



digitalSTROM

Developer Day
Virtual Devices

Christoph Hofmann

November 14th 2013

How to insert a non-dS device into dS?

digitalSTROM Configurator



Apps Activities **Rooms** Groups System Hardware Help

Rooms

Devices in Room "Room 3 - 1" 1 Device

Name	dSID	Output Mode	Input Mode	Input	Meter Name	Room Name	Con
EnOcean	000090ed	switched	Presets 0-4		dSM11	Room 3	

Refresh View Basic View English

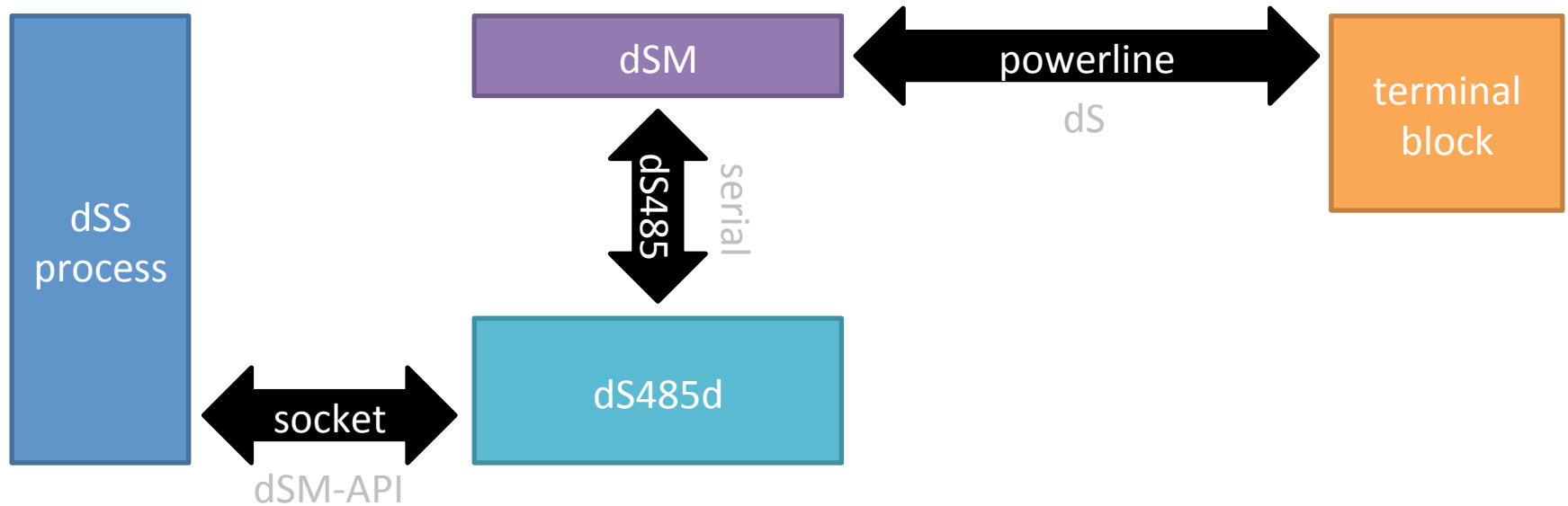
Solution: Virtual Devices

Topics

- Architecture for virtual device connection
- Virtual Device Connector API

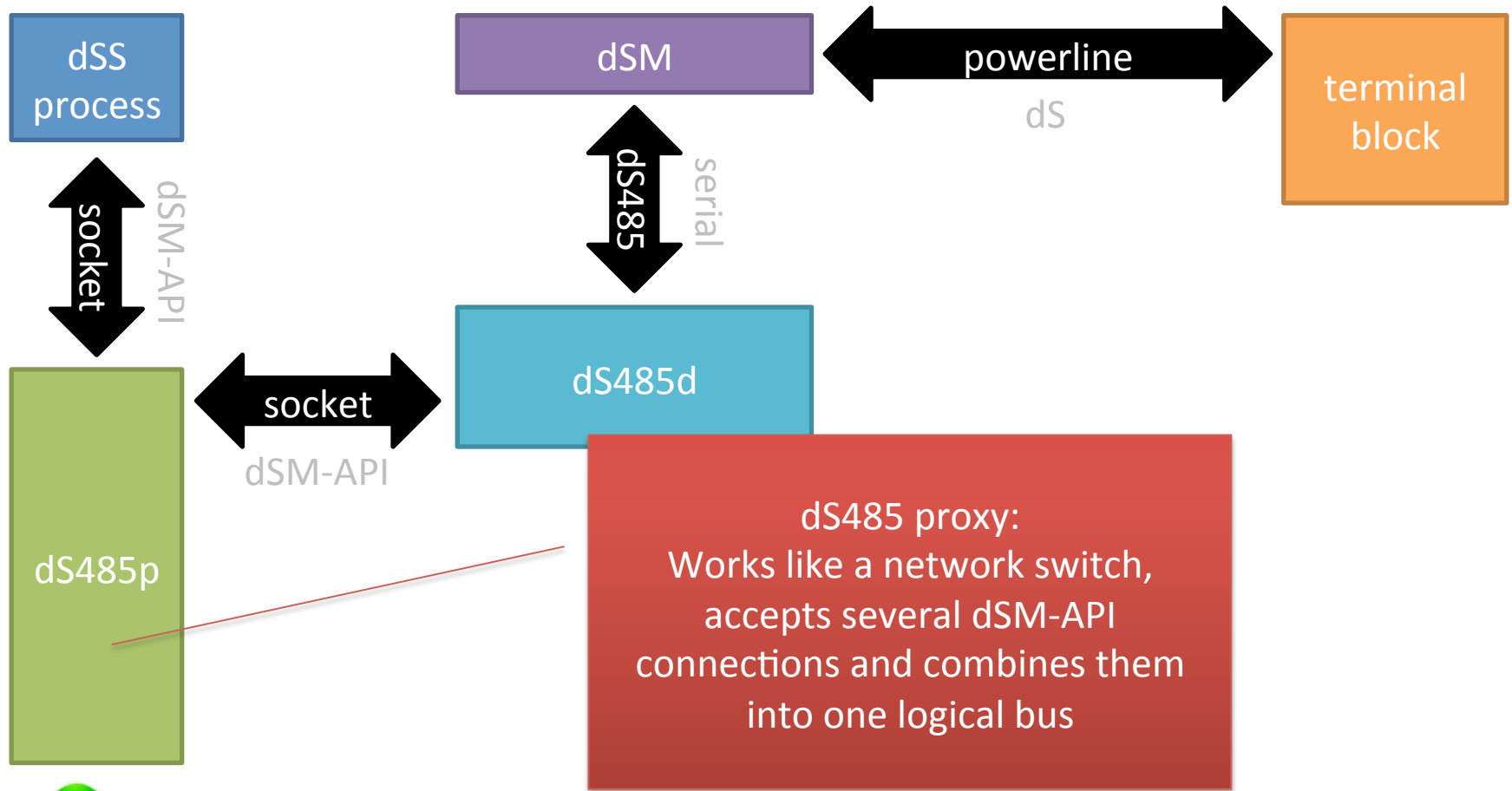
Architecture for virtual device connection

Current architecture



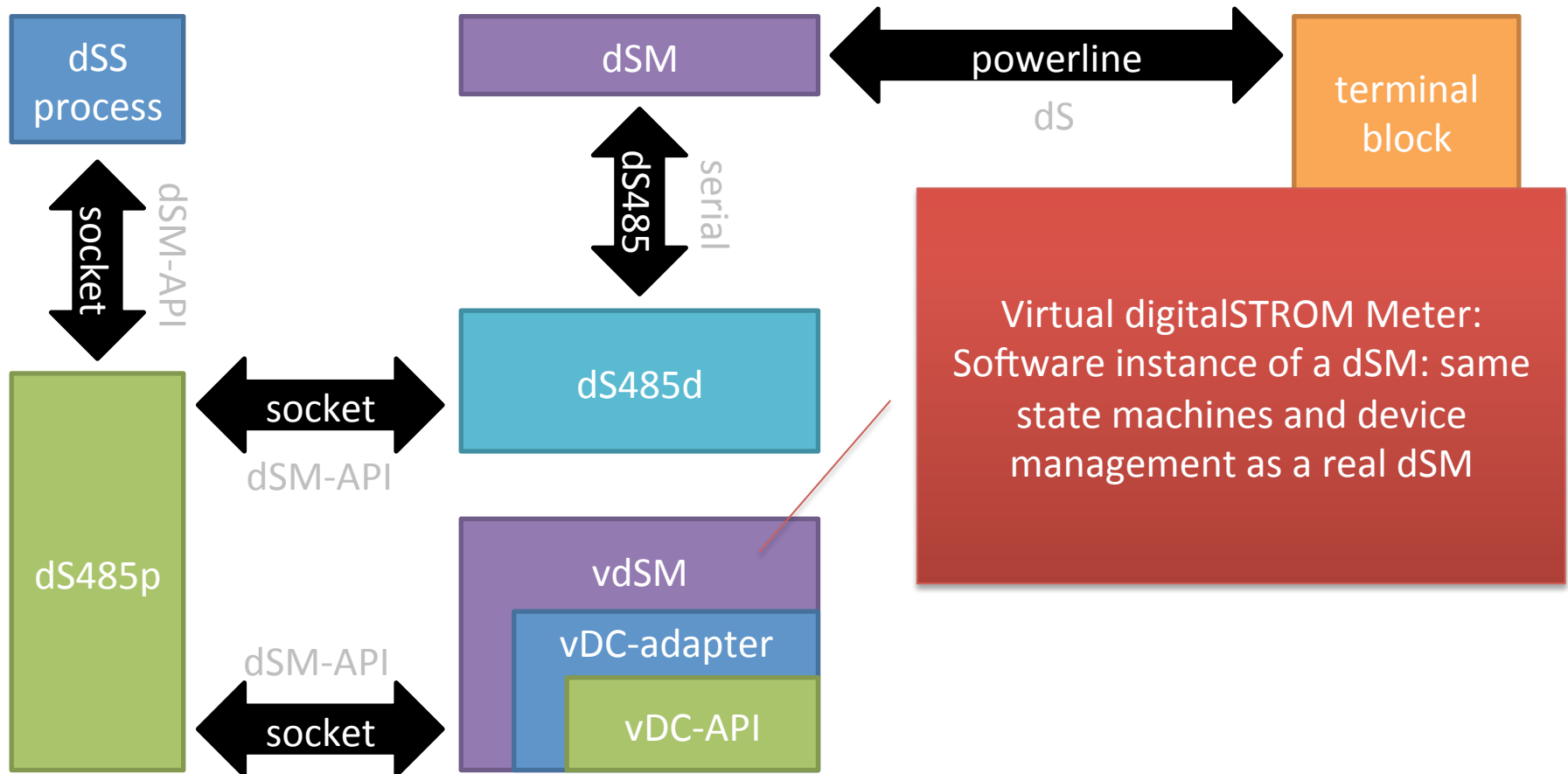
Architecture for virtual device connection

New architecture



Architecture for virtual device connection

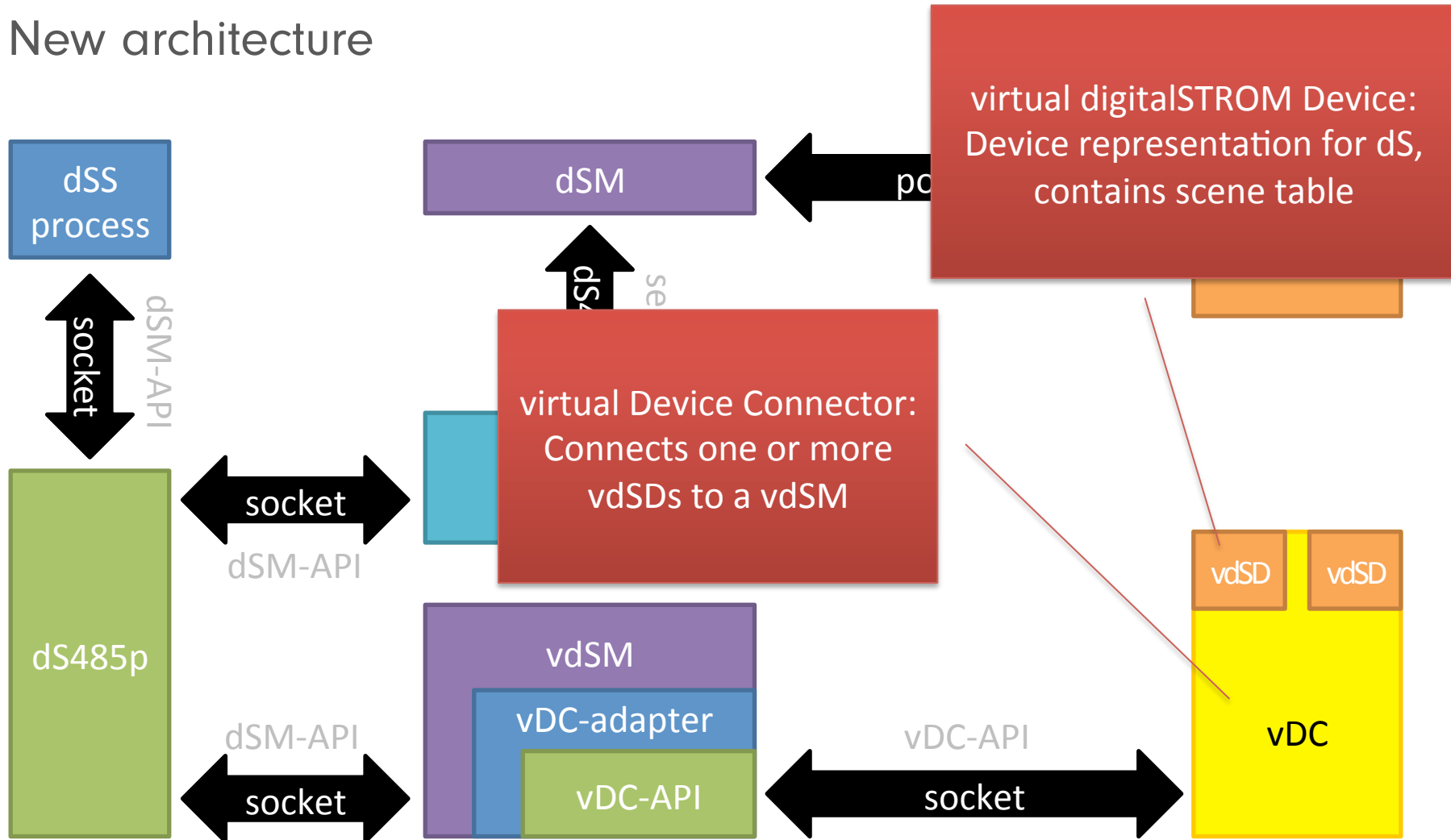
New architecture



Virtual digitalSTROM Meter:
Software instance of a dSM: same
state machines and device
management as a real dSM

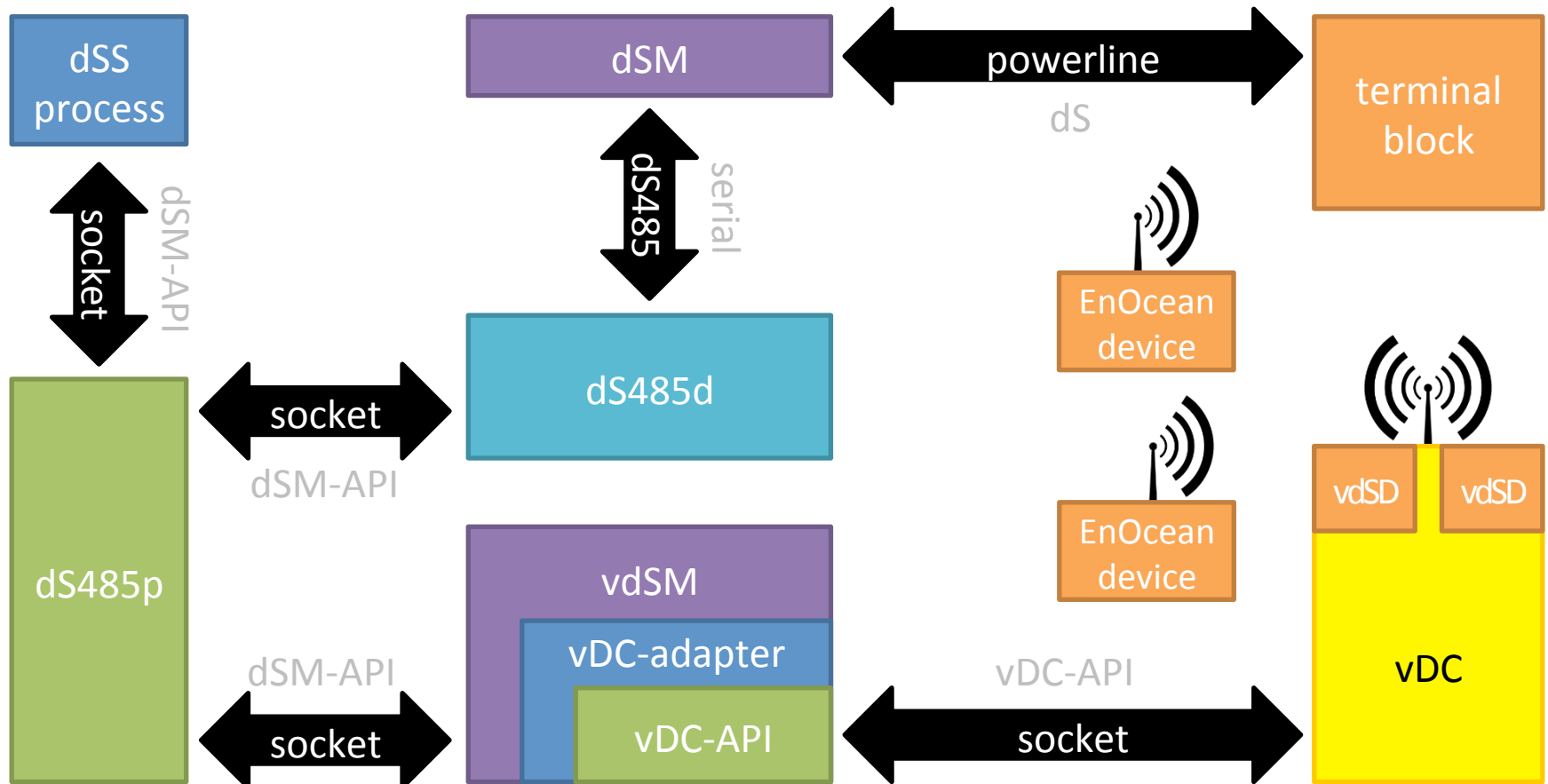
Architecture for virtual device connection

New architecture



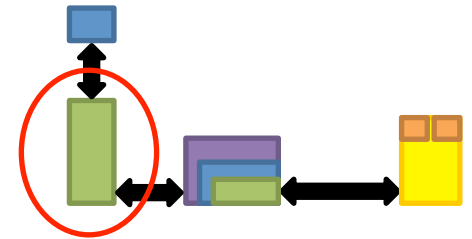
Architecture for virtual device connection

New architecture



Architecture for virtual device connection

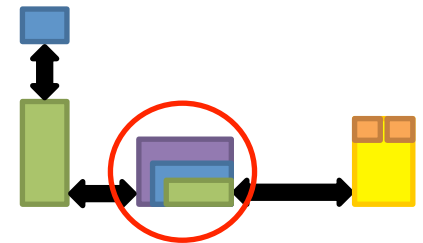
dS485p - dS485 proxy



- Works like a network switch, accepts several dSM-API connections and combines them into one logical bus
- Forwards all messages received on one bus to all other buses

Architecture for virtual device connection

vdSM – virtual digitalSTROM Meter

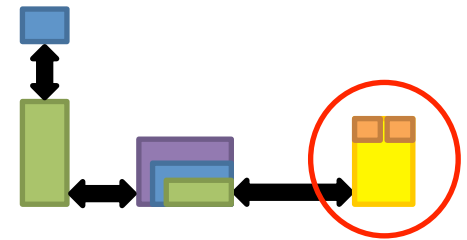


- Software instance of a dSM: same button state machines and device management as a real dSM
- Uses „named properties“ for communication with virtual devices
- Contains vDC-adapter submodule
 - translation between bank/offset parameters and named properties
- Provides vDC-API for virtual devices
- Discovers all available vDCs in the local network

Architecture for virtual device connection

vDC – virtual Device Connector

- Each device gets a virtual representation called vdSD (virtual digitalSTROM device)
 - Contains scene table
 - Button state machine generates click/tipp types
 - May have custom properties
- Can connect a single vdSD (e.g. a washing machine) or a whole bus of devices (e.g. EnOcean)
- May have custom (bus-)properties
- Same hierarchical level as dSM:
 - Maintains a default room for its attached devices
 - vDCs are shown in hardware tab alongside the dSMs



Virtual Device Connector API

vDC API

- Based on JSON-RPC 2.0
- TCP socket connection
- Reduced number of methods (callScene, saveScene, ...), most things handled via reading/writing named properties

Sessions

- vdSM connects to vDC and identifies himself sending "hello"
- vDC starts device sessions for all of the managed devices using method "announce"

Virtual Device Connector API

Methods

- hello, bye (session management)
- ping, pong* (communication check)
- announce*, vanish*, remove (device availability)
- getProperty, setProperty, pushProperty* (property exchange)
- callScene, callSceneMin, saveScene, undoScene, setLocalPriority (scene management)
- identify (visual device identification like blinking)
- setControlValue (dS sensor value transmission)

* communication vDC → vdSM

Questions

