

パフォーマンス比較（前機種 vs 新プラットフォーム試作）

スケジュール比較

		前機種	DummyWinOS 全体 (アプリ)	アプリ改善内容(案)	アプリ速度 改善予測値	改善効率 (全体)	アプリ以外の改善要因 による速度改善見込	備考
アプリ起動	5件 100件 200件	200msec以下	1531msec	スケジュール一覧画面 カスタムウィジェット化	278msecを2倍に 278msecを2倍に 276msecを2倍に	約10% 速度Up	1. 基本動作性能改善 2. 表示性能改善 3. DB改善案件	・カスタムウィジェット化は、修正インパクト大。
		200msec以下	1666msec					
		200msec以下	1800msec					
カーソル移動	Scroll無 Scroll有	100msec以下	5msec	※速度に問題無し	-	-	1. 基本動作性能改善 2. 表示性能改善	・改善は不要。
		100msec以下	38msec					
登録	-	200msec以下	732msec	※DB改善があれば許容範囲？	-	-	2. 表示性能改善 3. DB改善案件	・表示は共通LIB (ViewLib) を使用。 ポップアップ画像リソース取得も含む。

タスクリスト比較

		前機種	KCP+共通ソフト 全体 (アプリ)	アプリ改善内容(案)	アプリ速度 改善予測値	改善効率 (全体)	アプリ以外の改善要因 による速度改善見込	備考
アプリ起動	5件 50件	200msec以下	1285msec	スケジュール一覧画面 カスタムウィジェット化	226msecを2倍に 236msecを2倍に	約10% 速度Up	1. 基本／2. 表示改善 3. DB改善案件	・カスタムウィジェット化は、修正インパクト大。
		200msec以下	1307msec					
カーソル移動	Scroll無 Scroll有	100msec以下	1msec	※速度に問題無し	-	-	1. 基本動作性能改善 2. 表示性能改善	・改善は不要。
		100msec以下	83msec					
登録	-	200msec以下	482msec	※DB改善があれば許容範囲？	-	-	2. 表示性能改善 3. DB改善案件	・表示は共通LIB (ViewLib) を使用。 ポップアップ画像リソース取得も含む。

カレンダー比較

		前機種	KCP+共通ソフト 全体 (アプリ)	アプリ改善内容(案)	アプリ速度 改善予測値	改善効率 (全体)	アプリ以外の改善要因 による速度改善見込	備考
アプリ起動	10件 250件	200msec以下	1321msec	※既にカスタムウィジェット化済	-	-	1. 基本動作性能改善 2. 表示性能改善 3. DB改善案件	・これ以上の最適化は厳しい。
		300msec以下	1703msec					
カーソル移動	0件 10件 250件	100msec以下	8msec	※アプリ側改善案無し。 DB速度向上に期待。	-	-	1. 基本動作性能改善 2. 表示性能改善 3. DB改善案件	・DBアクセス向上により改善可能。
		100msec以下	145msec					
		100msec以下	1536msec					
登録	-			-	-	-		・カレンダーからは登録画面へ遷移するのみ。

◇「アプリ起動」の速度改善

さまざまなオーバーヘッド要因が存在し、それぞれに改善余地がある。

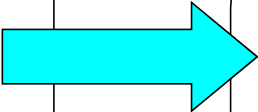
- 1. 基本動作性能改善
- 2. 表示性能改善
- 3. DB改善案件
- A. アプリケーション改善

◇「カーソル移動」の速度改善

現行にて前機と同等速度を実現している。積極的改善は不要。

◇「登録」の速度改善

大部分はデータベース部・OEM部の速度改善に依存し、アプリ側での改善効果は微小であると考えられる。



アプリ側で改善可能な部分として、
「アプリ起動」の速度改善
実施は有効である。

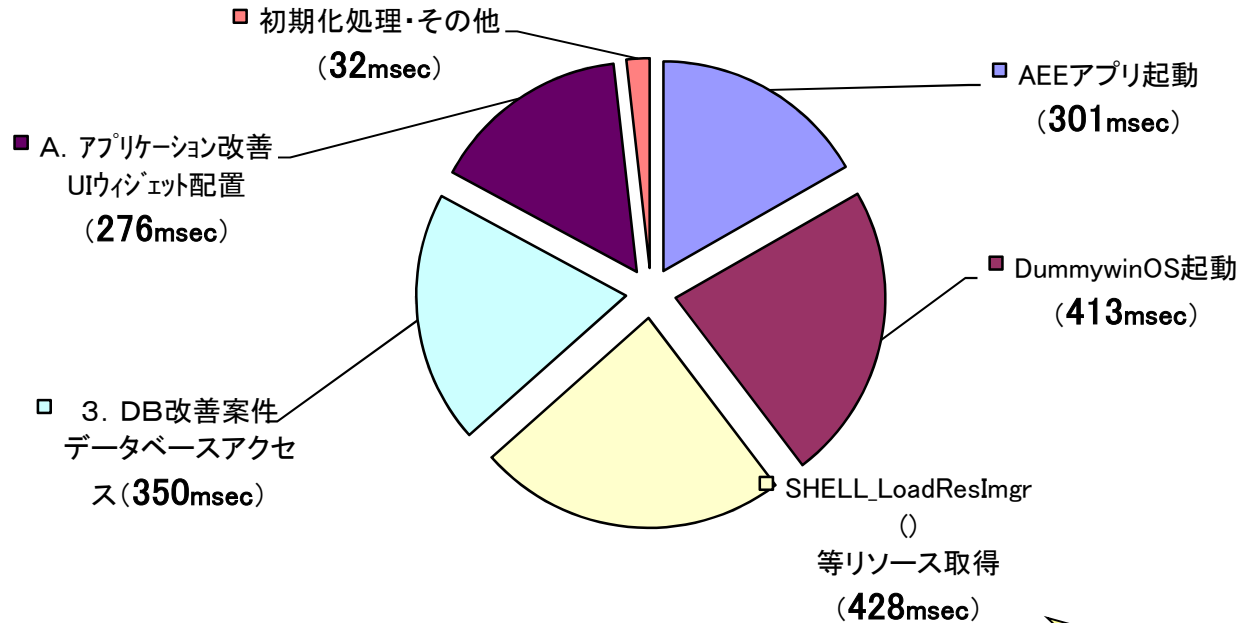
ウィジェット配置の最適化
(現行処理廃止・カスタムウィジェット化)
内部処理の最適化 etc...

しかし改善効率は、
アプリ部の速度を2倍に改善した場合、
全体のわずか「10%速度Up」に過ぎない。

パフォーマンス内訳（アプリ起動）

新モデル(DummyWinOS)
スケジュールアプリ起動（MAX200件時 1800msec）

前モデル(ネイティブ)
スケジュールアプリ起動
(MAX200件時 200msec以下)



前モデルを目標値とした場合、
約9倍の速度を実現しなければならない。
(1800msec vs 200msec)

全チームが、
仮に「速度20%Up」を実現しても、
1800msec → 1440msec
目標値(200msec以下)には、
到達できない

- ・「AEEアプリ起動」(301msec)
キー入力によるアプリ起動から、
アプリ起動完了(EVT_APP_START_WINDOW)まで。
- ・「DummyWinOS起動」(413msec)
イベント「EVT_APP_START_WINDOW」から、
ウィンドウ生成手続き完了(PROP_WINDOW_VISIBLE)まで。
- ・「SHELL_LoadResImgr()系リソース取得」(428msec)
画面用の画像リソース・文字列リソース等の取得処理。

◆結論：

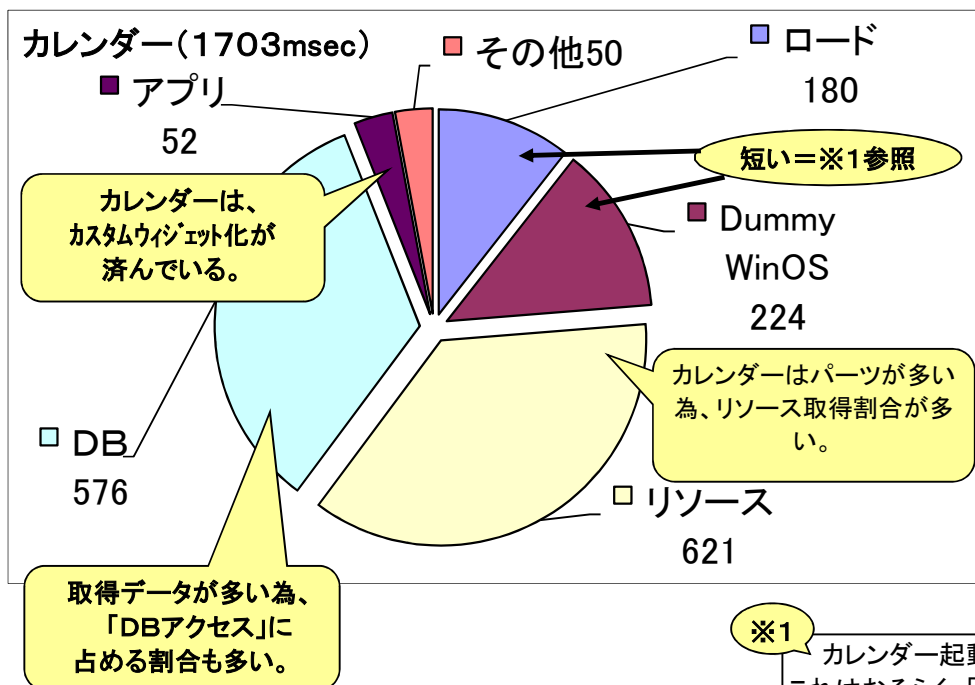
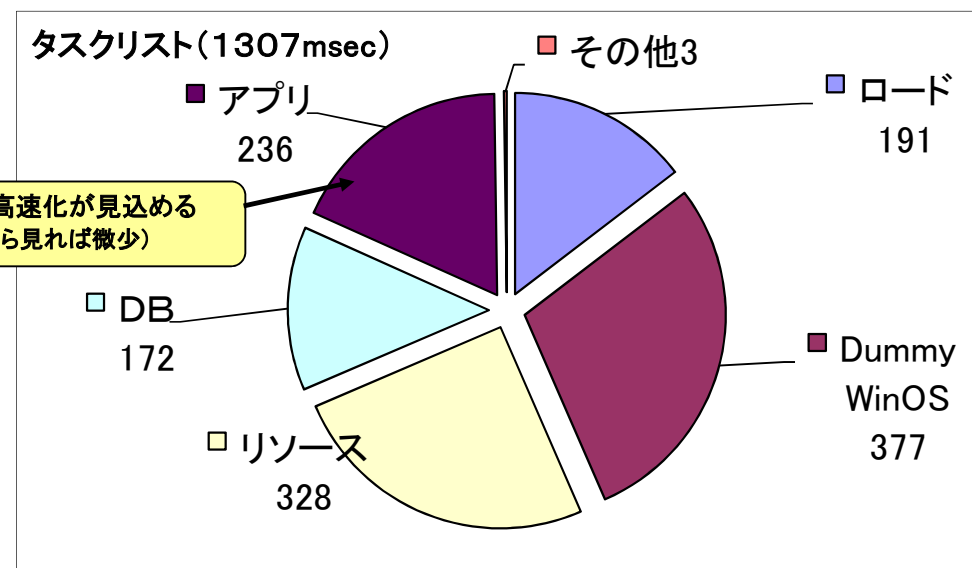
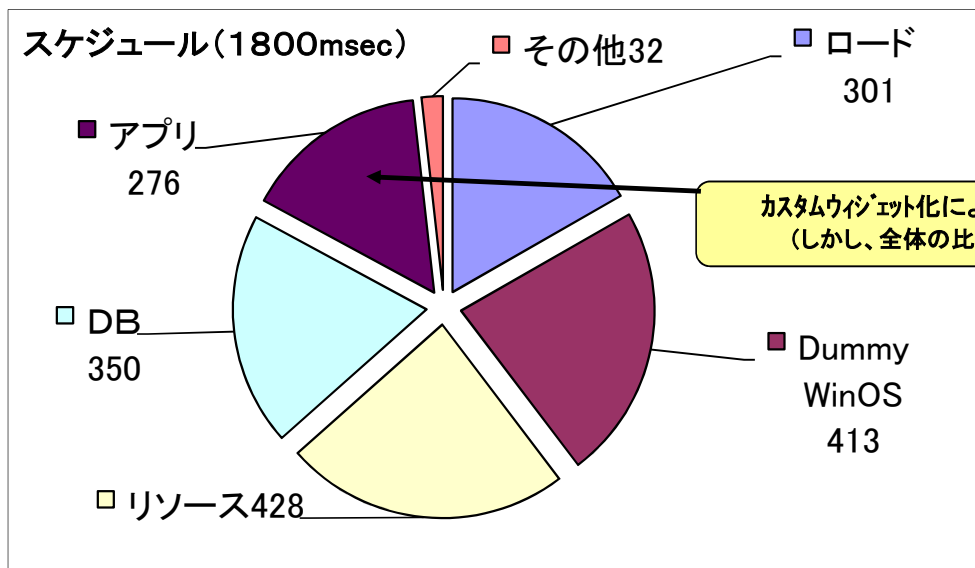
全ての起動処理の「85%」(15246msec)は
アプリ以外での処理となっている為、アプリのみの改善では、
目標値(=前モデル・ネイティブ機と同速度)を実現できない。

◆アプリグループ以外に願うアクション：

目標を達成するならば、以下の改善案が必要であると伝える。

- ・AEEアプリ起動部の速度改善
- ・DummyWinOS起動部の速度改善

パフォーマンス内訳（アプリ起動） 各機能比較



アプリ部の改善余地について

スケジュール 276 msec (改善の余地は少ない、カスタム化はインパクト大。)

タスクリスト 236 msec (改善の余地は少ない、カスタム化はインパクト大。)

カレンダー 52 msec (改善済み。)

「カスタムウィジェット化」で、仮に2倍の速度になったとしても、140msec程度。
カスタムウィジェット化は、モジュールの作り直しが発生する為、インパクト大。

- ... AEEアプリモジュールロード完了まで (AEECreateInstance->EVT_APP_START_WIN)
- ... DummyWinOS Window起動完了まで (PROP_WINDOW_VISIBLE)
- ... NAND画像リソース取得 (SHELL_LoadResImg系)
- ... SQLiteDBアクセス
- ... アプリ ウィジェット配置
- ... アプリ その他 (初期化、画面遷移等)

※1

カレンダー起動時は、アプリ実装は同じなのに、AEEロードとWindow起動完了時間が短かった。
これはおそらく、「スタティック実装」・「ダイナミック実装」のロード速度差等と、マルチアプリ起動によるメモリ圧迫等が関連しているのではと推察される。端末の実装に依存する内容である為、アプリ側のみでの速度差改善は厳しい。

パフォーマンス測定(詳細測定)

	アプリ	大項目	小項目	件数	W52SA	完了まで	KSW09						備考
							AEE アプリ ロード 完了 (AEECIs Create Instance)	Dummy WinOS Window 起動完了 (PROP_WI NDOW_ VISIBLE)	Qualcomm 画像リソース 取得 (SHELL_ LoadRes Img系)	SQLite DB アクセス	アプリ ウィジェット 配置	アプリ その他 (初期 化、 画面遷 移等)	
1	スケジュール	起動	カレンダーサブメニューから起動	5件	200以下	1531	300	414	449	61	278	29	DBアクセス無し DBアクセス無し
			カレンダーサブメニューから起動	100件	200以下	1666	301	413	447	198	278	29	
			カレンダーサブメニューから起動	200件	200以下	1800	301	413	428	350	276	32	
		カーソル移動	カーソル移動(画面スクロール無)	200件	100以下	5					5		
			カーソル移動(画面スクロール有)	200件	100以下	38					38		
		登録	新規1件登録	-	200以下	732				243	405	84	
2	タスクリスト	起動	カレンダーサブメニューから起動	5件	200以下	1285	192	375	331	158	226	3	DBアクセス無し DBアクセス無し
			カレンダーサブメニューから起動	50件	200以下	1307	191	377	328	172	236	3	
			カーソル移動	50件	100以下	1					1		
		登録	カーソル移動(画面スクロール有)	50件	100以下	83					83		
			新規1件登録	-	200以下	482				63	400	19	
			ツールメニューから起動	10件	300以下	1321	179	225	625	194	51	47	
3	カレンダー	カーソル移動	ツールメニューから起動	250件	300以下	1703	180	224	621	576	52	50	カスタムウィジェット化済 カスタムウィジェット化済 ※リストゼロで配置負荷低。 ※DB負荷が高い。 ※画面のみなので除外
			カーソル移動先にデータ無し	0件	200以下	8				4	4		
			カーソル移動先にデータ有り	10件	200以下	145				104	40		
		登録	カーソル移動先にデータ有り	250件	200以下	1536				1496	40		
			※他機能へ遷移のみ※										

※注:旧機種のログ取得方法は、
目視・ビデオ録画等による測定を行った。

カレンダーは、当日リスト描画の関係上、
カーソル移動を行う度にDBアクセスが発生する。
スケジュール・タスクリストとは、構造が違う。

アプリが単独で高速化できるのは、
この部分のみ。

※この資料はサンプルです。

数値はダミーです。