

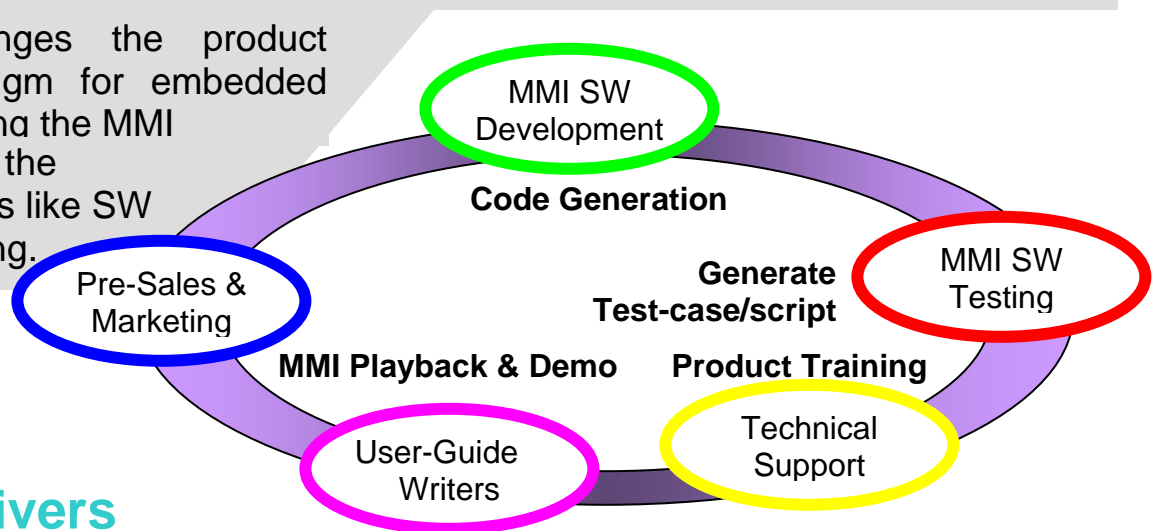
COVER THE ENTIRE STRETCH OF PRODUCT DEVELOPMENT

# EntireStretch

COVER THE ENTIRE STRETCH OF PRODUCT DEVELOPMENT

**EntireStretch** provides an integrated development environment for creation of multi-product MMI (Man Machine Interface / Human Machine Interface) specifications, playback of MMI and automatic generation of related testcases & testscripts. It supports the concept of multiple product variants right at the MMI specification level itself.

**EntireStretch** changes the product development paradigm for embedded systems by integrating the MMI Specification with all the downstream activities like SW development & testing.



## Business Drivers

**With the increasing role of ODM/OEM products, MMI becomes the key factor that differentiates the brand.**

In order to reduce the time-to-market and to lower the development costs, consumer electronics manufacturers are moving more and more towards the OEM/ODM model of product development. The same ODM supplier may be providing products to multiple & often competing manufacturers. In such a scenario the MMI becomes the main differentiating factor that can be related to the brand. The MMI specification becomes the key tool for aligning the product customization requirements.

Big customers (e.g. Cellular Operators, Retail Chains etc) often demand their own custom MMI/HMI for branding. They want to evaluate the look-n-feel at a very early stage in product development.

## MMI Playback is vital.

Traditional MMI documents fail to fulfill the needs of the increasingly rich feature set of present day gadgets,. They provide only a static view of the User-Interface. The dynamic behavior of MMI

## COVER THE ENTIRE STRETCH OF PRODUCT DEVELOPMENT

and the navigation flow across a sequence of screens is not depicted in an intuitive form. There is no support for MMI playback.

### **MMI Specification is not integrated with the downstream activities in the product development process.**

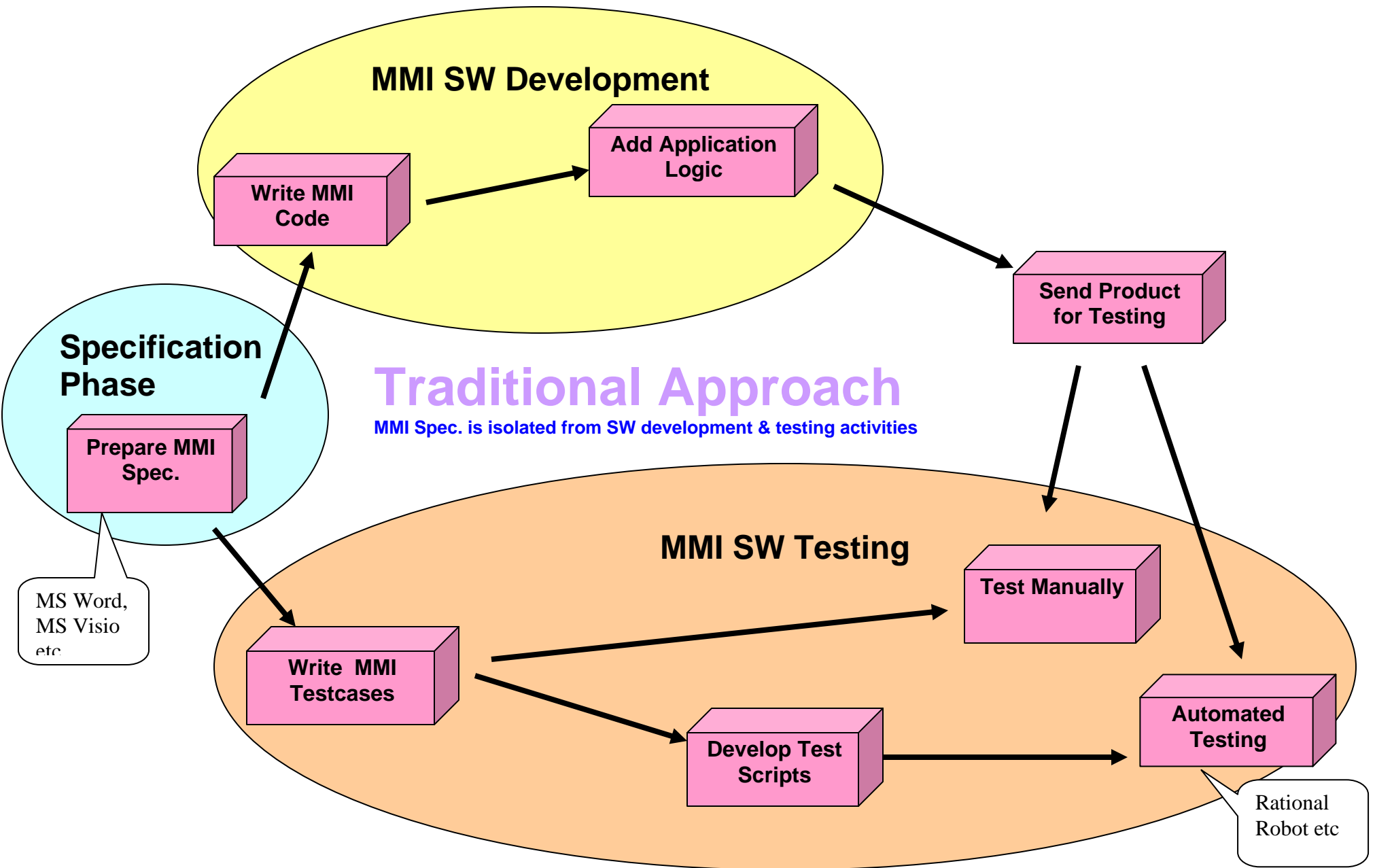
Often the screen layouts, text screens and icons/bitmaps get changed during the coding phase. This can cause a huge gap between the final product and the original requirements thereby resulting in customer disappointment. However this is caught too late in the development cycle.

In addition, a high percentage of the total number of bugs that are reported by testing team are often there just because the MMI Specification was obsolete. This presents a dilemma for the testing teams : what is the correct behavior –the implemented in code or the one specified in the MMI specifications ? Very often the testing teams base their test-cases taking the product as reference rather than the original MMI specifications.

The traditional method for development of MMI Specifications consists of primarily creating a sequence of MMI screen diagrams and providing a text narrative describing the MMI operations. The specification writers typically use the tools like MS Word, MS Visio, MS PowerPoint etc for documenting the MMI specifications. These tools provide set of drawing-objects that be used to create a MMI screen layout. This kind of MMI specification provides a static picture of the MMI.

There are certain disadvantages with the traditional method of MMI specifications:

- It captures only a static picture of the MMI. No support for playback of MMI usecases. User does not get a clear picture about the dynamic aspects of the device.
- There is no inherent support for linking MMI Screens to Flowcharts / State Diagrams.
- There is no support for planning multiple product variants at the stage of MMI specification. Reuse of the specification for the product variants is achieved by copying the contents from the master specification. This leads to lots of rework if the basic spec gets changed later in the project.
- No support for generating the testcases or testscripts directly from the MMI specification document. System Test teams have to spend considerable amount of time to create testcases/scripts.
- No support for performing any kind of consistency checks e.g. broken links, inconsistent references.
- No support for configuration management of MMI specifications.
- No support for text management. MMI design teams have to manage text listings in Excel Files and SW development teams often use totally different mechanism for managing text. Often the text strings in MMI specification are totally different from the text strings used by the programmers. No support for generating a language specific MMI Specification / User Guide.
- There is no easy way of comparing two versions of MMI specification other than manually reading both the versions. This is one of the leading causes of bugs creeping into the MMI SW as well as testcases/scripts.



# COVER THE ENTIRE STRETCH OF PRODUCT DEVELOPMENT

## EntireStretch Advantage

**Create & Manage multi-product MMI Specification for the entire product family/generation.**

- Support for multiple variants right at MMI Spec. stage itself.
- Share common MMI Screens by deriving from a base product.
- Setup active links between Flowcharts & MMI Screen.

### **Playback of MMI Specification.**

- Get a feel of the real product without writing any code.
- Vital marketing tool as product demo to big customers.
- More accurate representation of MMI behavior for in-house developers as well as ODM/OEM suppliers.

**Automatically generate majority of MMI Test-cases directly from the MMI Specification.**

- Lower the costs of test-case development.
- Reduce the lead-time for development of test-cases and improve the time to market.
- Automatically generate test-scripts for test automation tools.

## Value for Customers

By using the MMI playback mechanism the customer can provide a more accurate representation of the MMI requirements to the product developers. Similarly the ODM suppliers can demonstrate the MMI behaviour even at the product conceptualization stage without building any prototype. This can provide highly competitive advantage by reducing the time-to-market and lowers the development cost.

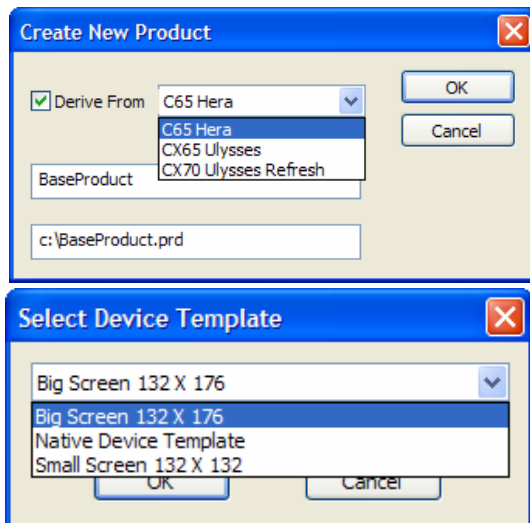
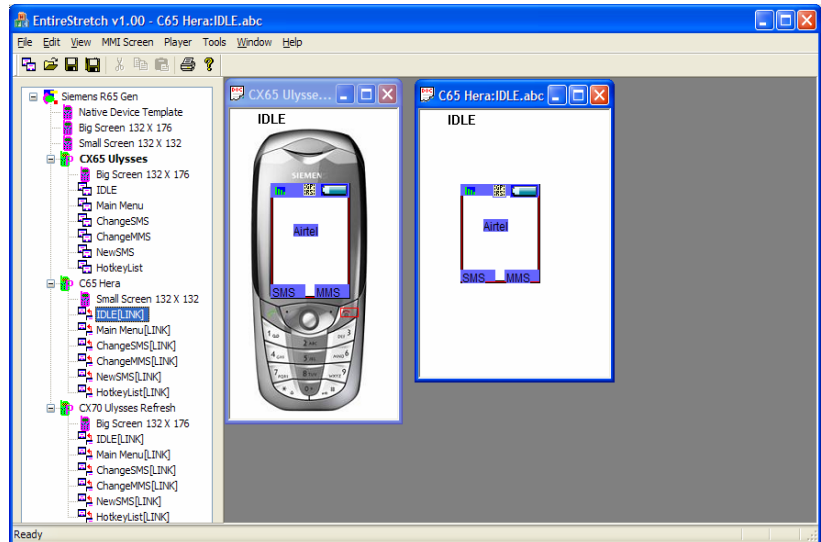
- Demonstrate the MMI behavior at the product conceptualization stage itself without even building any SW prototype.
- MMI Playback can be used as a key marketing tool for introducing new consumer gadgets to big customers (e.g new Digital Camera for retail channels like CircuitCity, new mobile phones for Service Providers like Vodafone, T-Mobile, Cingular etc. )
- Easily manage the complexity of multi-product MMI specifications. Reduce the cost by sharing the MMI screens & navigation flow across different products of a generation. No need to redo everything for new products.
- Enhance the productivity of test-case development and reduce the costs by automatic generation of MMI test-cases. Improve the reliability of test-cases by generating test-cases directly from MMI specification itself.
- Reduce the costs by automatic generation of the MMI test-scripts.
- Generate text strings/icons/animations for the development team directly from the MMI Specification itself.
- Version comparison feature allows users to identify which MMI Screens / Flowcharts have been changed since a particular release.

# COVER THE ENTIRE STRETCH OF PRODUCT DEVELOPMENT

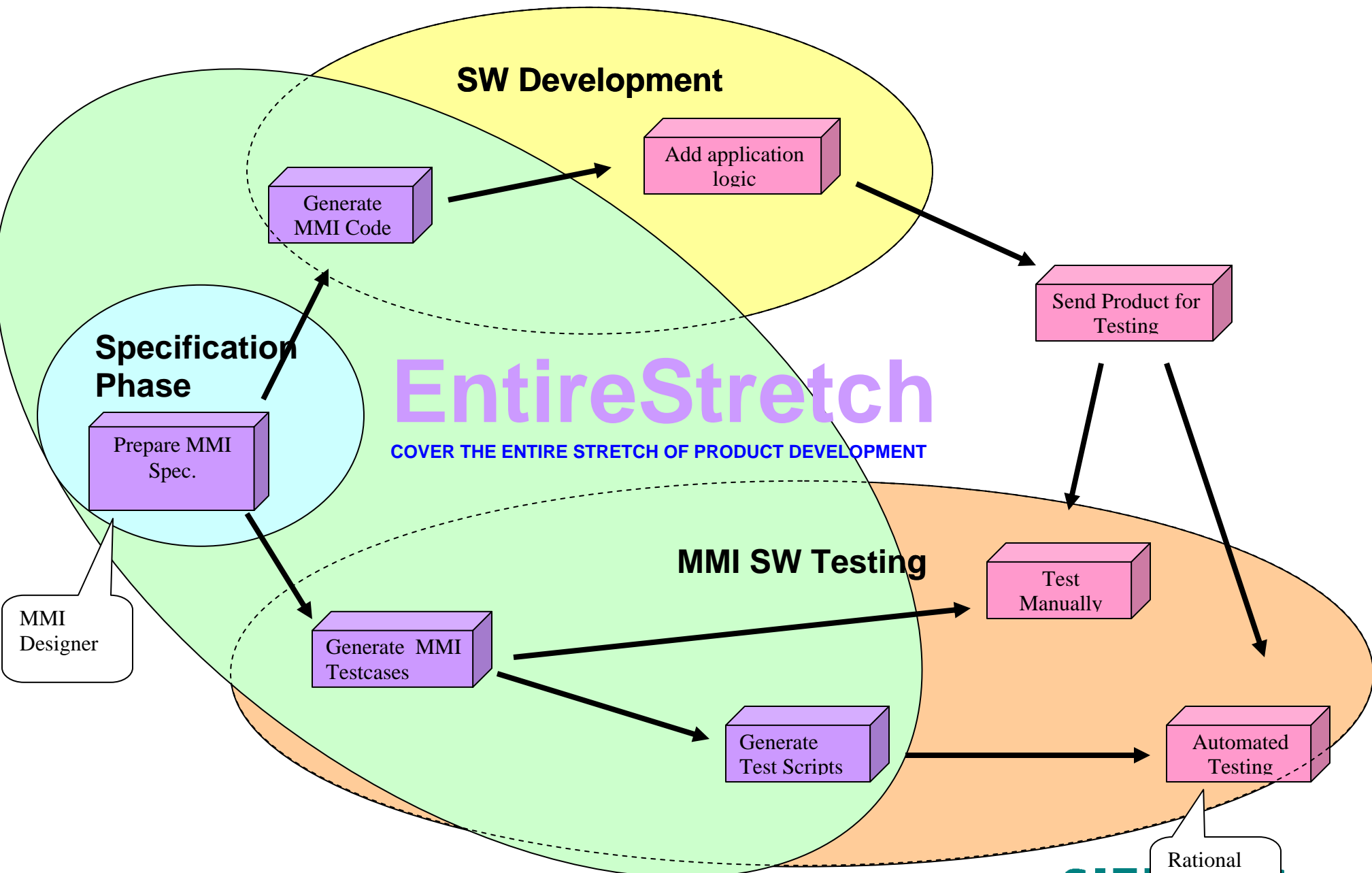
## Multi-Product MMI Specification

The support for multiple product variants allows the product managers to manage the entire family of products with having to develop the MMI specifications for each variant separately. They can easily determine the products that get affected due to a change in one of the screens or use-cases in the MMI specification. This also helps the technical writers who often have to wait till end of the project to get enough details of the MMI screens for writing the user guide.

- Define Product Family consisting of multiple Device Templates & multiple Products.
- Device Template holds the model specific details like Screen resolution, number of keys, type of keys – Soft Keys, Side Keys, Back Key, Operator Key etc
- Each product is linked with one Device Template.

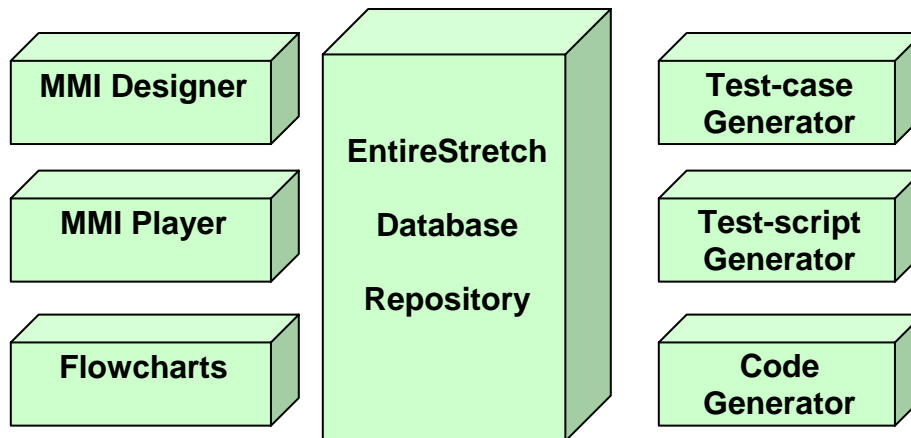


- Share the common features by deriving product from another product. All the MMI Screens from the base product are linked to the derived product.
- The derived product may override the functionality of these screens.
- MMI screens are rendered according to the display capabilities of the active device template. For majority of the features you have to create screen only once. The tool takes care of the changes in appearance due to changes in display resolution etc.
- Easy to understand views/reports help in analysing the impact of a specific MMI screen or use-case on different products.



# COVER THE ENTIRE STRETCH OF PRODUCT DEVELOPMENT

**EntireStretch** provides an integrated environment for creation of MMI specifications, playback of MMI and automatic generation of related test-cases. It supports the concept of multiple product variants right at the MMI specification level itself. The entire suite consists of following components:



**MMI DESIGNER** allows the end-user to very easily create a pictorial representation of the proposed MMI. It provides a rich library of UI Objects & Widgets. This library can be tailored to the specific requirements of the customer. By simple drag-n-drop operations the user can quickly specify the layout of on-screen objects. It can be used to generate the traditional MMI specifications (Word, Visio etc) automatically.

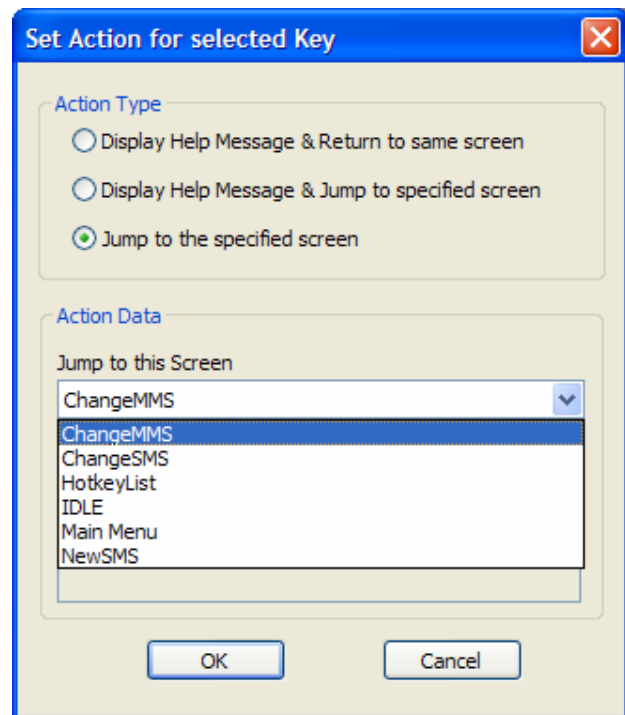
- Create MMI Screens using Drag-n-drop operations.
- Add narrative/annotations with each screen
  - Abstract / Overview
  - Detailed Description
  - Special Notes
  - Development Notes
  - Testing related Notes
- Generate MS Word documentation
- Generate MMI Navigation Tree
- Assign Actions to desired keys
- Specify Actions for non-Keypad events (HW events, external network events etc)

The screenshot shows a dialog box titled "MMI Screen Documentation Information". It contains several text input fields for documentation: "Abstract" (with a preview of text), "Detailed Description", "Special Points / Notes" (with a preview of text), "Spec Writer's Comments", "Development related comments", and "Comments related to testing aspects". There are "OK" and "Cancel" buttons in the top right corner.



**MMI PLAYER** Once the MMI screens have been created, they can be linked to depict the flow of operations without writing any scripts. **MMI Designer** allows users to generate an executable MMI-Player that can be used for customer demonstration. The MMI player enables the user to walk through the MMI screens in a manner very similar the actual product.

- During preparation of MMI Screens, the user can associate Actions with different keys and specify what screen should be displayed next (without writing any code/scripts!). The tool stores the MMI Screen links & Action mappings in an internal database.
- In the playback mode, the tool displays the screens based on the keys pressed by the user.
- Playback provides an almost-real feel of the product MMI without actually writing any code. User can navigate through the entire MMI similar to the actual product.
- Non key-pad inputs can also be simulated (e.g. Incoming Call, Accessory connection, Changes in battery power etc)
- Great for showing product demo at very early stages in the product development lifecycle. It can be used for training customer care & service personnel.



**TEST-CASE GENERATION** **EntireStretch** can automatically generate test-cases from the MMI specifications thereby reducing the chances for error related to subjective interpretation of MMI specifications. These test-cases can then be processed by **EntireStretch** to automatically generate test scripts for GUI test tools like Rational Robot etc.

- Based on Key-Action mapping specified during creation of the MMI Screens, the tool can generate test cases to exercise the related functionality.
  - Automatically generate the entire MMI navigation tree and related test-cases.
  - Automatically generate test-cases for next N steps starting from a given MMI Screen.
  - Automatically generate test-cases till some pre-defined sync points starting from a given MMI Screen.

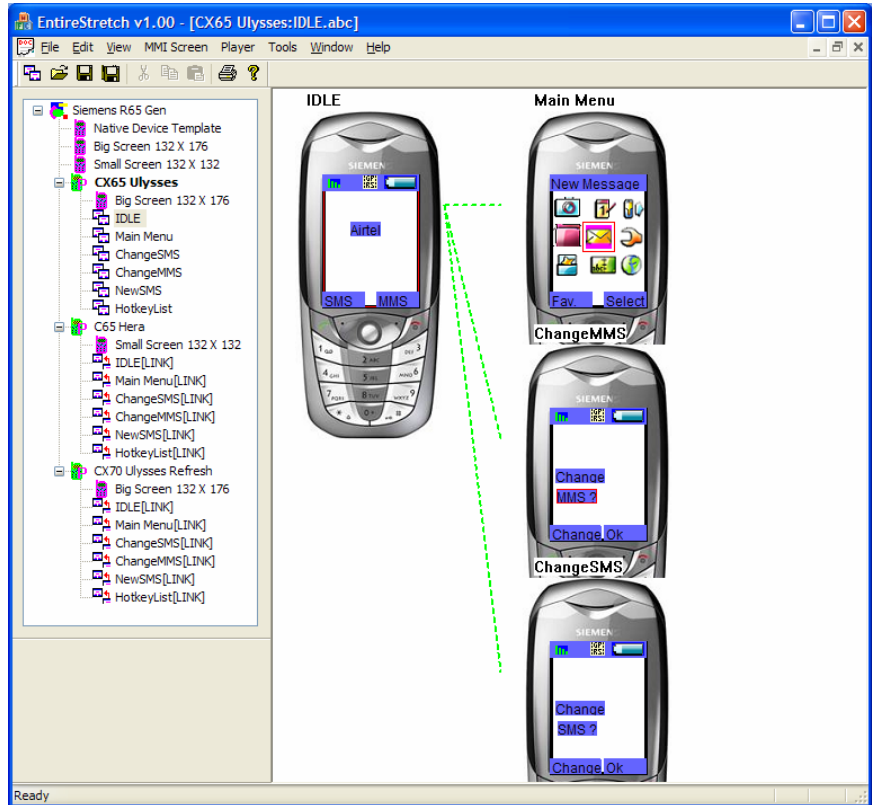


# COVER THE ENTIRE STRETCH OF PRODUCT DEVELOPMENT

- Test case Wizard walks through the MMI navigation tree and lets the user pick and choose the actions to build custom test-cases.
- Generation of MS Word document.
- Significant improvement in productivity of test-case development team.

The test case designers require a higher grade of skills & know-how as compared to the actual testers. By automatic generation of MMI test cases, the testing teams can lower their costs related to the testcase development. It can shorten the lead-time for testcase development.

**EntireStretch** uses the specification as the basis rather than the actual MMI itself. Since the tool generates the testcases based on the MMI specification, the testcases are expected to be more accurate. This can help in increasing the reliability of the testing process.



**MMI Code Generator** Once the MMI screens have been created, and linked to depict the flow of operations, the **code generator** tool can generate skeleton C/C++/Java code for the target device and the developer simply needs to fill in the code just for interfacing it to the application layers.

## MMI Resource Database

- All the text strings (multi-language), bitmaps, animations, icons & other graphics resources are maintained in a database repository.
- It generates special source code files that can be used directly by the development team. This ensures that the development team actually uses the same resources as specified in the MMI spec.
- MMI Playback can be performed using different language sets.
- MMI Playback can be performed using the customer branding specific skins & themes etc.
- Consistency checks can detect missing text definitions in non-English languages. This can catch language translation related bugs much earlier.

# COVER THE ENTIRE STRETCH OF PRODUCT DEVELOPMENT

## Summary

**EntireStretch** enables embedded product development teams to easily manage MMI Specifications for multiple-product variants. It helps the pre-sales, Marketing & Service personnel in creating demo/playback of MMI even at the conceptualization stage itself.

### Traditional MMI Specification

- Manual MMI Specification
- No support for Playback of MMI
- Complex to manage MMI spec for multiple products.
  - Screens-shots have to be created in multiple display resolutions
  - No easy way to find which screens are common/different
- Manual & time consuming process to generate MMI Test-cases. Long lead time for test-case development.
- Documents in Word/Visio. Format cannot be changed later.
- No inherent support for config. Management.
- No support for Text/Graphics management.

### EntireStretch Specification

- Tool based MMI Specification
- Playback helps in more accurate definition.
- Easy to manage MMI spec for multiple products.
  - Screen resolution related changes are handled automatically
  - Impact of screens/use-cases on different products
- Automatically generate MMI Test-cases.
- Reduce lead time & cost.
- Document Type & Template can be configured.
- Support for version labeling. Interface to CM tools.
- Inherent support for Text Manager.