



Ako Vyvíjať Driver, Časť 1

V prvej časti tejto lekcie sa pozrieme na základné princípy vývoja Pluginu typu Driver, ktorý je určený pre integráciu externých systémov.

Špecifické rozšírenia Drivera, ako je napríklad integrácia prístupových systémov alebo klientska integrácia kamery budú spracované v samostatných lekciách.

Driver je komponent prepájajúci Systém C4 s externým systémom. Jeho úlohou je monitorovať a ovládať jednotlivé časti externého zariadenia v prostredí Systému C4. Driver komunikuje s externým systémom nepretržite od naštartovania komunikácie až po jej zastavenie.

Každá instancia Drivera vždy komunikuje so sadou zariadení, ktoré majú jeden konkrétny sieťový prípojný bod. Ak máme viacero zariadení s viacerými prípojnými bodmi, Driver sa spustí vo viacerých instanciách pre každý prípojný bod samostatne. To znamená, že Systém C4 automaticky vytvorí a spravuje instance Drivera toľkokrát, koľko prípojných bodov je nadefinovaných v strome zariadenia. Pre vývojára je dôležité, aby pri návrhu integrácie externého systému bral tieto základné princípy do úvahy. Zásadne sa neodporúča integrovať jednou instanciou Drivera všetky spravované zariadenia spoločne. Takýto postup extrémne zaťažuje vnútorné zdroje Systému C4, ktoré sú vyhradené pre správu integrácie externých systémov.

V Systéme C4 má každá instancia Drivera zadefinovanú samostatnú konfiguráciu zariadenia, ktoré má spravovať. Takáto konfigurácia obsahuje všetky podsystémy v konkrétnom zariadení. Preto ak máme dve zariadenia, ktoré sa líšia v konfigurácii (ako vidíme v našom prípade, jedno zariadenie má jeden výstup a druhé dva výstupy), tak aj ich stromy v Systéme C4 sa budú odlišovať. Driver musí byť naprogramovaný tak, aby vedel pracovať so všetkými možnými konfiguráciami, ktoré výrobca zariadenia podporuje.

Ak vývojár použije generované funkcie od Gamanetu, bude mať k dispozícii predpripravené kódy podľa jednotlivých typov podsystémov. Jeho úlohou bude napísať kód pre daný objekt, napr. výstup tak, aby bol schopný pracovať vo viacerých instanciách. Pri inicializácii Kontextu zariadenia sa tieto instance automaticky vytvoria podľa konfigurácie v strome zariadení. Týmto sa výrazne šetrí práca a čas vývojára. Vzhľadom na architektúru Drivera sa pri programovaní zásadne neodporúča používať statické premenné alebo objekty.

Logika zariadenia je modul vyvinutý vývojárom, v ktorom je implementovaná Logika správania sa zariadenia. Reprezentuje fyzické časti systému, a zároveň je spojnícou medzi nimi a ich reprezentáciou v Systéme C4. V Driveri sa snažíme zrkadliť a čo najpresnejšie nasimulovať Logiku správania sa zariadenia, ako aj implementovať všetky podmienky, podľa ktorých sa zariadenie správa. Táto Logika nám následne umožní ovládanie zariadenia, či zasielanie informácií o stave zariadenia do Systému C4.



Zrkadlenie zariadenia je v Driveri rozložené do častí, ktoré reprezentujú jednotlivé podsystémy. Keď máme na zariadení dva výstupy, tak budeme mať dve instance objektu typu výstup. Každá z nich obsahuje tú istú Logiku, ale iné dáta.

Vnútná architektúra Driveru sa z pohľadu funkčnosti skladá z niekoľkých logických celkov.

Framework 4 je SDK dodávané Gamanetom pre vývojárov Driverov. Obsahuje podmodul Link, ktorý zabezpečuje komunikačný kanál so zariadením, Packet Processor, ktorý zabezpečuje fyzickú komunikáciu paketov - posielanie dát a spracovanie odpovedí, a podmodul Access Calculator, ktorý pomáha vývojárovi so spracovaním nastavení prístupových oprávnení v zariadeniach spravujúcich identifikátory. Každú z týchto častí si bližšie popíšeme v samostatnej lekcii.

Logika zariadenia je blok kódu vyvinutý vývojárom Driveru. Zatiaľ čo je Framework 4 je dodávaný Gamanetom, Logiku zariadenia musí naprogramovať vývojár.

Keďže každé zariadenie má iný protokol, tak aj každý Driver má iné typy paketov. Logika zabezpečuje vytváranie paketov, ktoré sú zasielané do zariadenia a taktiež spracovanie odpovedí zo zariadenia. Taktiež zabezpečuje spracovanie príkazov zasielaných zo strany Systému C4 do Driveru. Samozrejme, jednou z hlavných úloh Logiky je zasielanie údajov do Systému C4, či už ide o udalosti alebo informácie o stavoch zariadenia.

Štruktúra Driveru popísaná v tejto lekcii je univerzálna a je možné ju aplikovať pri integrácii väčšiny externých systémov. V nasledujúcich lekciiach si popíšeme, ako použiť jednotlivé moduly Frameworku 4 pri rôznych typoch systémov.