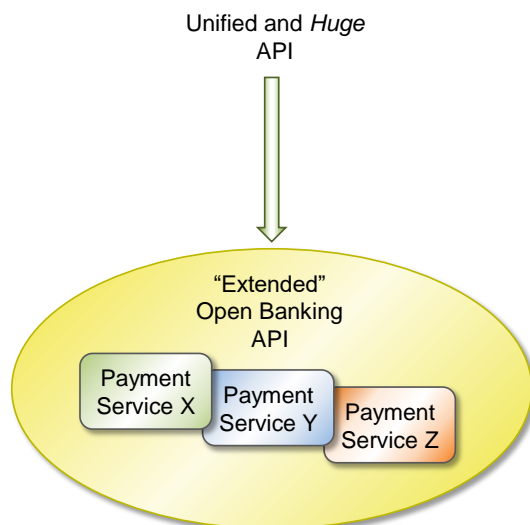


## Embedded SCA versus Direct Mode

Feature	Embedded SCA	Direct Mode	Comment
Independent Development	✗	✓	Direct Mode permits <i>anybody</i> to develop a payment system including keeping the design <i>private</i> .
Platform Independence	✗	✓	Since Direct Mode services are <i>external</i> to Open Banking, they may build on different platforms.
Easy to Standardize	✗	✓	Direct Mode only requires a <i>single</i> and fully standardized change to work.
“Sandbox” Support	✗	✓	Direct Mode payment systems can be developed, demoed, and marketed based on “Sandbox” implementations.
“Cloud” Compatible	✗	✓	Direct Mode permits multiple deployment models, including running payment services in the cloud.
Service Isolation	✗	✓	Direct Mode lends itself to run payment services on dedicated hosts making maintenance and upgrades more manageable than potentially complex ( <i>and entirely proprietary</i> ) Open Banking API implementations.
Supports P2P Payments like “Swish”	✗	✓	Direct Mode services are <i>unrestricted</i> with respect to flow, security, and processes.
Cost Efficient	✗	✓	Direct Mode makes it technically possible creating a <i>single implementation/code base supporting all Banks</i> .
Update Management	✗	✓	Updating Embedded SCA applications should be <i>quite challenging</i> due to the diversity of Open Banking implementations.
Security / Authentication	✓	✓	<p>In the <b>Direct Mode</b> security maintained by each of the connected services on their own.</p> <p>In the <b>Embedded SCA</b> model most of the security is confined to implementations-specific solutions inside of the Open Banking framework.</p> <p><i>These methods should be comparable although it seems that it should be easier to audit software that is 1) used by more customers 2) is not a part of another framework.</i></p>

### Embedded SCA



### Direct Mode

