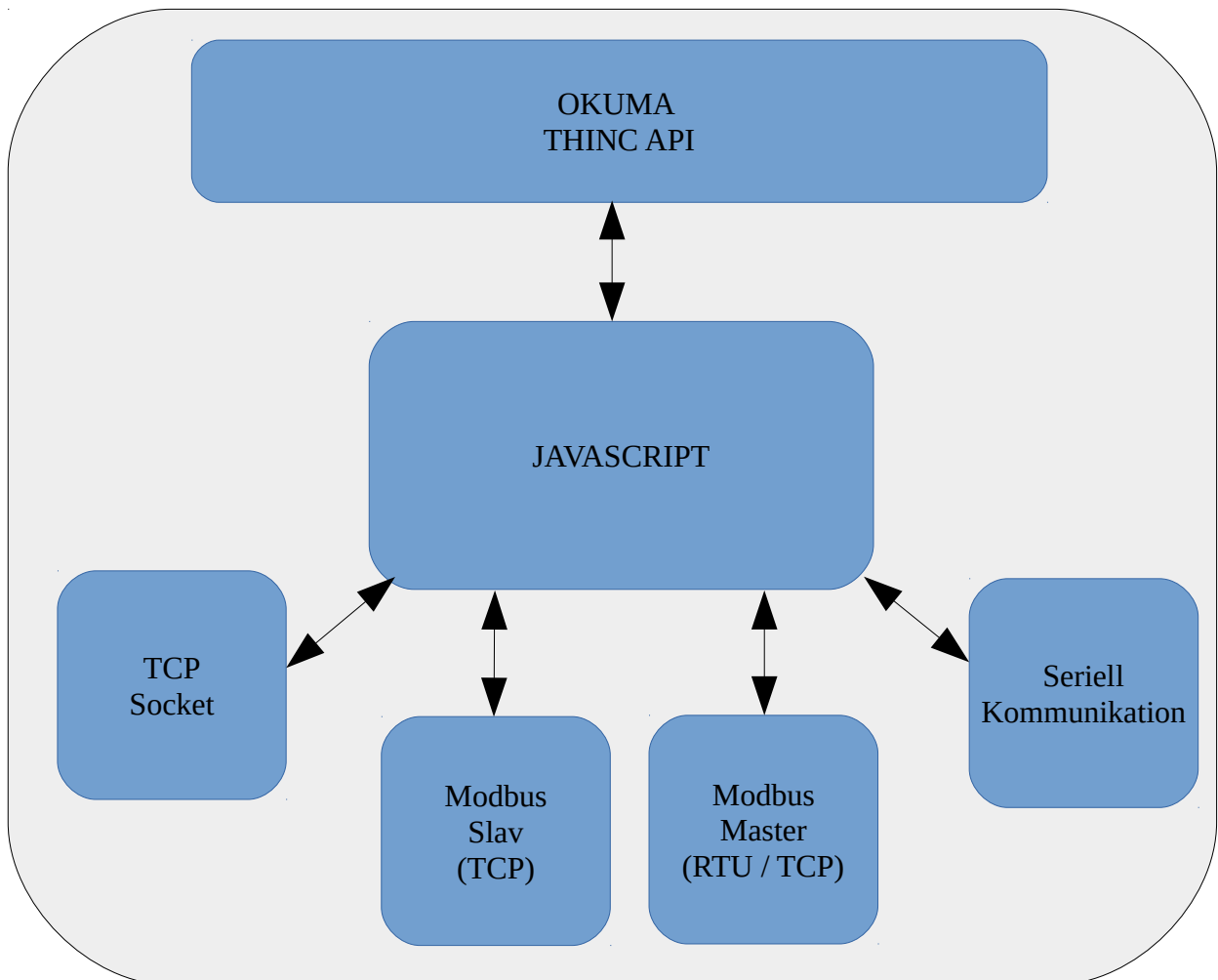




## Javascript Easy Communication



jsE är i grunden ett kommunikationsverktyg.  
Designat för att enkelt kunna skapa gränssnitt (interface) mellan olika utrustningar.

Programmet har för närvarande 5 delar.  
Beskrivet nedan finns dessa nedtecknade med exempel på vad de olika delarna kan hantera.

#### **Okuma Thinc API**

- Globala / Gemensamma Variabler
- Produktionsuppföljning (Macman)
- Verktygshantering & Verktogsdata
- I/O-status

#### **Modbus Slav / Server (TCP)**

- PLC
- HMI
- Anybus

#### **Modbus Master / Klient (TCP/RTU)**

- I/O
- PLC
- Anybus

#### **TCP Socket Server**

- Anpassad nätverkskommunikation

#### **Seriell Kommunikation**

- Anslutning till äldre utrustningar

jsE har ett inbyggt programmeringsgränssnitt (skriptspråk), man programmerar alltså direkt i programmet. Inga extra programmeringsverktyg behövs.

jsE är uppbyggd runt Chrome V8 Javascriptmotor utvecklad av Google. Alltså används delvis samma programvara som i Webläsaren Chrome.

# Användningsexempel

## **HMI (Operatörsinterface)**

Modbus är en industriell standard och stöds av nästan alla HMI-paneler. Exempelvis kan man visualisera verktygsdata på extern HMI. Alternativt kan man använda en HMI-mjukvara i maskinpanelen, för att göra avancerade tillägg till Okumas HMI.

## **Anybus**

För att kommunicera med externa enheter kan man använda en Anybus-enhet. För t.ex. parameterstyrning till/från extern utrustning.

## **I/O**

jsE kan även direkt hantera in och utgångar via Modbus.

## **PC-Mjukvara**

TCP Socket kan användas för att skapa gränssnitt mot extern PC-mjukvara.

## **Kommunikationsbrygga**

jsE kan levereras utan koppling till Okumas API, för användning i maskiner utan API eller i en extern dator.

Via det seriella modulen kan man kommunicera med äldre Okumamaskiner (med UserTask Option) och hantera data skickad från bearbetningsprogrammet.

Man kan även tänka sig mätdata skickad via nätverk och sedan vidarebefordrad till maskinen. (Okuma stöder endast mätdata via seriell överföring).

## **Anpassningar**

Anpassade kommunikations lösningar för t.ex. mjukvaror för produktionsuppföljning.

# Framtida Möjligheter

## **Databas**

Lagring av maskindata, produktionsdata m.m.  
Lokalt eller i en central databas.

## **Fanuc**

Ett möjligt framtida modul kan vara Fanucs Focas bibliotek. Vilket ger extern tillgång till vissa parametrar, macrovariabler, NC-data, in och utgångar m.m.

## **Heidenhain**

Med Python – programmeringsspråk som är inbyggt i Heidenhain och jsE's TCP modul kan man skapa gränssnitt för kommunikation.