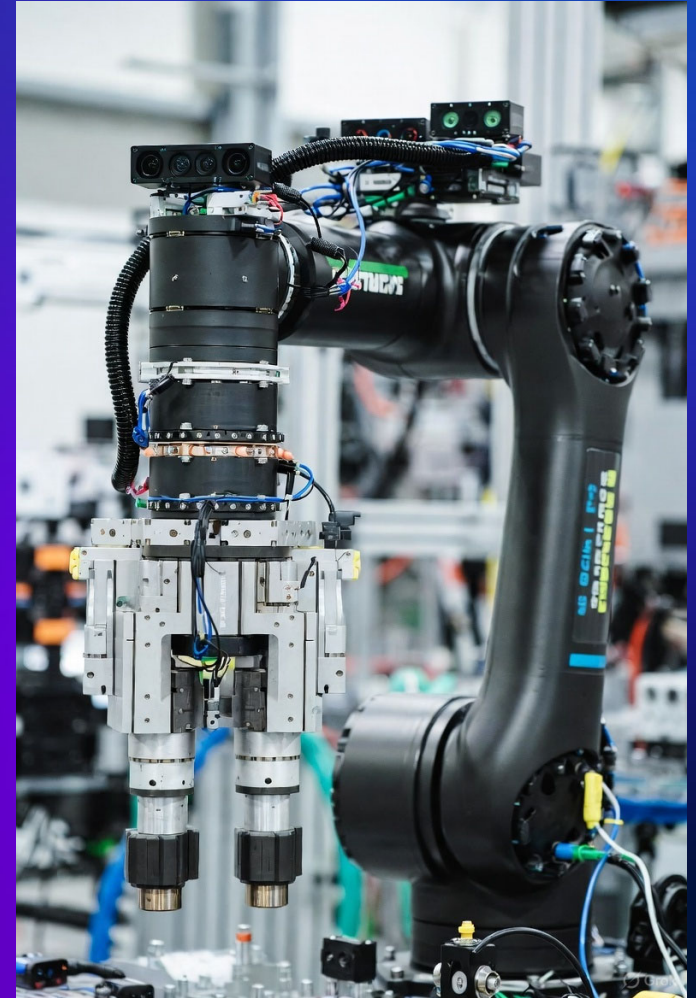


AGI

Artificial General Intelligence



Slimme productielijnen





Humanoid robots

WAAROM

KRIJGEN ZE OOK ONZE VORM?



WAAROM



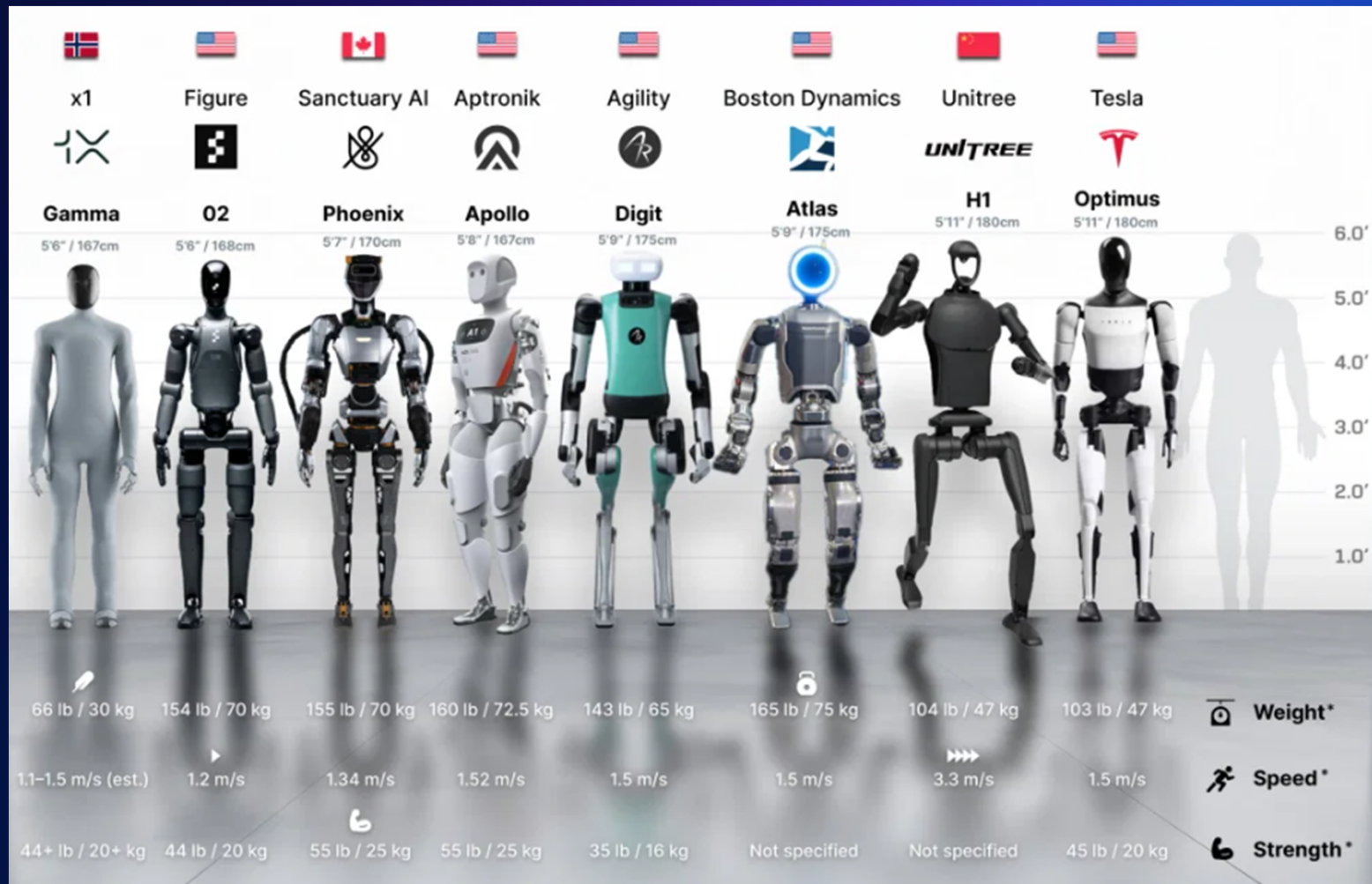
- **Arbeidstekorten**
- **Mens is de maat van alles**
- **Acceptatie**
- **Aanvullend op bestaande robots**
- **Tientallen taken per robot**
- **Gebruik standaard gereedschappen**



ZO LEREN ZE



Spelers (2025)



Wie wint de race?

1. Eigen Ai ecosystem (data, training en brein)
2. Beste hand
3. Massa productie (supply chain)

Salto's op YouTube zijn onbelangrijk



25k dollar / stuk (bij massaproductie 2026)

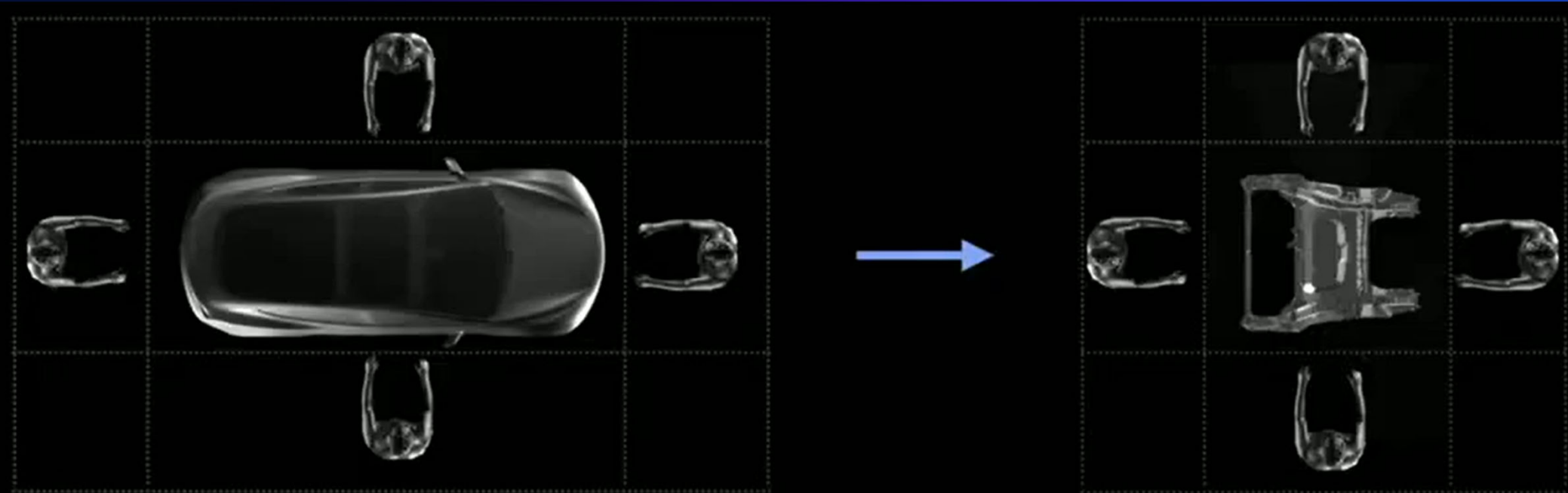




Fully Autonomous 1.5x

Optimus is learning many new tasks

Veranderingen in productie lijnen



Conclusie

1. Denk exponentieel
2. $\text{Ai} = \text{data} + \text{energie} + \text{compute}$
3. Werk aan data in je bedrijf
4. Ai leert daarmee echt zelf
5. Robots krijgen vele vormen; humanoid dominant
6. Totaal nieuwe productie lijnen worden mogelijk

