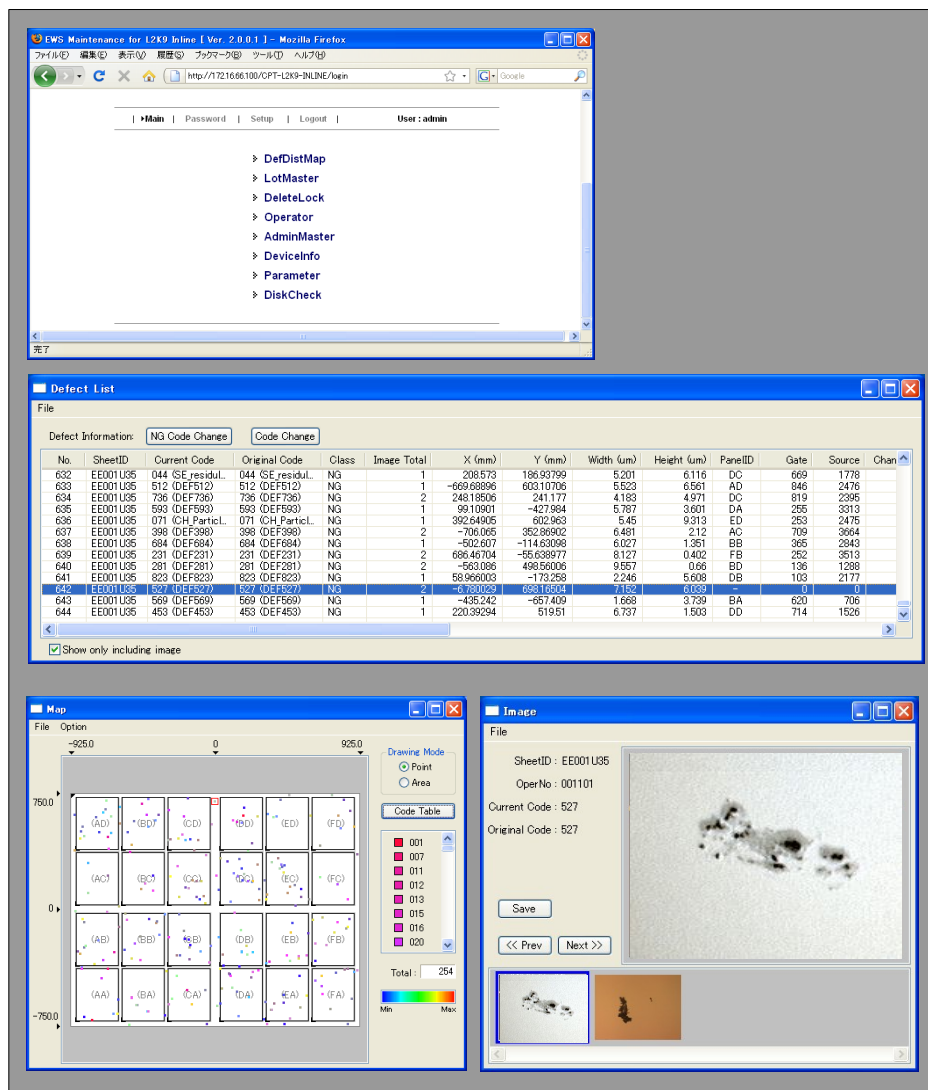


欠陥マップ表示システム



- 起動はブラウザから
- 端末ごとのインストールは不要
- 検索条件を指定してデータベースからデータ抽出
- 抽出したデータをマップ表示
- マップ表示とリスト表示と画像表示は常にリンクされている
- ユーザビリティに優れているアプリケーション

【開発環境】

Java SDK 1.6.0

【依存ソフト】

サーバ側：

Linux
Apache2.0
Java SDK 1.4
Tomcat4.1
PostgreSQL 7.3

クライアント側：

WindowsXP
Webブラウザ
Java Runtime Environment 1.4

【アプリ概要】

工場で生産される液晶パネルには、製造過程で欠陥が生じることがあります。検査装置ではガラス基板全体をスキャンして欠陥を検出し、その欠陥を情報格納したファイルをサーバへ渡します。サーバではデーモンプログラムが常時動いており、渡された欠陥情報を解析しデータベースに蓄積します。工場のスタッフの方々はそれら欠陥情報を閲覧し、場合によってはその情報を工場全体を管理するシステムに報告する必要が生じます。この欠陥マップ表示システムはそれらの業務を簡単に行えるようGUIアプリで提供しており、データベースやファイルサーバに蓄積された情報をわかりやすく表示することができます。

工場のシステムは製造する品目の仕様が変わると制御プログラムやそれに付随するプログラムもカスタマイズして運用しているという現実があります。プログラムのカスタマイズは現状では避けられないため、柔軟に対応することが望まれています。つまり仕様の変更があったら、少ない工数で変更ができることが望まれているのです。ここで紹介しているGUIプログラムはそれらにスピーディに対応するためJavaのアプリケーションとして実装しています。

案件ごとの事情に合わせてカスタマイズしたプログラムをご提供いたします。

【お問い合わせ】

有限会社ユニコムシステム
Tel:0984-48-5588 Fax:0984-48-5556
〒889-4314 宮崎県えびの市大河平4144
unics@unics.jp
<http://www.unics.jp/>

欠陥マップ表示システム(詳細)

【液晶パネルの製造工程】

液晶テレビやパソコンの液晶モニタには液晶パネルが組み込まれています。液晶パネルは「アレイ製造工程」→「アレイ検査工程」→「セル製造工程」→「セル検査工程」→「モジュール組立工程」→「モジュール検査工程」の過程を経て製造されています。

最初のアレイ製造工程は巨大なガラス基板上に薄膜トランジスタを形成していく半導体製造工程であり、製品の歩留まりを上げるためにこの工程内でさらに検査をしています。これは「インライン検査工程」と呼ばれます。

弊社のプログラムは上記工程のうち「インライン検査工程」「アレイ検査工程」「パネル検査工程」「モジュール検査工程」の全ての検査工程で使われていますが、ここでは「インライン検査工程」で使われているプログラムを例として説明します。

【ガラス基板の検査】

液晶パネルの材料であるガラス基板はのサイズ以下のように決められています。

第8世代基板サイズ (2,160mm×2,460mm)

第9世代基板サイズ (2,400mm×2,800mm)

第10世代基板サイズ (2,880mm×3,130mm)

ガラス基板にトランジスタの薄膜を何層も重ねていく途中で、光学的な検査を行います。つまり写真を取り、画像を解析して異常と思われる点のサイズと座標と欠陥種別を取得します。これは検査装置の仕事です。検査装置はネットワークで接続されたサーバの所定のディレクトリにこの検査結果を書込みます。

【サーバのデータ解析用デモンプログラム】

サーバは検査装置から受け取った検査結果を解析します。OSはLinuxであり、弊社が言語で開発したデモンプログラムが常時起動しています。このデモンプログラムは常に、検査装置から検査結果のデータが渡されていないか確認をしており、もしデータが来ていたときは、そのデータの解析を行います。

検査装置から渡されたデータは正常な場合ならばあてははいけない情報です。デモンプログラムはその情報とユーザが別途作成していた判定用のパラメータから、それらの情報が欠陥であるかどうかを判断し、結果をデータベースに格納します。

【データベース】

データベースにはPostgreSQLを使用しておりサーバ上で動作しています。検査データの項目は1000種類に及ぶこともあり、複雑なデータを管理するためにはデータベースを利用する方が便利です。

【作業端末】

液晶パネル工場内でスタッフが操作する作業端末はWindowsパソコンです。このパソコンにインストールしなければならないプログラムはブラウザとJavaのランタイムだけです。解析データを表示するためのアプリケーションはインストールする必要はなく、作業を開始するたびにブラウザのメニューをクリックするだけで、解析サーバからプログラムそのものが送られてきて自動的にクライアント側で起動します。

【欠陥マップ表示クライアントプログラム】

このプログラムはJavaで構築されています。GUIにはSWTというツールキットを使用しており、軽快に動くよう作成されています。このプログラムは下に示すように複数のウィンドウから構成されています。

検索画面：日付や製品ロット番号を指定してデータベースから所望のデータを検索します。検索結果が検索画面に表示されますので、ここで再度欠陥マップを表示したい製品ロットを選択します。この検索画面からだけでも各製品ロットの所定の工程までの製品に含まれる欠陥データを閲覧することができます。問題のありそうなデータをユーザが判断して選択することができます。

基板ガラス情報：今はインライン検査工程の結果を表示することが目的です。この工程における製造物は薄膜トランジスタが形成された基板ガラスです。それを検査装置で検査し、そのデータをデモンプログラムが解析してデータをデータベースに格納するところまではご説明しました。データベースには基板情報のみを保持するテーブルを定義しており、ここで表示されるデータはこのテーブルの内容です。

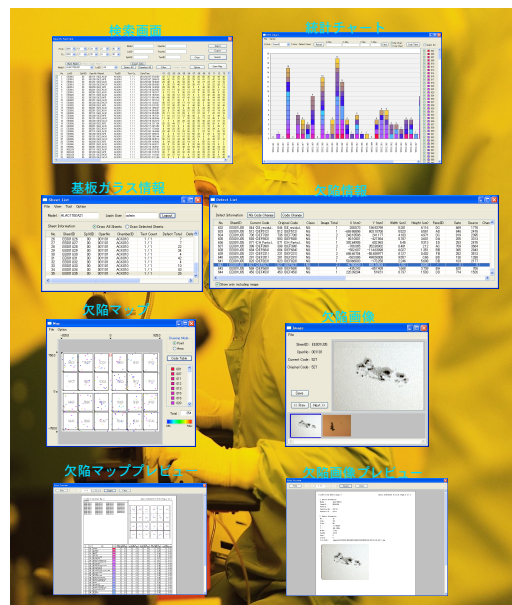
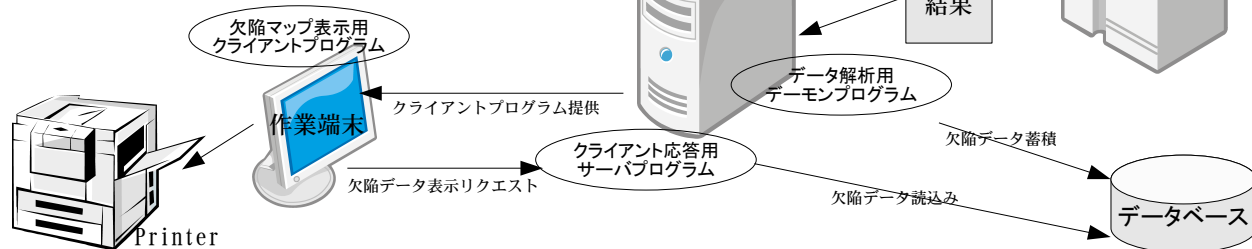
欠陥情報：基板ガラス情報に表示されているデータにひもづいている欠陥データをこの欠陥情報ウィンドウに表示しています。基板ガラス情報ウィンドウにあるチェックボックスを変更すれば、全て表示するのではなく、選択した基板ガラスにひもづく欠陥情報のみをこのウィンドウに表示することができます。大量にある情報のなかから目的のものを簡単に絞り込むことができます。

欠陥マップ：欠陥情報ウィンドウに表示されている情報をマップ表示します。ウィンドウサイズを変更すればそれに応じて表示領域も拡大、縮小します。マップ表示領域をマウスでドラッグすると、選択領域をウィンドウサイズまで拡大して表示することができます。欠陥情報を定義することができますので種類ごとに設定された色で欠陥マップが表示され、視覚的にもわかりやすい表示になっています。

欠陥画像：欠陥情報ウィンドウまたは欠陥マップウィンドウで欠陥を選択すると、それにひもづく画像を表示します。画像もマップと同様に拡大縮小表示機能をもっているのが希望の場所をドラッグすることで拡大表示できます。

統計チャート：欠陥数を視覚的に見るようにするため各種統計チャート表示プログラムを用意しています。棒グラフ表、折れ線グラフなどの表示が可能です。データとしては基板ガラスごと、ロットごと、欠陥種類ごと、欠陥サイズごと、検査装置ごと、検査時間ごとなど様々な切り口から欠陥数をグラフ表示することで、問題のある要素を見つけやすくなっています。

プレビュー機能：欠陥マップと欠陥画像表示ウィンドウにはプレビューウィンドウを用意しており、プリンタからの印刷イメージを画面上で表示することができます。そこに表示されたものと同様のものを印刷で得ることができます。



【クライアントプログラム】

これまで述べたようなクライアント側GUIプログラムは欠陥マップ表示だけでなく、その他にも各種パラメータ設定用のアプリケーションにも使用しています。

あるお客様の例ですと、(1)欠陥マップ表示、(2)ロット管理、(3)ロット削除マネージャ、(4)ユーザ管理、(5)管理者管理、(6)製品情報パラメータ設定、(7)各種パラメータ設定の7つのプログラムを利用していたみたいです。

これらはすべてサーバ側でコンパイルされ、クライアント作業端末のブラウザからのリクエストにより起動されるプログラムですので、プログラムの配置はサーバにだけ行えばよいことになり、アプリケーション管理作業から解放されます。

クライアントプログラムはそれ自身がサーバとサーバレット間通信を行ってリクエストとレスポンスのやりとりを行っています。

【クライアント応用サーバプログラム】

作業端末で起動しているJavaUIアプリケーションはサーバ内のjavaプログラムとapache tomcatを介してサーバレットにより通信を行ってデータのやりとりを行っています。工場内のネットワークに負荷をかけることは極力避けなければならないため、このようにリクエストのみをサーバに渡し、その結果だけをレスポンスとしてもらえば不要なデータのやり取りが少なくなるため、ネットワークにやさしいと言えます。

サーバ側のjavaプログラムはTomcatから起動されて実行されて所望の処理を行い、必要なデータを取得してまたtomcatに渡して、クライアント側に送ってもらいます。

サーバ側のjavaプログラムは、リクエストに応じて、データベースからデータを取得します。このときデータベースのテーブル1つに対応したデータだけを格納するためのBeanクラスを定義しておきます。このクラスはSerializableクラスを継承することでシリアル化してネットワーク上をテキストデータとして行き来することができるようになります。

データベースのデータを読んだこのBeanクラスに値を設定したり、逆にBeanクラスのデータをデータベースに格納するためのメソッドを持つクラスをデータベースマネージャクラスとして定義しています。

データの取得先や格納先リソースがデータベースでなくファイルの場合もあります。欠陥情報はパネルごとオペレーションごとにファイルとして保存されています。この場合はファイルのフォーマットごとにそのデータを格納するBeanクラスを定義します。そしてそのBeanクラスに対しファイルからデータを取得して書込むメソッドや逆にBeanクラスをファイルに書込むメソッドをもったファイルマネージャクラスを作成します。

1つのファイルフォーマットに対し、Beanクラスとファイルマネージャクラスがそれぞれ1つずつ存在するような作りになります。サーバ側でもクライアント側でもデータの保持はテーブルやファイルごとに定義されたクラスで保持しており、ネットワークの中はそれらのクラスをシリアル化したデータが流れているだけなので、取り扱いが簡単です。

【Webアプリケーション】

GUIプログラムは工場スタッフの利便性を考えると必須のものですが、簡単なプログラムならばGUIを使わず、ブラウザだけで処理が完了するWebアプリケーションの形で提供することもできます。

例えばログインパスワードを設定する機能など単純なものはすべてWebアプリケーションで作成され、操作性の必要に応じて作成するアプリケーションの種類を変えています。

【アプリケーションの操作性】

JavaのGUIプログラムの実現方法としてAWTやSwingを利用する場合もありますが、このアプリケーションではSWTを使っています。SWTの場合OSごとにライブラリが特化された作りとなっているので、汎用性は低くなりますが、操作性が良くレスポンスが速く感じられます。VC++などでWindowsのアプリケーションを構築したときと同程度の操作性が得られます。

またこれらのプログラムは工場内の生産現場で使われるため、場合によっては快適なオフィス環境内ではなく、クリーンルームなどで使われる場合もあり、その場合はクリーンウェアと手袋着用して作業をしなければならない場合もあり、操作性の良いアプリケーションを提供することは必須でもあります。

【サーバ内のミドルウェア】

サーバOSはLinuxです。クライアントからのリクエストをうけてレスポンスを戻す役割はサーバが担います。サーバには以下のミドルウェアをインストールして本システムの機能を実現しています。

サーバ側：

Linux
Apache2.0
Java SDK 1.4
Tomcat4.1
PostgreSQL 7.3

【クライアント側の必要環境】

クライアント側の保守が容易になるように、必要となるプログラムは以下のように必要最低限のものです。

クライアント側：

WindowsXP
Webブラウザ
Java Runtime Environment 1.4

【開発、デバッグ環境】

Java SWT GUIアプリケーションを開発、デバッグするための環境は以下を使用しています。

開発、デバッグ環境：

WindowsXP
Java SDK 1.6.0
Tomcat
Eclipse
Tomcat eclipseプラグイン
ブラウザ
ファイルサーバとsamba共有
開発用postgresqlサーバ

【カスタマイズ】

お客様の要望に応じてプログラムをカスタマイズしてご提供いたします。本システムは液晶パネル工場用に作成したプログラムですが、製造現場に限らずオフィス系のアプリケーションでも顧客管理などの用途にお使いいただけます。