



Silicon Linux

PlusG SMART Solution
組込み向けGUIライブラリ

+G

PlusG SMART Solution

with Android



**PlusG SMART Solution を使えば
スマートフォンやタブレットを
組込み機器の表示 / 入力デバイス
として利用できます。**

www.si-linux.co.jp

PlusG SMART Solution は

スマートデバイス（タブレットやスマートフォンなどの Android 端末）を組み込み機器の表示デバイス、入力デバイスとして活用するソリューションです。お客様の既存システムの「スマートデバイス対応」も、スマートデバイスを使った、新規サービスの開始も、簡単に実現できます。スマートデバイスだけではできなかった制御機器の操作を可能にし、組み込み機器では難しかった綺麗で滑らかな画面表示を可能にします。

組み込み装置を GUI 負担から解放

優れた GUI 操作には、綺麗な画面表示や滑らかな動きなどが必要です。それらを実装するには高速な CPU が必要です。

また書体のライセンス、日本語変換など有償のミドルウェア コストも必要になります。

「PlusG with Android」を使えば、GUI に関わる処理をスマートデバイスに任せることができ、組み込み装置は、リアルタイム制御など本来の仕事に専念できます。

組み込み装置側（サーバ）から GUI の仕事を減らした分、CPU パワーを下げることができ、コストカットも期待できます。

Android スマートデバイス側のプログラミング不要

Android プログラミングの知識（Java 言語、Android 専用 API、流儀、決まりごとなど）の習得、学習の必要はありません。

Android 端末へは専用のアプリケーションをインストールするだけです。

クライアント・サーバ方式

タブレットやスマートフォンなどのスマートデバイス側がクライアント、組み込み機器側がサーバとして動作します。

スマートデバイスにインストールした PlusG with Android のモニタプログラムはサーバから送られてくる画面情報からオブジェクトを生成し画面に配置します。

配置されたオブジェクトは Android の機能としてサーバからの指示がなくても独立して動作します。

具体的には、ボタン押下、チェックボックス / ラジオボタンの状態変化、リストの表示 / 選択、エディットボックスへの数値や日本語の入力、

スライドジェスチャーによる画面スクロールなどの動作は（サーバと通信を行うことなく）端末単体で機能します。

端末は予め登録されたオブジェクトに変化が起けると、その内容をサーバへ通知します。

1 対 N 接続に対応

1つのサーバ（組み込み機器）に、複数の端末を設置可能です。

Android 端末はそれぞれ独立して画面を表示するため、組み込み機器側は各端末の状態遷移を管理する必要がありません。

また、あえて各端末の状態遷移を管理する事も可能です。

例えば、教室の先生と複数生徒という様な場面において、生徒全員の状態を一斉に切り替えるということも可能です。

PlusG SMART Solution

PlusG with Android

クライアント・サーバ型

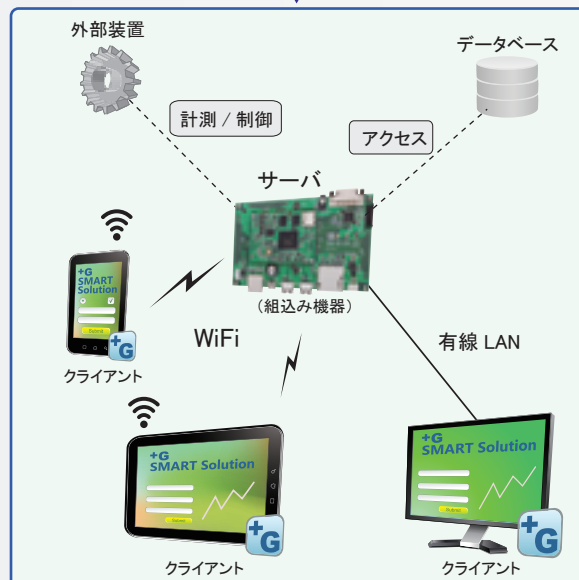
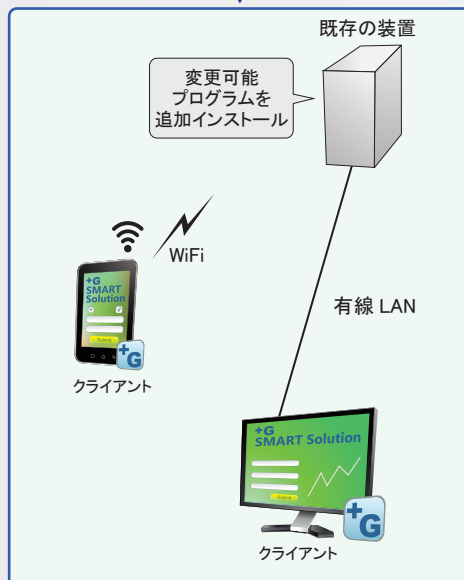
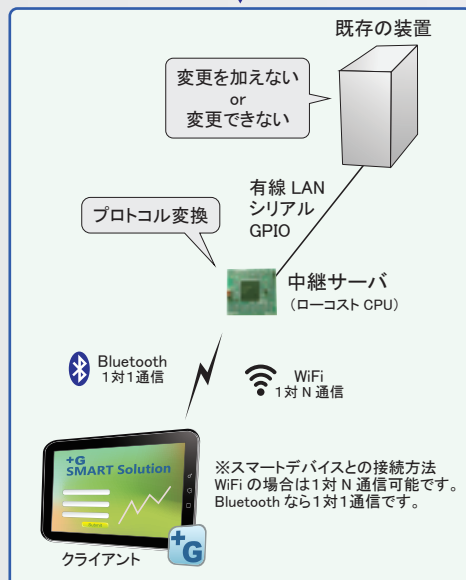
既存システムのスマートデバイス対応

装置に有線 / 無線 LAN、Bluetooth が無い、もしくは、新規にプログラムを追加インストール出来ないが、シリアル通信、GPIO など別の手段でコントロール可能な場合は、プロトコル変換中継サーバを設置します。

装置に有線 / 無線 LAN、Bluetooth が有る、もしくは増設可能で、新規にプログラムをインストールすることができる場合は、PlusG SMART Server を使用してサーバ機能を追加します。

組み込み機器を新規に作成する

スマートデバイスに対し情報の発信を行う組み込み機器（サーバ）を新規に作成します。PlusG Server はソースコード提供ですので、Linux/Windows、ARM/SH/intel など、アーキテクチャに依存しません。



PlusG SMART Solution

さまざまな接続方法

クライアント・サーバ間通信は WiFi、Bluetooth はもちろん 3G 回線インターネット経由での遠隔操作も可能です。
また、有線 LAN、RS-232C、GPIO 接続などいろいろな接続方法に対応可能です。

自由なデザイン

ボタンやチェックボックスなどの外観は準備された豊富なデザインコレクションから選ぶだけで簡単に変更できます。
また、自作のオリジナル画像を使用することもできますので、既存のアプリケーションと全く同じ外観にする、オリジナリティを高めることも可能です。
専用のダイアログエディタが付属していますので、手軽に画面設計ができます。

スマートなデバイス

タップやスライド、フリックなどのタッチパネル入力、スムーズかつ高速なスクロールなど、スマートデバイスが得意とする優れた入力デバイス、表示デバイスを活用できます。また、GPS、傾斜センサ、カメラなど端末に標準装備されているデバイスからもイベントを作成できます。

豊富な機能

サーバ側の開発は C++ を使用します。サーバ側で使用する PlusG のサーバ用ライブラリは、ベクトルデータの描画、グラフ表示、データベースへのアクセス、シリアル通信、ネットワーク通信などアプリケーション開発に必要な機能を多数装備しています。

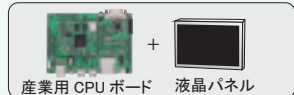
組込み機器（サーバ）側プログラミングは従来の CUI とほぼ同様

GUI 機能をスマートデバイスに任せているのでサーバ側のプログラミングもシンプルになり CUI アプリケーションと同様の感覚でコーディング出来ます。
クライアント側で発生した「ボタンが押された」などのイベントはネットワークを通じて組込み機器側へイベントとして通知されます。
組込み機器側でイベントに対応した処理のプログラミングを行います。
組込み装置側で発生したイベント（ハードウェアの状態変化など）はクライアント側にプッシュ通知し画面を更新します。
C++ の知識だけで、Android スマートデバイス開発を行うことができます。

PlusG Stand-alone 独立型

365日24時間稼働し続ける場合
など、産業品質が必要な場合
（組込み CPU ボード、液晶パネル
を組み合わせた専用装置）
でも使用可能です。

スタンドアロン型の 組込み装置



クライアント



※Stand-alone(独立型)をサーバとして
スマートデバイスから表示・コントロール
させることも可能です

開発手順

手順1 画面のデザイン

画面をデザインします
専用のダイアログエディタを使い
オブジェクトをマウスで配置するだけで
ダイアログが作成できます。
オブジェクトの外観にオリジナル画像を使用する場合は、
その画像を用意する作業が必要です。
PlusG 標準のリソースを使用する場合は必要ありません。

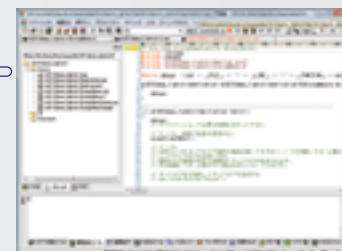


手順2 スケルトン ソースコードの出力

デザインした画面データを元に、C++ のソースコードを自動生成します。
生成されたソースコードには、ボタンのクリックやエディットボックスの
テキストの変化などのイベントに対応した空関数がつくれます。

手順3 サーバ側アプリケーションのプログラミング

PlusG にはプロジェクト管理機能付きテキストエディタ
「統合開発環境 SWEET」が付属しています。
自動生成された各ダイアログソースのイベント関数に必要な処理を
コーディングします。
デザインしたダイアログのオブジェクトは、専用のコマンドで、
文字列や選択状態などの変更や、現在の状態の取得などが可能です。



手順4 モニタプログラムのインストールと実行

スマートデバイスにモニタプログラムをインストールし、実行します。
サーバのアドレスを登録すると作成したダイアログが表示されます。

ライセンス

PlusG のライセンスは、開発プロジェクト単位になっており、開発者の人数、使用する PC 数に制限はありません。もちろんランタイムライセンスも不要ですので出荷台数にも影響しません。
ライセンス契約後、スマートデバイスへインストールするプログラムの起動スプラッシュ画面の内容とアイコンを貴社専用のデザインに変更します。

受託開発

アプリケーションの開発、外部 H/W との連携、通信などの開発、Linux デバイスドライバの作成、Linux の移植など、一部または全部をお手伝いさせて頂くことも可能です。
スマートデバイスにインストールするクライアントソフトのカスタマイズも対応できます。

オブジェクト

オブジェクトはボタンやエディットボックスなど、機能（動作）別に分けられていて、大きく分けて「機能オブジェクト」と「グループオブジェクト」の2つの種類があります。
「機能オブジェクト」は外観があり、画像ファイルを指定することで自由にデザインをカスタムできます。
この時使用する画像ファイルにスマートデバイスで使用されている「NinePatch」というフォーマットの画像が使用可能です。
このフォーマットは画像を井の字のように9つのパーツに分け、パーツごとに拡大方法を変化させるもので、ボタンやフレーム画像を拡大表示する際にデザインが崩れないようにすることができます。オブジェクトの外観イメージファイルに「NinePatch」を使用することで、表示される時のサイズが複数存在する場合も、1つのファイルで対応可能になります。
「グループオブジェクト」は外観がなく、複数のオブジェクトをまとめ、配置する役割があります。
グループオブジェクトの中には別のグループオブジェクトを含めることができ、いくつものグループを使い分けることでダイアログのデザイン（配置）が決定されます。

ダイアログ作成ツール

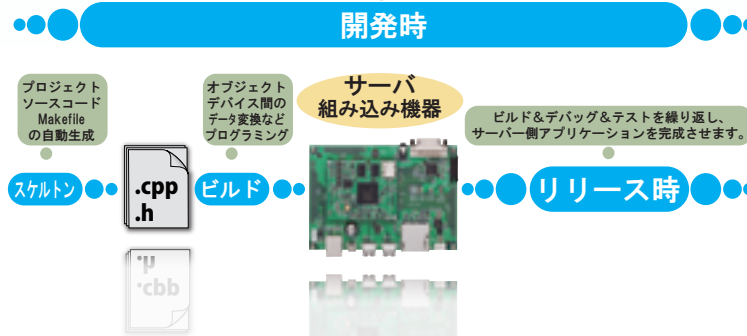
- 特徴1 付属のリソースエディタを使って驚くほど簡単にダイアログが作成できます。
- 特徴2 実機を使って、画面遷移 / 画面レイアウト / スライド・タッチ・スクロールなど実際に動作させながら開発できます。
- 特徴3 自作画像や標準添付画像を使って、簡単にオリジナルのオブジェクトを作ることができます。
- 特徴4 完成したレイアウト情報をもとに C++ のソースコード、プロジェクト、Makefile など自動生成されます。

エディタ & 疑似サーバ



画面遷移 / レイアウト / スライド・タッチ・スクロールなどの操作感など実機で確認しながら開発します。送信ボタンを押すだけで、すぐに確認できます。

クライアント



ボタン 	プルダウン 	チェックボックス 	スライダー 	トグル On/Off があり、スイッチとして入力、ランプとして出力に使用できます。									
テキスト あいう	エディット 	ラジオボタン 											
ピクチャー ドロー系描画機能 写真表示 9パッチ対応 (ドラッグ移動、ピンチで縮小・拡大)		メーター 	グリッド <table border="1"> <tr> <td>Apple</td> <td>100</td> <td>358</td> </tr> <tr> <td>Banana</td> <td>200</td> <td>629</td> </tr> <tr> <td>Cherry</td> <td>300</td> <td>174</td> </tr> </table>	Apple	100	358	Banana	200	629	Cherry	300	174	トレンドグラフ
Apple	100	358											
Banana	200	629											
Cherry	300	174											
フレーム オブジェクトを重ねて表示する場合に使用します。 	リニア 横 / 縦方向に並べて配置します。均等に配置することもできます。 	スクロール 表示範囲を超える場合にオブジェクトをスクロール表示することができます。 	スライドページ フリック動作、もしくはドラッグ動作でページを変えるように、他のオブジェクトに切り替えることができます。 	テーブル 列配置グループと行配置グループを組み合わせて表型配置を行います。 									